

**ECMS 2005  
SCIENTIFIC PROGRAM**



# Plenary Papers



# Human Factors in Complex Systems

## The Modelling of Human Behaviour

Bernd Schmidt  
University of Passau  
Innstr. 33, 94032 Passau, Germany  
<http://www.or.uni-passau.de>

### SUMMARY

Human beings are most often an integrated part of complex systems. In order to describe such a system with appropriate accuracy it is necessary to model the human components with the same accuracy as the technical components. Human factors must be included and modelled with the same degree of precision as the system's mechanical parts.

A human being is perceived as a psychosomatic unit with cognitive capacities embedded in a social environment. Human behaviour is structurally highly complex. As human behaviour is influenced by physical, emotional, cognitive and social factors, it is highly intricate. Consequently, a human being is perceived as a psychosomatic unit with cognitive capacities embedded in a social environment.

The PECS reference model makes it possible to specify and model these factors and their interactions.

PECS stands for:

Physical conditions  
Emotional state  
Cognitive capabilities  
Social status

The PECS reference model aims to replace the so-called BDI (Belief, Desire, Intention) architecture. [Rao 1995]. Architectures, such as BDI, which conceive of human beings as rational decision makers, are sensible and useful to a very limited degree only. Restriction to the factors of belief, desire and intention is simply not appropriate for sophisticated models of real systems where human factors play an important role.

A detailed description of the PECS reference model and its underlying methodology including some basic examples can be found in [Schmidt 2000] and [Schmidt 2005].

The actual importance of models that include human factors will be shown in a case study which is intended for use in real decision-making processes.

- PAX

PAX is a model, which is used to describe peaceful military operations in the distribution of food care parcels in an occupied war zone. The aim of the model is to investigate strategies for the soldiers under varying circumstances. It is obvious that in a situation like this, aside from rational and cognitive

considerations, emotional and social (especially psychosocial), aspects have to be taken into account.

Further examples are:

- MedSim

As prevention and screening become more important in health care, it would seem obvious to investigate the intended measures for screening by means of a simulation model before the actual implementation of the measures. Models to evaluate screening programs for the early detection of diseases must include factors related to the patients' compliance. Compliance is determined by physical, emotional, cognitive and social influences. Therefore, the PECS reference model offers an adequate framework for these investigations.

A detailed description can be found in [Brailsford 2002]

- Adam

Recently the importance of emotional intelligence has been realised. It is obvious that emotional control and social competence are more decisive than pure intelligence in determining success in coping with difficult problems, which embrace human beings in various settings. The Adam model describes the process of emotional control, its dependencies on other factors and its consequences on behaviour and decision-making.

A detailed description can be found in [Schmidt 2000]

The PECS reference model opens up new challenging possibilities for the modelling of systems that include human factors as important and decisive subcomponents. PECS is especially useful when complex human behaviour has to be taken into account. This includes physical conditions, emotional states, cognitive capabilities and the social status along with accompanying mutual interactions.

### 1 THE MODELLING OF HUMAN BEHAVIOUR

The first question, in modelling human behaviour, is whether the attempt to investigate human beings scientifically and capture their nature in a model is at all possible or whether it is simply an example of hubris.

### 1.1. The Unfaithful Nature of Human Beings

Human behaviour is determined by a wide variety of influencing factors, which interact in complex ways. The following examples chosen at random serve to illustrate this:

- **Personal Experience**  
The behaviour of human beings is influenced by their life history and by the experiences they have had. These include early childhood experiences as well as consciously learned experiences through interactions with their environment.
- **Social Norms and Role Expectations**  
Every human being, to a certain extent, conforms to the norms and role expectations which society imposes on them. These demands often conflict with their individual wishes, plans and ambitions.
- **The Unconscious**  
Human beings are very often not conscious of their own actions. Again and again they act or react in ways which surprise them or which they did not expect of themselves. Is St Paul not right when he says: "I do not do the good that I want to do, but I do the evil that I do not want to do" (Rom. 7, 19)?
- **Conflicting Motives**  
In conflict situations human beings are torn between different motives that vary and are often in conflict with each other. Should they in the Kantian sense, for example, do their duty or should they rather follow their inclinations?
- **Human Freedom**  
One also has to take into account free will, which enables human beings to decide what to do, who to trust, what roles to play and what norms to follow. These decisions take place independent of external influences and stimuli and are based solely on the individual's personality and free choices.
- **The Experience and Response to Art**  
In the arts, human beings have access to a world of experience, which eludes logical assessment. Beethoven noted in his diary: "Music is a higher revelation than all wisdom and philosophy."

Can all this be modelled?

Can we really fully understand human beings? Will we really be able to reproduce their behaviour in a computer model?

Do these examples indicate that human beings are far too complex, far too contradictory and far too unfathomable to be made comprehensible and in turn predictable by means of a model? Would such an attempt be bound to remain superficial and shallow and thus fail to capture precisely what constitutes being human?

Should we agree with Pascal, who wrote in the *Pensées*:

*What a chimera the human being is! Wonder, confusion, contradiction! Judge of all things, powerless earthworm, dark room of uncertainty, the glory and the shame of the universe. When he praises himself, I will*

*humble him; when he humbles himself, I will praise him; and I will go on contradicting him until he comprehends that he is incomprehensible.*

Is the human being truly incomprehensible?

Of course it can be assumed that human behaviour is very complex and many-layered. The PECS research project however, is based on the conviction that it is possible to reduce this complexity by means of conscientious functional decomposition and to dissolve this multi-layered quality by carefully isolating the individual layers. These layers are first studied in isolation and then their interaction and their interplay are investigated. To these ends, it is hoped we can approximate a deeper understanding and comprehension of human behaviour.

### 1.2 The Difference between Model and Replica

In order to answer the question, "can human behaviour be modelled?", we must distinguish between developing a *model* of human behaviour and producing a *replica*.

A replica is an identical copy of an original. It is completely indistinguishable from the original. It appears to be impossible, at least for the foreseeable future, to produce an artificial replica of a human being.

A model, on the other hand, is an abbreviated depiction of an excerpt of reality based on abstraction and idealisation. It does not have to conform to reality in every aspect and all respects. An example of such a model is that of the model aeroplane used in wind tunnel experimentation. Such insights are valid in the human sciences too. For example, literary and historical scholarship have developed a picture and hence a model of Goethe. This picture is of course not identical to the real Goethe. It does not claim to be a replica. Nonetheless it does provide useful and useable insights. The more precise and the more detailed our image and thus our model of Goethe is, the better he can be understood and the more accurately his behaviour in a certain situation can be predicted. Not exactly of course, but in terms of a general tendency. If we had a good model of Goethe, we would know something about his physical condition, his emotional state, the state of his knowledge of the world and his social position. It would then be conceivable we could understand, for example, why at an advanced age he falls in love with a young woman. We would even perhaps be in a position to forecast something of the kind. We would not be able to state exactly where and when this would take place. But we could assume that it might happen.

A robust and useful model, capable of providing valuable insights, does not necessarily have to be insurmountably complex and difficult. It could turn out in fact to be quite simple, so simple that modelling could be successful. This means a model of a human being does not necessarily have to contain all the qualities that distinguish the individual as a human being. One can try to begin in a simple way and concentrate on the dominant facts in the problem under investigation.

Applying this to the PAX model of peaceful military operations, this insight leads us to hope that such a project is not impossible from the outset.

In order to understand the behaviour of soldiers and civilians and make it predictable within limits, it is not necessary to model the participants in all their complexity. There is no need for a replica. Many qualities and modes of behaviour that normally typify an affected person can be sacrificed to the filtering abstraction and idealisation process without rendering the modelling process completely futile.

### **1.3 Models of Human Behaviour in the Empirical Sciences**

When considering the modelling of human behaviour, it should be kept in mind that excellent models of human beings and their behaviour already exist in certain disciplines.

Physiology has developed very detailed and expressive models of the human body and its behaviour in changing circumstances. It is possible to model, understand and predict physical and chemical processes in the human body.

In a similar manner, psychology attempts to develop models of human psychological life which deepen our understanding of internal processes. This category includes cognitive aspects such as intelligence, learning, memory and powers of imagination. In addition, considerable work is being done to improve our understanding of emotion in its healthy as well as pathological forms.

Sociology also works with models and attempts to understand human behaviour in its non-individual form, i.e. in relation to society. Sociological models investigate, for example, the development, the passing on and the implementation and development of norms. In both cases, sociology attempts to identify human behaviour in social groupings.

If one were to question the explanatory value and prognostic capacity of these models, then physiology, psychology and sociology would cease to exist as no one would wish to adopt such a stance.

Everyday experience provides a further example. The better one knows a person, the better one is generally able to understand this person and in some circumstances even to predict their behaviour.

These arguments have shown that it should be possible, at least in certain areas, to model and understand human behaviour and make it reasonably predictable. However, the important proviso remains. Many people themselves consider a model to be an abbreviated and crude version of the original, never being identical. A model of an individual is fundamentally different from the individual themselves. Nevertheless a model can be useful and meaningful.

Many critics who doubt the possibility of modelling a human being are not aware of this point. They confuse a model with a replica and because of this become engaged in incomprehensible polemics.

### **1.4 The Human Being as a Psychosomatic Unit with Cognitive Abilities in a Social Environment**

The sciences have so far concentrated on investigating partial aspects of human behaviour under laboratory conditions appropriate for each field of research. As a result there has been a tendency to lose sight of the interactions between the emotional, the cognitive and the social areas. It is a fundamental conviction of the PECS research program that an understanding of human behaviour can be achieved only if all 4 aspects and their interaction are taken into account. According to this interpretation, a human being is a psychosomatic unit with cognitive capacities who is capable of surviving only in society. Their behaviour will always be determined and shaped by the interaction between their physical situation, emotional state, cognitive capacities and social position.

As soon as one attempts to model real human behaviour, it is essential to have a reference model that permits the possibility of this interaction and this interplay. Engineering sciences with their architectural models and theory with its virtual realities do not need to take this possibility into account.

### **1.5 Behaviour Control**

Generally speaking we can start by assuming that every organism has certain needs, which it wishes to satisfy. In the course of time, evolution has constantly developed and improved more powerful forms of behaviour control in order to guarantee the satisfaction of these needs. For reasons of clarity it is useful to distinguish the following forms of behaviour control:

- Reactive behaviour
- Deliberative behaviour
- Reflective behaviour

We can assume that a human being, as a product of evolution, has all these modes of behaviour control at their disposal. They are capable of the highest form of reflective behaviour without having completely liberated themselves from elementary forms of instinctive behaviour. The human being is a citizen of several worlds.

A reference model that presumes to model human beings as a whole must provide an architecture that makes it possible, in principle, to model all forms of behaviour control.

## **2 THE HUMAN BEING IN THE MODEL**

If human beings with their diverse modes of behaviour are to be represented in the model, a fundamental concept is at first required. The present study assumes that the model of a human being should have the structure of a system.

### **2.1 System-Theoretical Principles**

A system in terms of system theory is first charac-

terised by unequivocally defined state variables. These state variables can change their value on the basis of their own dynamism or an external input. The modified internal system state will then lead to an output, which one can regard as an action.

This can be shown in the simplified diagram 2.1.

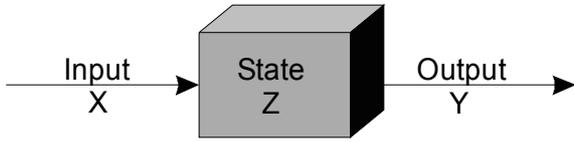


Diagram 2.1. A System with Input and Output

A number of simple examples will illustrate how the fundamental concept of a system in the case of a human being could look:

- Let the internal state variable be body temperature. By taking a medicine as an input the body temperature is increased. This leads to observable behaviour that causes the person concerned to take a towel and wipe away the sweat.
- Let a possible internal emotional state of a person be joy. Let it at first have a low value. A positive piece of news from outside leads to an increase in the value of the joy state variable. The person might then respond by jumping in the air or by yodelling.
- The state of a person's knowledge may also be described with the aid of state variables. A person's knowledge level may be increased by the acquisition of information. As a result the person is now able to visit a restaurant, the location of which he has just found out. He knows the place coordinates.
- A person's social status may be increased by a promotion as input. He may be promoted from the position of subject teacher to that of head of department. This new state as head of department leads him to take new actions that have now become possible. He could for example rent a new and better flat.

These examples may appear somewhat artificial and even silly in their simplicity as they certainly do not do justice to the complexity of human behaviour. However their function here is simply and solely to illustrate the basic underlying principle.

A further objection to the concept of producing a model of the human being on a system-theoretical basis has been raised by representatives of the human sciences. They consider a terminology which talks of internal state variables being modified by external inputs which lead to actions as output as mechanistic, technocratic and therefore inappropriate. Such terms, they argue, may be appropriate for machines, robots and even for trained rats but not for human beings.

It should be made clear at this point that the present research program does not accept this objection. Thanks to evolution human beings have developed ever newer, more complex and more efficient forms of behaviour

regulation. It is not immediately clear why a descriptive procedure appropriate for the rest of the natural world should not apply to the human being. There are no insurmountable barriers between human beings and the rest of the natural world, from which the human being originated and as a part of which he may be seen. The present research program is based on naturalistic principles.

Formally a system can be described in terms of eleven elements

( $T, X, Y, Z, W, \mathbf{X}, \mathbf{Y}, \mathbf{Z}, F, H, g$ )

- $T$  Set of time values
- $X$  Set of inputs
- $Y$  Set of outputs
- $Z$  Set of internal state variables
- $W$  Set of dependent variables
- $\mathbf{X}$  Set of time-dependent input functions  $X \subset X^T$
- $\mathbf{Y}$  Set of time-dependent output functions  $Y \subset Y^T$
- $\mathbf{Z}$  Global state transfer functions  $\mathbf{Z}: T \rightarrow Z$
- $F$  Local state transfer function,  $F: (T \times Z \times X) \rightarrow Z$
- $H$  Algebraic function,  $H: (T \times Z \times X) \rightarrow W$
- $G$  Output function,  $G: (T \times Z \times W \times X) \rightarrow Z$

The behaviour of an agent can be described using the terminology of system theory.

The transfer function  $F$  indicates the way in which the current state  $z(t_n)$  at time  $t_n$  is transformed into the subsequent state  $z(t_{n+1})$  as a result of the input  $x(t_n)$ . Therefore, we have:

$$z(t_{n+1}) = F(t_n, z(t_n), x(t_n)) \quad (\text{eq. 2.1})$$

Usually, the state variables  $z$  are not directly related to observable behaviour. Other variables, known as dependent variables, because they depend on the state variables, are ultimately responsible for an agent's behaviour. The relationship between a state variable  $z$  and a dependent variable  $w$  can be described by an algebraic function  $H$ . Therefore, we have:

$$w(t_{n+1}) = H(z(t_{n+1})) \quad (\text{eq. 2.2})$$

The output function  $G$  determines the manner in which the new internal state of the agent, described by the state variables  $z(t_{n+1})$  and the dependent variables  $w(t_{n+1})$ , is transformed into an externally observable output  $y(t_{n+1})$ .

$$y(t_{n+1}) = G(t_{n+1}, z(t_{n+1}), w(t_{n+1}), x(t_{n+1})) \quad (\text{eq. 2.3})$$

The basic assumption made in PECS is that an agent's personality depends on the form of the functions  $F$  and  $H$ .

The transfer function  $F$  changes the internal state variables of an agent, either as a result of experiencing an

input from the outside world, or of its own accord. The state variable  $z$  could be Anger, for instance. This state variable might be changed by an external input  $x$ , when the agent experiences a personal failure.

$$\text{Anger}(t_{n+1}) = F(\text{Anger}(t_n), \text{Experienced\_failure}(t_n)) \quad (\text{eq. 2.1a})$$

Another example of change in a state variable would be Energy demand. This state variable increases either continuously of its own accord, or changes according to the kind of action the agent performs.

$$\text{Energy}(t_{n+1}) = F(\text{Energy}(t_n), \text{Action\_performed}(t_n)) \quad (\text{eq. 2.1b})$$

The state variable Energy does not directly influence the agent's behaviour. The function  $H$ , which relates Energy to the drive Hunger, acts as a motive. This means, that the state variable Energy is converted into the dependent variable Hunger.

$$\text{Hunger}(t_{n+1}) = H(\text{Energy}(t_{n+1})) \quad (\text{eq. 2.2a})$$

In both examples the agent's behaviour depends on the form of the two functions  $F$  and  $H$ , and in particular on the constants contained within these functions.

As a reference model, PECS offers a pattern or framework containing empty spaces which have to be filled in order to adapt the general reference model to a specific, real task or an actual problem. The specific state variables and the functions  $F$ ,  $H$  and  $G$  are freely definable. By assigning values to the constants in the functions  $F$  and  $H$ , agents can be given individual personalities, which determine how their inner states change. The output function  $G$  depends on these internal state variables and describes how the agents behave.

It is important to emphasise that PECS is almost entirely theory-independent. It is the task of a theory to determine the mathematical form of the functions  $F$  and  $H$ , and which variables should appear as arguments. All possible functions  $F$  or  $H$  proposed by a particular theory can be used in PECS and their consequences investigated.

As a reference model, PECS provides a conceptual framework that can be implemented in arbitrary agents in any simulation language whatsoever.

### 3. DESIRES, MOTIVES AND ACTIONS

In some simple cases, the state variables directly determine the behaviour of an agent. This is particularly the case with reactive behaviour as situations that cause reactive behaviours are in general more complex.

Behaviour is usually dependent on drives, needs or desires which can be regarded as motives. The strength or intensity of these motives is a function of the state variables. In this case the state variables do not determine behaviour directly, but rather indirectly, via the motives belonging to them. This basic idea was adopted

from [Dörner 1999] and generalised to include all four possible classes of motives.

For example, let's examine the state variable Energy. This variable was introduced in section 2 (see eq. 2.1b) and did not influence behaviour directly. The function  $H$  was used to define the drive Hunger. It is the intensity of this drive, which determines whether the agent goes to the refrigerator or whether it does something else.

Similarly to the state transition function  $F$ , the function  $H$  contains constants, which give an agent his characteristic and individual nature.

A PECS agent can be endowed with various drives, needs or desires. The agent experiences these drives, needs or desires as internal forces that motivate him to perform corresponding actions.

Drives, needs and desires can be very diverse. The PECS reference model provides no directions about which ones should be included. PECS simply contains empty spaces into which the user can insert the drives, needs or desires he considers being relevant.

It is possible to arrange the desires in a hierarchical order, as in the humanistic approach of Maslow [Maslow 1954]. It is equally possible to adopt a position where all the desires compete with one another on the same level, as in the approach of Reiss. Reiss assumes 16 different basic desires that motivate our behaviour and define our personality. [Reiss 2000].

Unfortunately, psychology does not offer a clear-cut definition of the concepts of drive, need, urge, desire or motive. In order to explain the following processes more clearly, these concepts are defined arbitrarily. These definitions are not claimed to be generally valid: they apply to this presentation only.

#### 3.1 Intensity of Drives

Drives are related to physical state variables like blood pressure, body temperature and energy. They denote the urge a person experiences or feels in order to satisfy a particular physical need. Drives usually serve to maintain the homeostatic equilibrium of an agent's body in order to support his physiological functioning. Once again, the above-mentioned state variable Energy and the drive Hunger serve as examples.

The body strives to maintain a fixed level of energy. If this level is not achieved, for example it is too low, the body tries to regain the desired state by urging the agent to look for food.

The intensity of this drive is a function of the state variable Energy and can be calculated by means of the function  $H$ . In general, the lower the available Energy the more intense the Hunger drive will be.

$$\text{Hunger}(t_{n+1}) = H(\text{Energy}(t_{n+1})) \quad (\text{eq. 3.1})$$

In the special case of Energy and Hunger, the function  $H$  could have the following form:

$$\text{Hunger}(t_{n+1}) = \text{MaxHunger} * (1 - f(\text{Energy}(t_{n+1}))) \quad (\text{eq. 3.1a})$$

$$f(\text{Energy}(t_{n+1})) = [1 + \exp(-\text{HungerIncrease} * \text{Energy}(t_{n+1}) - \text{HungerMean})]^{-1}$$

If ( $\text{Energy}(t_{n+1}) > \text{EnergyLimit}$ )  
Then  
Hunger = 0

The function  $f(\text{Energy})$  is the so-called Richard's curve, which is frequently used to describe dependencies of this form. [Horgan 2001]

Diagram 3.1 shows the course of the intensity of the Hunger drive depending on the available Energy, according to equation (3.1a). As the available Energy decreases, we see an increase in the Hunger perceived.

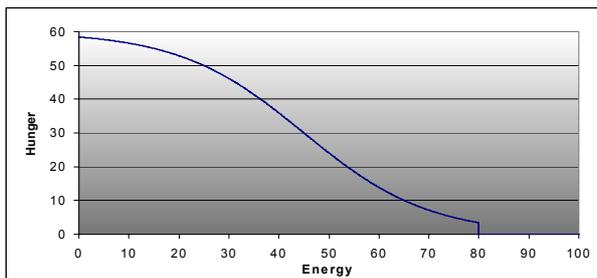


Diagram 3.1 The intensity of the drive Hunger, depending on the available Energy

Clearly there is a distinct difference between the physical state Energy and the perceived Hunger drive. In particular, if the Energy is high, the agent does not experience any drive as long as the Energy stays over the threshold value of the Energy limit.

Another possibility for the intensity of the Hunger drive could be:

$$\text{Hunger}(t_{n+1}) = \text{HungerIncrease} * (\log(\text{EnergyDeficit}(t_{n+1}) + 1)) \quad (\text{eq. 3.1b})$$

If ( $\text{EnergyDeficit}(t_{n+1}) < \text{EnergyMin}$ )  
Then  
Hunger = 0

The equation (3.1b) was taken over from [Dörner 1999].

The exact form of the Hunger drive is determined by the three constants MaxHunger, HungerIncrease and HungerMean in equation (3.1a).

The equations (3.1a) and (3.1b) with their two constants MaxHunger and HungerIncrease determine how intensively the agent really perceives or experiences the internal state Energy. It is essential that it is not the real internal state Energy itself, but only the experienced intensity of the corresponding Hunger drive, which is responsible for the actual form of an agent's action.

It follows that the two constants MaxHunger and HungerIncrease, together with the constants that determine the value of the state variable Energy in equation (3.1b), are the values that make up the personality trait

with respect to the Energy state and the Hunger drive.

The greater the Energy deficit, and the stronger the Hunger drive is, the more vehemently the agent will act. An agent with a personality that is very sensitive to possible Energy deficit and the Hunger drive, will attach great importance to the fulfilment of this particular desire.

### 3.2 Emotional Intensity

In PECS, emotions like anger, fear, surprise or envy are treated as basic state variables. Their change can be described by the state transition function F.

Similar to the relationship that exists between the Energy state and the Hunger drive, there is a relationship between an emotion and the experienced intensity of this emotion. The function H connects the emotion state variable, for example Fear, with an intensity, such as UrgeFear. (UrgeFear is an artificial construct, since colloquial English does not possess a separate word for the intensity of an emotion in comparison with the emotion itself.)

For the intensity with which Fear is perceived, the following equation can be used:

$$\text{UrgeFear}(t_{n+1}) = H(\text{Fear}(t_{n+1})) \quad (\text{eq. 3.2})$$

### 3.3 Willpower

Deliberate behaviour is focussed on a goal. A goal is a situation, which can be described in terms of cognitive state variables. An agent pursues a goal more or less resolutely, according to his willpower. As before, willpower can be calculated using the function H. The cognitive state variables of the goal are used as arguments for H.

As an example, the state variable KnowAct might describe the quantity of knowledge an agent possesses at a particular point in time. One of the agent's goals might be to increase that quantity.

The agent pursues the goal with the strength WillKnowledge. The dependent variable WillKnowledge will usually increase as the value of the corresponding state variable KnowAct gets smaller. The function H will be of the following form:

$$\text{WillKnowledge}(t_{n+1}) = H(\text{KnowAct}(t_{n+1})) \quad (\text{eq. 3.3})$$

In its most simple form, equation 3.3 might look as follows:

$$\text{WillKnowledge}(t_{n+1}) = - \text{WillIncrease} * 1/\text{KnowAct}(t_{n+1}) \quad (\text{eq. 3.3a})$$

Another possibility could be:

$$\text{WillKnowledge}(t_{n+1}) = \exp(-\text{WillIncrease} * 1/\text{KnowAct}(t_{n+1})) \quad (\text{eq. 3.3b})$$

The less the agent knows, and the stronger the will to

change that situation, the more vehemently the agent will act. It could be said that the agent has a personality with a very strong will as far as the acquisition of knowledge is concerned.

### 3.4 Intensity of Social Desire

Social state variables describe facts about the agent in relation to other agents. For instance, the state variable SocAct measures an agent's current social satisfaction. SocAct increases if the agent is in the company of others and decreases if it is by itself. An agent's current social satisfaction shows itself through a corresponding desire for company. The intensity of this desire can be calculated using the function H, thus:

$$\text{DesireCompany}(t_{n+1}) = H(\text{SocAct}(t_{n+1})) \quad (\text{eq. 3.4})$$

The exact form of the function H, in equation (3.4), will depend upon the nature of the problem. For example, it could have a similar form to equations (3.1a) or (3.1b). The function H determines how rapidly an agent feels lonely and how strongly the agent desires to do something about it. Therefore, these constants describe the personality trait, sociability.

### 3.5 Motives and the general procedure

Initially, in the above cases, changes in the state variables are calculated using the transition function F. The transformed internal state may then result in the agent feeling or experiencing an internal urge, which may drive it to perform a particular action.

Drives, emotional intensity, will power and social desire are all called motives. Thus, "motive" is a collective concept comprising four different constructs.

Motives are not static but change continuously over time. Moreover, they compete with one another. The strongest one becomes the action-guiding motive and determines the agent's behaviour.

Since drives, emotional intensity, will power and social desire are all regarded as motives, and since each of these motives has a corresponding intensity, motives can be compared with each other. It is thus possible to establish which motive is the strongest at a given point in time and hence determine the action to be executed.

For example, it is possible for an agent to experience hunger at the same time as following the goal of tidying the house. In addition, it can feel lonely and wants to go out to see friends.

We then have the following scenario:

- 1) Intensity of the drive Hunger  
Drive-controlled behaviour: Go to the fridge
- 2) Intensity of will power  
Will-controlled behaviour: Tidy the room
- 3) Intensity of social desire  
Socially controlled behaviour: Go to a party

At the beginning, the agent's will power may have the highest intensity. That means the agent will start to tidy

the house. However, over time Hunger may become stronger and stronger. At some point the intensity of Hunger will overtake the intensity of the will power. The action of tidying stops and is replaced by going to the fridge.

The three motives are not constant, but change over time. Therefore different motives may be action-determining at different times. Thus, for example, it is possible that initially the intensity of will has the highest value, and so the agent is interrupted. A new motive takes control and the agent goes to the fridge.

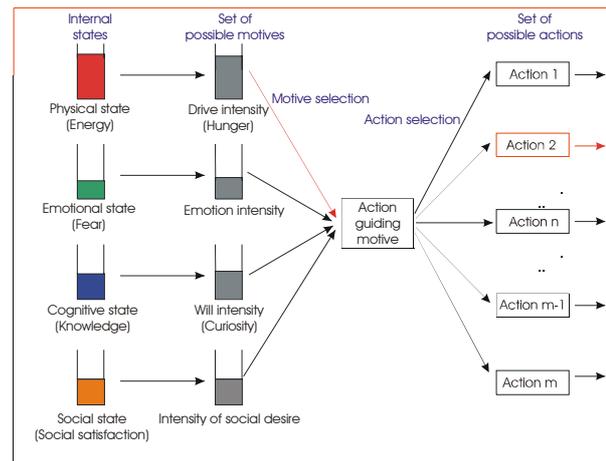


Diagram 3.2 Motives and motive selection

The proposed methodology makes it possible to combine motives as diverse as the intensities of drives, emotion, will power and social desire. Furthermore, the rich and vivid dynamics, which exist within the mind of an agent, can be modelled in a clear and manageable way. Diagram 3.2 shows the competition between the four different kinds of motives.

Under the proposed methodology, the following steps are carried out before an agent undertakes an action:

- 1) Determine the new values of the internal state variables using the state transfer function F.
- 2) Calculate the corresponding intensity of each motive using the function H.
- 3) Compare the various competing motives and select the one with the highest intensity as the action-guiding one.
- 4) Perform the action which is demanded by the action-guided motive.

## 4 THE PAX MODEL

PAX is a model that is used to describe the peaceful operations of the military for the distribution of food care parcels in an occupied war area. The aim is to investigate strategies for the soldiers in a variety of different circumstances. It is obvious that in such a situation, emotional, social, and especially psychosocial aspects have to be taken into account in addition to rational and cognitive considerations.

The model contains the soldiers who distribute the food parcels, the supply vehicle and the civilians. The setting of the scene is as village such as one found in Bosnia, Macedonia or Afghanistan. A general description of the project can be found in [Schwarz 2000].

Diagram 4.1 shows the real environment and its representation as a chessboard in the model. In the model the colours stand for the following:

Black areas: houses

Blue square: the supply vehicle

Blue circles: soldiers

Yellow circles: civilians

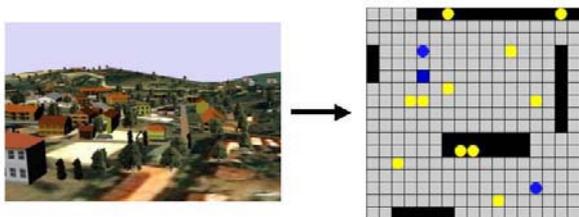


Diagram 4.1 The real system and its model

Each human being is represented by an agent. There agents are constructed internally using the PECS structure.

Via the Sensor component, the civilian-agent realises the actions of a soldier. According to his momentary internal state and his persistent personality the agent's actor component responds with an action.

#### 4.1 Internal states, motives and actions

Each agent is characterised by the following three internal variables:

- Fear
- Anger
- Need for food

Each of these internal variables can change its value for the following three reasons:

- Self-dynamics  
This variable changes its value independently without any external influences. For instance, Fear or Anger decrease with time if nothing happens. Individual psychology provides information on how this happens.
- Actions of the soldiers  
The various actions the soldiers are capable of influence the internal state of an agent. For instance, if a soldier calms down a civilian, the civilian's Anger and Fear will decrease.
- Influence of the group  
The common state of the surrounding group affects the state variables of the agent. For instance, if a peaceful agent enters a hostile, aggressive environment the agent's Anger will increase. Social psychology provides descriptions of this type of situation

Diagram 4.2 shows the threefold way the internal state of an agent can change and as a consequence how actions are induced.

#### 4.2 The change of the state variables

As previously mentioned, the internal state variables of an agent can change in a threefold way. The procedure will be described in more detail using the state variable Anger.

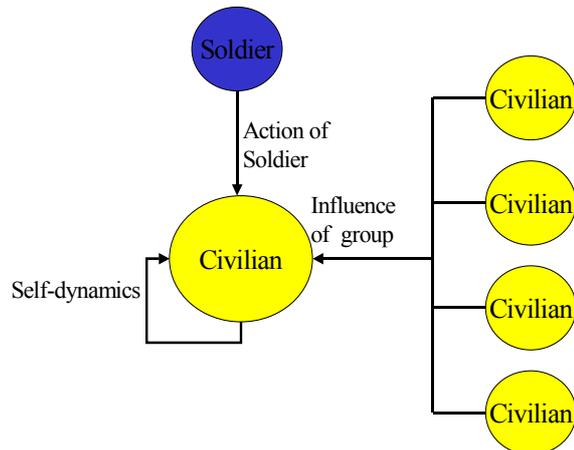


Diagram 4.2 The threefold way to change the internal state of an agent

The overall change of the state variable is specified by means of a differential equation and by means of a time-discrete event. The differential equation 4.1 accounts for the continuous change of the state variables. It has the following form:

$$\text{Anger}' = ((\text{AngerMax} - \text{Anger}) / \text{AngerMax}) * \text{AngerChange} * \text{Anger} \quad (\text{eq. 4.1})$$

AngerChange describes the rate with which the Anger changes over time.

It has two parts:

$$\text{AngerChange} = \text{PersonalAnger} + \text{GroupAnger} \quad (\text{eq. 4.2})$$

PersonalAnger is a variable that describes the self-dynamics with which the variable Anger decreases of its own accord. GroupAnger adds the amount of Anger that is induced by the surrounding group members.

Diagram 4.3 shows the natural decrease of Anger without external influences.

Eq. 4.3 describes the discrete change as a consequence of an action by a soldier, thus:

$$\text{If ActionSoldier} \\ \text{Anger} = \text{Anger} + \text{AngerActionSoldier} \quad (\text{eq. 4.3})$$

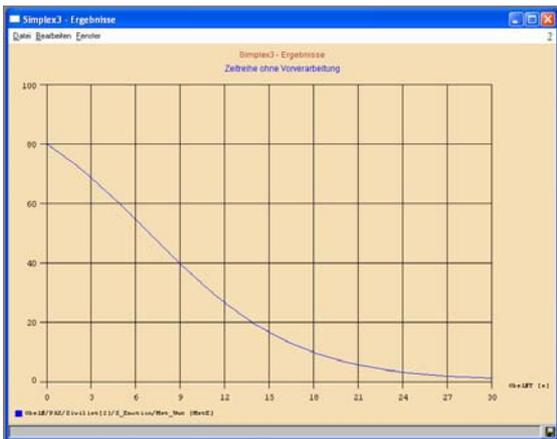


Diagram 4.3 The natural decrease of Anger without external influences

If a threat is present this action leads to a sudden increase in Anger by the amount of AngerAction Soldier, which is added to the already existing amount of Anger. Diagram 4.3 shows the course of the natural decrease in Anger if there are no group influences and no actions of the soldiers are experienced

Diagram 4.4 shows how the state variable Anger changes in a discrete way, when a soldier exerts an action such as a threat. The methodology is the same as the one described by diagram 1.

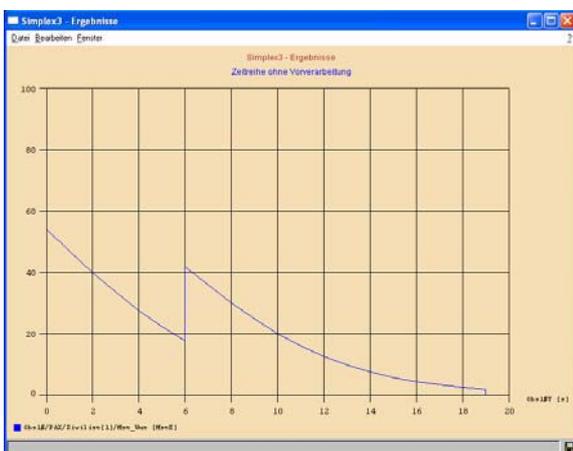


Diagram 4.4 The discrete increase of Anger as a consequence of an action

### 4.3 The repertoire of actions and behaviour

Each agent has a limited repertoire of actions at his disposal.

The possible actions for the soldiers are the following:

- Calm down
- Threat
- Attack

The possible actions for the civilians are:

- Retreat
- Wait

- Attack
  - Queue for a food parcel in front of the food vehicle
- The component Behaviour within the PECS architecture determines - according to a set of rules - which action is selected and finally performed. This selection depends on the strength of the motives and modifying factors.

### 4.4 Results

The goal of the PAX model is to investigate the best possible set of rules for the soldiers under various, diverse conditions. For this purpose the parameters of the model can display a wide range. Modifiable variables are among others:

- The personality traits of the soldiers
- The personality traits of the civilians
- The strategies for the soldiers
- The environment

One interesting investigation attempts to find a robust strategy for the soldiers. It should be applicable and successful in as many different circumstances as possible.

Diagram 4.5 shows the reaction of the civilians as a consequence of six different rule sets or strategies. One sees that strategy number 4 leads to a very low value for both the state variables Fear and Anger. This means that a behaviour of the soldiers that follows these instructions will lead to a peaceful and successful operation.

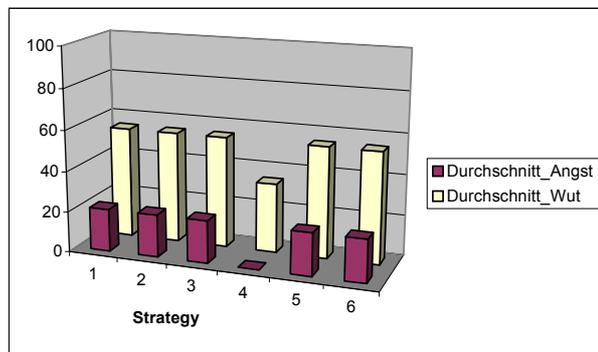


Diagram 4.5 Various strategies for the soldiers and their consequences for civilians

## 5 THE PECS REFERENCE MODEL

A reference model can serve as a blueprint for a class of real systems. It shows the structure of a model for all real systems that have a common deep structure and that differ only in superficial qualities.

A PECS model in this sense is a reference model for the modelling of human behaviour. The architecture proposed here claims to be universally applicable. Adaptation to individual conditions occurs by means of filling in the empty spaces provided by the architecture. This means for example that the number and the type of state variables, the structure of the transfer function F and the development of the output function G can be modified without difficulty. Similarly the agent can be endowed

with a varied repertoire of actions that state the external actions of which the agent is capable. As a result, very diverse agents and agent communities develop but they all have the same deep structure and therefore they can all be described by one and the same reference model. The PECS reference model is based on [Urban 2000a], where a more detailed and wide-ranging description is given.

The agent world of the reference model PECS consists of the following fundamental components:

- the environment component
- the connector component
- the agents

Diagram 5.1 shows the basic structure.

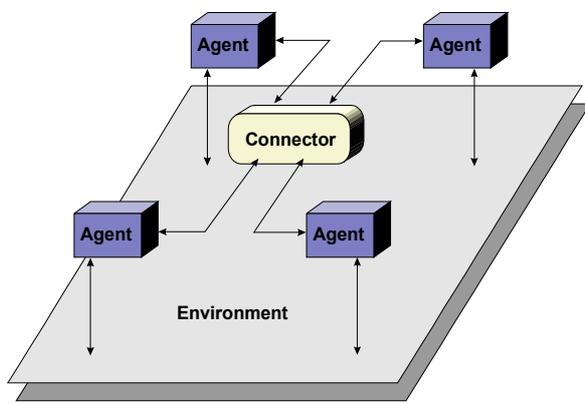


Diagram 5.1 The Structure of the PECS Reference Model

## 6. CONCLUSIONS

It is possible to construct a wide range of models for agents whose dynamics is determined by physical, emotional, cognitive and social factors and their interactions. Especially valuable is the possibility to specify the following three modes of behaviour control:

- Reactive behaviour
- Deliberative behaviour
- Reflective behaviour

It was shown in an exemplary and prototypical fashion the methodology to be followed in the modelling of human behaviour in general.

## BIBLIOGRAPHY

Brailsford S.C. et al. (2002); European Journal of Operations Research; forthcoming  
 Cañamero, D. (1997); Modeling Motivations and Emotions as a Basis for Intelligent Behaviour, in: Johnson, W. L. (ed.), Proceedings of the First International Symposium on Autonomous Agents (Agents '97), The ACM Press, New York, 148-155  
 Damasio, A. R. (1994), Descartes' Error, Emotion Reason and the Human Brain, Grosset/Putnam Books, New York

Davies et al. (2000); Using Simulation Modelling for Evaluating Screening Services for diabetic Retinopathy; Journal of the Operational research Society 51  
 Dörner, D. (1999); Bauplan für eine Seele; Rowohlt Verlag, Reinbeck bei Hamburg  
 Goleman, D. (1995); Emotional Intelligence. Why it can matter more than IQ, Bantam Books, New York  
 Horgan, G. (2001); <http://www.bioss.sari.ac.uk/smart/unix/mgrow/slides/slide02.htm>  
 Klein, R., et al. (1984); Visual Impairment in Diabetes; Ophthalmology 91, 1-9  
 Maslow, A. (1954); Motivation and Personality; Harper and Row, New York  
 Mayer, J. D., Salovey, P. (1990); Emotional Intelligence, in: Imagination, Cognition and Personality, 9, 185-211  
 Mayer, J. D., Salovey, P. (1997); What Is Emotional Intelligence?, in: Emotional development and emotional intelligence, Basic Books, New York, 3-32  
 Moffat, D., Frijda, N. H., P. (1995); Where there's a Will there's an Agent, in: Wooldridge, M. J., Jennings, N. R. (eds.), Intelligent Agents, Lecture Notes in Artificial Intelligence 890, Springer Verlag, Berlin, 245-260  
 Ortony, A., Clore, G. L., Collins, A. (1988); The Cognitive Structure of Emotions, Cambridge University Press, Cambridge, UK  
 Picard, R. (1997); Affective Computing; MIT Press, Cambridge  
 Rao, A.S., et al. (1995); BDI-Agents: From Theory to Practice; in: Proceedings of the First International Conference on Multi-Agents-Systems (ICMAS); San Francisco  
 Reiss, S. (2000); Who am I? The 16 basic desires that motivate our behaviour and define our personality; Penguin Putnam, New York  
 Schmidt, B. (2000); The Modelling of Human Behaviour, SCS-Europe BVBA, Ghent  
 Schmidt, B. (2005); Human Factors in Simulation Models; [http://www.or.uni-passau.de/2/Human\\_Factors.pdf](http://www.or.uni-passau.de/2/Human_Factors.pdf)  
 Schwarz G., et al. (2002); Agent Based Modelling in Military Operations Research: A Case Study; 3rd Workshop on Agent Based Simulation, Passau, 2002; SCS-European Publishing House, Erlangen  
 Sloman, A. (2000); Architectural requirements for human-like agents both natural and artificial. (What sorts of machines can love?), in: Human Cognition and Social Agent Technology, Advances in Consciousness Research, John Benjamins, Amsterdam, 163-195.  
 Urban, C. (2000a); PECS – A Reference Model for the Simulation of Multi-Agent Systems, in: Suleiman, R., Troitzsch, K. G., Gilbert, G. N. (eds.), Tools and Techniques for Social Science Simulation, Physica Verlag, Heidelberg  
 Urban, C. (2000b); PECS – A Reference Model for Human-Like Agents, in: Thalmann, D. (ed.), Deformable Avatars, Kluwer academics publishers, Boston  
 Velásquez, J. (1997); Modeling Emotions and Other Motivations in Synthetic Agents, in: Proceedings of the National conference on Artificial Intelligence (AAAI-97), MIT/AAAI Press

# Cognitive Computing: Principles, Architectures, and Applications

Jerzy W. Rozenblit  
Professor and Head  
Dept. of Electrical and Computer Engineering  
The University of Arizona  
Tucson, Arizona 85721-0104, USA  
Email: [jr@ece.arizona.edu](mailto:jr@ece.arizona.edu)

## KEYWORDS

Cognitive computing, agents, high autonomy systems, simulation-based design

## ABSTRACT

This paper is a summary of the plenary presentation. The objectives of the presentation are threefold: a) to discuss conceptual foundations of cognitive computing, b) to demonstrate their impact on intelligent systems design, and c) to present a brief summary of relevant project experiences. An introduction to knowledge-based and cognitive systems, and the explanation of their origins and principles are given. Then, an agent metaphor is introduced as the basis for design of high autonomy, cognitive architectures. Examples of projects from both industry and research laboratories that leverage from the above concepts are discussed. Some recent work that focuses on decision making in complex, information rich environments, multi-agent gaming models, and implementation of symbolic representation techniques in highly flexible, reusable, object-oriented visualization systems is presented.

## INTRODUCTION AND MOTIVATION

Cognitive computing is an emerging approach that builds upon a wealth of research and development work in Artificial Intelligence (AI). It strives to provide methods to construct and operate systems that “know what they are doing” (Brachman 2002). From a perspective of practicing modelers and systems engineers, the primary motivation behind adopting cognitive methods is to better support the design and deployment of complex, intelligent systems.

It is also the systems’ complexity that motivates us strongly to develop new integration techniques that help achieve high levels of autonomy and intelligence. As modelers and designers, we are excellent at constructing system modules and subcomponents. However, we often falter at the integration of those components not just in a structural, but also in a functional sense. For several years now, the Defense Advanced Research Projects Agency (DARPA) has been driving an effort to build systems that are able to acquire and accumulate knowledge, reason, learn, explain themselves, and be aware of their own behavior (and be robust).

Clearly, these are very highly sophisticated objectives and, realistically, there is currently no artificial system that can exhibit that kind of a complex, integrative behavior.

From an engineering perspective, computer-aided support at the higher design level is urgently needed. Whereas excellent support exists at lower design levels — for instance, in circuit, or VLSI design — support for integrating hardware and software components at higher system levels is poor. Thus, our desire is to develop adequate modeling tools that support the development of complex heterogeneous systems, allow for reuse of models and modules, and help us in rapid prototyping.

In the following sections, we examine how the cognitive techniques could help us accomplish those goals. We begin with a discussion of cognitive systems and their underlying AI paradigms.

## COGNITIVE SYSTEMS

The origins of cognitive systems work lie in cognitive science — a discipline that brings together researchers from the fields of psychology, linguistics, philosophy, computer science, and more recently, neurocomputing. We perceive “cognitive computing” as an approach that has emerged from, and attempts to subsume, the work done in AI. Given the computational power that we now have at our disposal, we are able to explore complex cognitive issues paradigms and supplement the often imprecise methods used in psychology by rigorous modeling. We can implement a lot of theories now in a computational mechanism that allows us to solve these problems computationally, not just necessarily analytically as has been tackled in the past. We could thus say that that cognitive systems are systems that understand, seek to understand how we perceive, how we think, remember, learn, and form models.

If we take a “computational approach”, we can view cognitive systems in an information-processing context. More specifically, we might see them as a kind of input, output, and transition systems. Such systems are well described by an agent metaphor, i.e., a system that perceives its environment, processes information, and takes actions that affect the environment. The classical definition of an agent stipulates that it be an entity

capable of information processing at various levels of sophistication and able to affect the world in which it operates (Russell and Norvig 1995). This general metaphor is depicted in Figure 1.

An agent could be a robotic machine, a segment of software, etc. Several examples are given in Table 1. Agents are typically given specific goals and act in a purposeful manner. The goals drive the behaviors and allow us to generate metrics that assess how good these behaviors are. For instance, in a medical diagnosis system an agent would be perceiving symptoms, findings, and data that are gathered through interviewing the patient. The actions would be more

questions, perhaps a deeper type of investigative technique, medical tests and treatments. The goal here would be a successful treatment outcome, that is a healthy patient with be a normal range of particular test values. The systems shown in the table can all be called agents. The question arises as to what degree of cognitive sophistication they exhibit. While we do not believe that computer programs or artificial systems that “know what they are doing” exist, we could argue that many systems do exhibit “knowledgeable behaviors”. Thus the question that we want to answer is: “How can we tell that intelligence has been achieved or is being exhibited by an artificial system?”

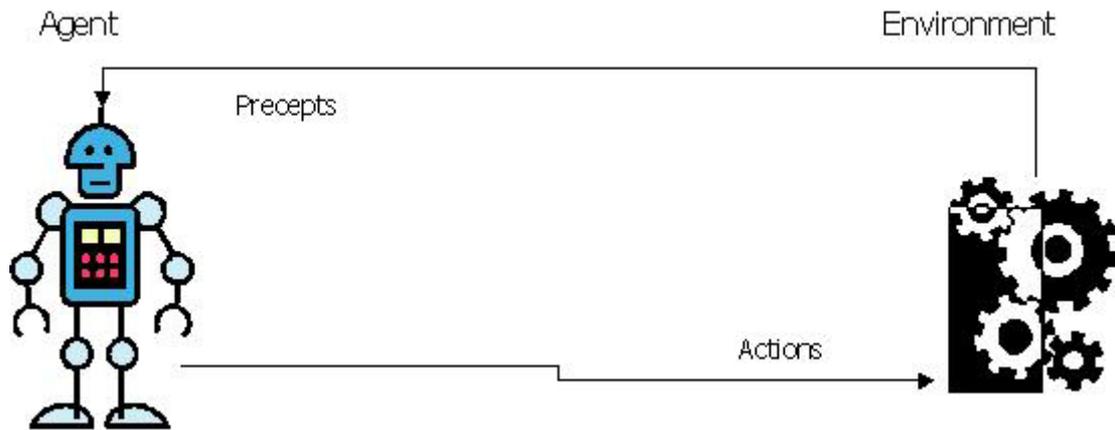


Figure 1: Agent Metaphor

Table 1 Examples of Agent Systems (adopted from “Artificial Intelligence: A Modern Approach”, S. Russell and P. Norvig)

Agent Type	Percepts	Actions	Goals	Environment
Medical Diagnosis System	Symptoms, findings, patient’s answers	Questions, tests, treatments	Healthy patient, minimize cost	Hospital, patient
Satellite image analysis system	Pixels of varying intensity, color	Categorization of scene	Correct categorization	Image processing computers/satellites
Part picking robot	Pixels of varying intensity	Pick parts and sort into bins	Place parts in correct bins	Manufacturing system
Reactor controller	Temperature, pressure readings	Open, close valves, adjust pressure, water temps.	Maximize safety, power	Reactor

## Attributes of Intelligence

To determine if a machine is intelligent, classically the Turing test is carried out in which the machine is called “smart” if its performance cannot be distinguished from that of a human performing a task. The fallacy of this approach is that systems can be programmed that mimic human behavior without actually exhibiting any cognitive skills. (A good example was the Eliza system that emulated behaviors of a psychoanalyst by simply analyzing the syntax of patients’ complaints (Weizenbaum 1966)).

Perhaps a broader test for discerning intelligence would be to ask what are the marks of intelligence. For instance, we might consider the following as representative attributes of intelligence.

We clearly have perception — our desire here is to build agents that are able to perceive. We perceive, we are able to recognize, we are able to classify and abstract certain common properties. We have mental states, in other words, we are thinking about something. We have certain beliefs and we could say that we believe something is true or false. We do learn (and so do animals). Here, we could argue that what clearly distinguishes us from other living beings is the ability to acquire knowledge, ability to improve that knowledge, and the ability to use it to solve new problems; that is something that machines do not do well.

We use language to communicate and disseminate knowledge in a purposeful way. And last but not least, we create models and use them to predict consequences of our actions and to explore our potential choices in an almost limitless way.

AI have so far achieved many of the above marks of intelligence in an isolated form. However, integrating those abilities in an artificial systems is a formidable goal. Ultimately this should be the objective behind the development of innovative cognitive computing architectures that can accomplish not necessarily the level of a created genius, but a level of a highly cognizant intelligent entity.

## Tools for Cognitive Systems Design

A wealth of AI methods and tools exist to assist us in the design of cognitive systems. In the presentation, we will examine in detail a number of approaches. The fundamental areas from which we draw in our practice are state space-based search and problem solving, knowledge representation (KR), rule-, and model-based reasoning, genetic algorithms and co-evolution.

In “the sciences of artificial” where most of the engineering systems are conceived and constructed, the state space approach is a rigorous method that allows us to represent the underlying problems and to solve them

using efficient (often heuristic) methods. Many of the computational problems we face lend themselves to the following paradigm: the system that we build or analyze can be in a finite number of states. Then, the task at hand is to transition from an initial state to the goal state. Thus, solving the problem is to find a trajectory or a sequence of state transitions that would take the system to the goal state(s). This is a powerful paradigm, deeply rooted in the classical control and operations research problems. Many of these problems exhibit combinatorial and exponential behaviors with respect to the number of inputs we work with. AI has been extremely helpful in finding heuristic techniques that allow us to solve search problems efficiently.

Rule-, and model-based reasoning provide a repository of methods that give us introspection into the causes and effects when examining systems’ behaviors. Traditionally, if-then productions (Russell and Norvig 1995) have been employed as a representational mechanism for encoding condition-action (premise-conclusion) pairs in expert and knowledge-based systems. Model-based reasoning allows for a higher level of cognition in which we built a repository of models which represent world states. Using such models (which have dynamic behaviors), we perform various diagnostic, prediction, and control functions (Zeigler 1984, Rozenblit 1992).

We extensively use genetic algorithms (GAs) and co-evolution (Peng et al. 2003, Suantak et al. 2001) as optimization tools that quickly generate suboptimal solutions when the numbers of solution possibilities are very large. GAs mimic the process of natural evolution where the fittest members of a population cross-over their best “genes”, or adapt to environmental changes by mutating some of their gene sequences. GAs are also employed in learning – a process essential to building the agent’s autonomy and its ability to improve how it determines its actions (Russell and Norvig 1995).

In our practical experience, we focus mainly on employing these, and other techniques, to design highly complex systems. Our design philosophy is firmly grounded in the simulation modeling enterprise. The following sections give an overview of our modeling approach, summarize some practical experiences, and propose a highly autonomous model-based system architecture.

## MODEL-BASED DESIGN

In our previous work (Schulz et al. 1998, Rozenblit 2001), we have developed a process that uses stepwise refinement of simulateable models and abstracts system components at multiple levels of representation. In this methodology, a set of requirements and constraints is obtained for the system to be modeled. The system is then described as an abstract model that is a

combination of its structural and associated behavioral specifications.

Given a set of design objectives, requirements and constraints, we first build a simulateable model of the system under design (SUD). Modeling entails the specification of structure (object model) and behavior (dynamics). Object modeling (i.e., model structuring) typically leads to a specification of a structure instance. This is commonly done in a graphical language such as the Unified Modeling Language (UML), which has become a *de facto* tool for object modeling. However, rather than generating a *single* instance of an object model, we advocate the development of a generative object representation that underlies the entire family of possible design configurations for a problem domain at hand. Indeed, UML allows us to capture the multiplicity of design views and taxonomies (specializations) of components through its decomposition and specialization relationships. An enormous variety of decompositions and specializations in large scale systems leads to a combinatorial explosion of design choices. To harness this complexity, procedures are needed that prune out instances of design which best fit design objectives and requirements. Thus, we use heuristic search methods that convert design requirements into selection (for choices from among alternatives offered by taxonomic relationships) and synthesis (for aggregations from among decompositions) into production rules. Then, we search design spaces for best alternatives. The outcome of the search is a set of sub-optimal instances of design object models (Rozenblit and Huang 1991).

The dynamics (behavior) of model components is specified using various modeling formalisms such as the discrete event system specification (DEVS) (Zeigler 1984), finite state machines, Petri nets, etc. The choice of the specification formalism is based on the system's domain. Both the structural and behavioral specifications constitute a virtual representation of the system under design (SUD). This is a "design blueprint" from which a system will be realized. Model components remain implementation and realization (i.e., hardware or software) independent.

We verify correctness of models through computer simulation. A simulation test setup is called an *experimental frame* (Zeigler 1984). It is associated with the system's model during simulation. A frame specifies conditions under which the model of the system is observed. Simulation is then executed according to the run conditions prescribed by the frames. At the end of the simulation process the "best" (polyoptimal) virtual system prototype is obtained. The design is then partitioned into hardware, software and corresponding interfaces using a process that we call *model mapping* (Schulz et al. 1998). We have applied

this framework to design a variety of highly autonomous systems by combining the above simulation modeling principles with the tenets of AI and cognitive systems. Examples are given below.

### Some Practical Experiences

Our laboratory conducts research in systems design and analysis, engineering of complex systems, and software engineering. Detailed principles for designing such systems will be shown including a testing methodology that ensures conformance to project's requirements. In the presentation we will show several instances of complex systems. Examples will include a *unified sensing system* model in which configuration, management, and tracking algorithms are implemented over a wireless, multi-sensor network (Vaidya et al. 2005), and a large scale object-oriented system for decision making in complex, information rich situations (such as military, peacekeeping, or disaster relief operations).

The purpose of this latter work is to provide visualization capabilities to decision makers using advanced computer technology that symbolically abstracts the most important features of the information space. The technology facilitates rapid creation of tailored, low resolution, high semantic content visualizations of complex operations. Recent extensions (Peng et al. 2003) include a hybrid software/hardware that builds on the symbolic, object-oriented visualization software.

### TOWARDS A COGNITIVE, HIGH AUTONOMY ARCHITECTURE

We postulate that simulation modeling could play the key role in designing highly autonomous, cognitive architectures. High autonomy, defined here as the ability to function with little or no intervention from the "operator", is a mark of cognitive sophistication.

The postulated architecture shown in Figure 2 consists of three major elements: a) the executive layer that comprises the planner and simulation, b) the coordination layer that includes the diagnoser, model base, monitor, and executor, and c) the execution layer that acts upon the real world through the effector, and collects observables through the perceptor.

The planner's function is to generate nominal action plans, given a task or mission description and the world states obtained from the models which reside in the model base. The simulator provides model-based

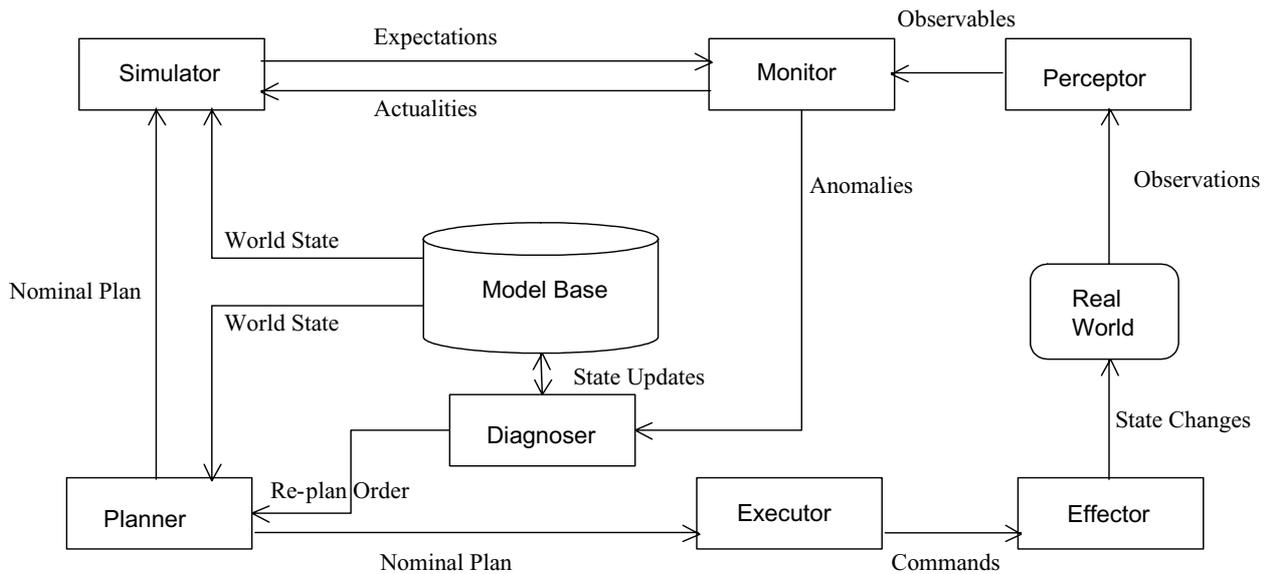


Figure 2: High Autonomy Cognitive Architecture

expectations that are compared with the actual observables in the monitor. Any discrepancies are reported to the diagnoser which, in turn, orders replanning directives.

Increasing levels of autonomy could be defined as: a) the ability of the system to achieve its objectives, b) the ability to adapt to environmental changes, and c) the ability to develop its own objectives. We believe that the model-based approach allows for building such functionality into the architecture presented above (perhaps with the exception of item c.).

## CLOSING REMARKS

The notion of cognitive systems and computing is not new. Well established AI-based methods have existed for several decades. However, to a large extent AI has not delivered an integrative capability to build complex systems that combine many of the intelligent features found in isolation in simpler components. We postulate that a simulation modeling approach to design of highly intelligent, autonomous computing architectures is a powerful tool in accomplishing this integration at both structural and functional levels.

## ACKNOWLEDGMENT

I am grateful to Ms. Rozanne Canizales and Mr. Jeffrey Peng for their assistance in editing this manuscript.

## REFERENCES

Brachman, R.J. 2002. "Systems that know what they're doing". *IEEE Intelligent Systems and Their*

*Applications*, 17(6) 67 – 71.

Cunning, S.J. 2000. "Automating Test Generation for Discrete Event Oriented Real-Time Embedded Systems." PhD Dissertation, The University of Arizona.

Peng, J., Rozenblit, J.W. and L. Suantak. 2003. "A Hybrid Architecture for Visualization and Decision Making in Battlespace Environments", *The Tenth IEEE Conference on Engineering of Computer-Based Systems*, 207-213.

Rozenblit, J.W., 1992. "Design for Autonomy: An Overview", *Applied Artificial Intelligence*, 6(1), 1-18.

Rozenblit, J.W. and Y.M. Huang. 1991. "Rule-Based Generation of Model Structures in Multifaceted Modeling and System Design," *ORSA Journal on Computing*, 3(4), 330-344.

Rozenblit, J.W. 2001. "Systems Design: A Simulation-Based Modeling Framework." In *Discrete Event Modeling and Simulation: A Tapestry of Systems and AI-based Theories and Methodologies*. (Eds. F. Cellier and H. Sarjoughian), Springer Verlag, 107-127.

Russell, S. and P. Norvig. 1995. "Artificial Intelligence: A Modern Approach". Prentice-Hall.

Schulz, S., Rozenblit, J.W., Mrva, M. and K. Buchenrieder. 1998. "Model-Based Codesign." *IEEE Computer*, 32(8), 60-68.

Suantak, L., Momen, F., Rozenblit, J.W., Barnes, M. and T. Fichtl. 2001. "Intelligent Decision Support of Support and Stability Operations (SASO) through Symbolic Visualization." In *Proceedings of the 2001 IEEE International Conference on Systems, Man, and*

*Cybernetics*, 2927-2931.

Vaidya, D and J. Peng, L. Yang, J. W. Rozenblit. 2005. "A Framework for Sensor Management in Wireless and Heterogeneous Sensor Network." In Proc. of the 12th IEEE International Conference and Workshops on the Engineering of Computer-Based Systems (ECBS'05), 155-162.

Weizenbaum, J. 1966. "ELIZA--A Computer Program For the Study of Natural Language Communication Between Man and Machine." *CACM* 9(1), 36-45.

Zeigler, B.P. 1984. "Multifaceted Modeling and Discrete Event Simulation." Academic Press, 1984.

## **BIOGRAPHY**

**JERZY W. ROZENBLIT** is Professor and Head of the Electrical and Computer Engineering at The University of Arizona, Tucson. He holds the PhD and MS degrees in Computer Science from Wayne State University,

Michigan and the MSc in Computer Engineering from the Technical University of Wroclaw, Poland. His research and teaching are in the areas of complex systems design and simulation modeling. His research in design has been supported by the National Science Foundation, Siemens AG, Semiconductor Research Corporation, McDonnell Douglas, and the US Army Research Laboratories.

Dr. Rozenblit serves as Associate Editor of ACM Transactions on Modeling and Computer Simulation, Associate Editor of IEEE Transactions on Systems, Man and Cybernetics, and Executive Board Member of IEEE Technical Committee on Engineering of Computer Based Systems. He was Fulbright Senior Scholar and Visiting Professor at the Institute of Systems Science, Johannes Kepler University, Austria and has held visiting professorship appointments at the Technical University of Munich, Central Research Laboratories of Siemens AG, and Infineon Technologies AG, in Munich. His e-mail is <jr@ece.arizona.edu>.

# **Invited Paper**



# EXPERIENCE IN THE DEVELOPMENT AND APPLICATION OF SIMULATION IN RUSSIA: REVIEW, ANALYSIS OF PROSPECTS

Stanislav A. Vlasov  
Department of information  
technologies and computation  
systems  
Russian Academy of Sciences  
32a, Leninsky prospect, Moscow  
119991 Russia  
E-mail: [savlas@yandex.ru](mailto:savlas@yandex.ru)

Vladimir V. Deviatkov  
Elina-Computer, Ltd.  
2, Yapeeva st., Apt. 25, Kazan 420111  
Russia  
E-mail: [vladimir@elina-computer.ru](mailto:vladimir@elina-computer.ru)

## KEYWORDS

Simulation, application, GPSS, AnyLogic.

## ABSTRACT

The paper deals with the historic aspects of simulation development in USSR. It shows that due to the improvement of economic situation in Russia and CIS countries interest in simulation started to resume. A brief review of the market of simulation tools used now in Russia is made. The necessity of establishing the Russian Simulation Society is emphasized. The paper includes a list of activities aimed to make simulation one of the most popular technologies in Russia.

## HISTORY OF SIMULATION DEVELOPMENT IN USSR AND RUSSIA

The basics of research and developments in simulation and its application in Russia (USSR) were laid in the 60s-70s. Simulation was one of the most developed aspects in the study of complex systems (systems analysis and study of operations). The achievements in simulation of scientific schools of that time in Moscow, Kiev, Novosibirsk and other cities are known all over the world.

Besides fundamental research, we have a number of important practical results. For example, such famous simulation systems were developed, as SLENG and NEDIS (Institute of cybernetics of Academy of Sciences of USSR), STAM (Bauman Moscow Higher Technical school), MODEL 6 (Computation Centre of Novosibirsk Academictown). Also various special-purpose systems were developed.

Orientation for the creation and use of home software and hardware was the reason why most of simulation means used at that time in the world were not widely used in USSR. The only exception were GPSS systems, which were rather popular. This was due to a number of circumstances:

- translation into Russian of application packages PMDS (GPSS/360) and PMDS 2.0 (GPSS V) and deliveries of mainframe series with these packages to Kazan Computer plant;

- issue of 10 thousand copies of Red book by T. Schreiber "Simulation using GPSS" at "Mashinostroenie" publishing house;

- intensive cooperation within CMEA with the experts from GDR and deliveries of SIMDIS (GPSS/360) and SIMDIS 2.0 (GPSS V) systems to USSR.

There were also single applications of SIMULA and GASP languages, which were developed jointly with experts of Czechoslovakia, Hungary and other countries.

Thus, due to the synthesis of its own scientific ideas, original developments and the best foreign technologies the Soviet school of simulation was recognized worldwide.

## CURRENT STATE OF SIMULATION IN RUSSIA AND CIS COUNTRIES

Unfortunately, after the disintegration of USSR and as a result of the crisis in industry and science, most of this potential was lost. Scientific contacts were broken, financing became unstable. But most important was that in the 90s practical application of simulation methods and means completely disappeared or was replaced by foreign firms, which delivered closed loop turnkey systems.

But gradually, when the economic situation in Russia and CIS countries started to improve, *interest in simulation began to rise*.

Now, we can say that simulation crisis in Russia is coming to its end. These are the main facts confirming our conclusion:

Firstly, activity of simulation experts in the Internet rose steeply.

A number of fundamental Internet-resources about simulation appeared: [www.xjtek.ru](http://www.xjtek.ru), [www.gpss.ru](http://www.gpss.ru), [www.simulation.org.ua](http://www.simulation.org.ua), [www.gpss-forum.narod.ru](http://www.gpss-forum.narod.ru). Statistics of [www.gpss.ru](http://www.gpss.ru) portal is a visual proof of growth of interest in simulation. It exists more than 3 years and every year the number of total visits increases more than thrice as much. At present more than 300 experts visit the portal every day. The geography of visits is either wide: more than 30 countries of the world.

Secondly, academic application of simulation is also constantly growing.

By the most conservative estimate, about 200 higher schools of Russia graduate annually more than 10000 experts who know the basics of simulation research.

Thirdly, a large number of works on this subject have been published.

These include not only simulation symposia and conferences proceedings. Many respectable journals started to publish such works (“Exponenta Pro”, “Computerra”, “Automation in industry”, “Automation and telemechanics”, “Information technologies and computation systems”, etc.).

A number of books about simulation have been published. They include four books, with 3 thousand copies each about GPSS World.

A landmark is the issue of the translation of one of fundamental books on simulation - “Simulation Modelling and Analysis”, third edition, Averill M. Law, W. David Kelton.

Fourthly, companies working professionally with simulation appear again.

There are a lot of them: XJ-Technologies (St Petersburg, [www.xjtek.ru](http://www.xjtek.ru)), B-Club Engineering (Ivanovo, [www.b-club.ru](http://www.b-club.ru)), “Elina-Computer” (Kazan, [www.elina-computer](http://www.elina-computer)), Department of simulation of IBS company (Moscow, [www.ibs.ru](http://www.ibs.ru)), Lengipromez (St Petersburg). New research and developments are carried out in organizations of the Russian Academy of Sciences (RAS): Computation Centre of RAS, Institute of Control Sciences of RAS, Institute of System Analysis of RAS, Institute for Information Transmission Problems of RAS, Institute of Informatics Problems of RAS, St Petersburg Institute for Informatics and Automation of RAS, institutes of Siberian Branch of RAS, etc.

Fifthly, a number of Russian developments have appeared which are highly competitive in the world market.

These developments include: AnyLogic by XJ-Technologies, Fantomat by IBS, Object GPSS by Korolev A.G. (Severodonetsk, Ukraine), etc.

Sixthly, practical interest in simulation appeared in the real sector of the economy.

Important simulation-based projects have been carried out at a number of enterprises.

For example, at Nizhni Tagil Integrated Iron-and-Steel Works a simulation model is used to control productive capacity of the main workshops and process units. At Magnitogorsk Integrated Iron-and-Steel Works simulation is used in the enterprise system resources control subsystem. At “Severstal-Group” metallurgical enterprise simulation is used to evaluate the ways of reconstruction of the main workshops and their logistic flows. Simulation models, developed by Institute of Control Sciences of RAS, were used for the analysis and

implementation of projects of reconstruction of several foreign metallurgical works, built jointly with USSR and Russia.

Simulation models, integrated into production program optimization system (RPMS), developed by Petrocom JV, are widely used at most of petroleum refineries of Russia and at some foreign petroleum refineries.

At present, simulation is widely used to evaluate the quality of information control and transmission in general-purpose geographically-distributed systems.

## **OTHER ASPECTS OF SIMULATION APPLICATION IN RUSSIA**

Taking into account the quick growth of Russia’s economy, gradual transition from mining to the high-level processing of extractives, the started modernization of industry and attempts of enterprises to increase their compatibility, it is natural to assume, that the use of such effective research tool as simulation will increase.

Facts show us, that interest in simulation methods appear now in many spheres.

### **Academic (educational) application**

In spite of the unfavourable situation of previous years and due to enthusiasm and patience of experts of many higher schools, the backbone of highly qualified simulation experts, mainly the teachers of technical universities, was preserved. Simulation is widely used now in training.

The following courses were introduced into the standards for some specialities in Russia: “Modelling of systems”, “Simulation”, “Computer modelling”.

The same situation is in the higher schools of Ukraine, Byelorussia, Kazakhstan and other republics of former USSR. During last years a course of modelling was also introduced in the university syllabus for students of economics. The fact that higher school is the most active participant of discussing and solving simulation problems is proved by the statistics of [www.gpss.ru](http://www.gpss.ru) portal. The rises and falls in the number of visits coincide fully with terms of semesters and holidays.

There is a long list of university experts, who have made a lot in simulation in recent years:

- Professor S.A. Yakovlev (St Petersburg Electrotechnical University “LETI”), Professor Yu.I. Ryzhikov (Mozhaisky Academy), Professor A.G. Varzhapetyan (St Petersburg State University of Aerospace Instrument-Making), Assistant Professor B.K. Yeltyshev (State Marine Technical University), etc. in St Petersburg;
- Professor V.M. Chyornenky (Bauman Moscow State Technical University), Professor Ye.M. Kudryavtsev (Moscow State Civil Engineering University), Assistant Professor N.N. Lychkina (State University of Management), etc. in Moscow;
- Professor S.A. Rodionov (Siberian State University of Telecommunications and Informatics), Assistant Professor V.V. Okolnishnikov (Novosibirsk State

University), Professor Ye.B. Tsoi (Novosibirsk State Technical University), etc. in Novosibirsk;  
 - Professor V.N. Tomashevsky (National Technical University of Ukraine “KPI”) in Kiev.

This list could be continued. We established and maintain contact with more than 250 university teachers. Many universities are changing their program-technical base and buy modern licence software. Thus we hope that the standard of teaching simulation will improve even more.

### Home developments

During the last decade a number of companies appeared in Russia with simulation as the main direction of their activities. These companies have many highly qualified simulation experts. On the basis of their research and work a number of simulation tools have been created, which are now widespread and generally recognized.

Among them are: simulation platform Fantomat (IBS, Moscow), professional simulation tool AnyLogic (XJ-Technologies, St Petersburg), etc.

We would like to emphasize the achievements of XJ-Technologies. In 13 years it became a company known to all in the simulation world. Their system AnyLogic is used at many world-known companies, such as Boeing Company, Hewlett-Packard, General Motors, etc. They constantly take part in many conferences and exhibitions (including WSC) and have their representatives in many countries of Europe and USA.

Table 1 gives brief information about the main simulation tools, developed in Russia and CIS countries.

Table 1: Simulation Tools, Developed in Russia

No.	Name	Manufacturer
1	Fantomat Simulation platform	IBS Department of simulation systems (9-b, Dmitrovskoe shosse, Moscow 127434, <a href="http://www.ibs.ru">www.ibs.ru</a> )
2	AnyLogic Professional simulation tool	XJ-Technologies (21, Politekhnicheskaya st., St Petersburg 194021, <a href="http://www.xjek.ru">www.xjek.ru</a> )
3	ISS2000 Interactive simulation system	National Technical University “Kiev Politechnical Institute”, Kiev, Ukraine (V.N. Tomashevsky)
4	Distributed simulation system for local network in QNX environment (OS UNIX)	Institute of Computational Mathematics and Mathematical Geophysics (Computing Centre), 6, Akademika Lavrentjeva prospect, Novosibirsk 630090, <a href="http://www.ssc.ru">http://www.ssc.ru</a> ,

5	“Queuing system”	Tomsk Polytechnic University (B.G. Oslin)
6	Object GPSS General-purpose simulation system	Severodonetsk Technological Institute, Severodonetsk, Ukraine (A.G. Korolyov)

### Use, distribution and support of world-known simulation tools

The present stage of application of simulation in Russia is characterized by the use of advanced simulation tools, which are used worldwide. It’s an objective process, and it allows our researchers to choose the simulation tool, which is most suitable for them, taking in account their financial situation. Let us divide all simulation tools into two groups – general-purpose and commercial simulators.

Table 2 gives information about support of some general-purpose systems in Russia.

Table 2: General-purpose simulation systems in Russia

No.	Name	Manufacturer	Use and representatives in Russia
1	Extend	Imagine That, Inc. 6830 Via Del Oro, Suite. 230 San Jose, CA 95119 USA <a href="http://www.imaginethatinc.com">www.imaginethatinc.com</a>	In use, no official representatives
2	Automod	5245 Yeager Road Salt Lake City UT 84116-2877 USA <a href="http://www.automod.com">www.automod.com</a>	In use, no official representatives
3	Process Model	ProModel Solutions 556 East Technology Ave. Orem, UT 84097 <a href="http://www.promodel.com">www.promodel.com</a>	B-Club Engineering, Ltd. (Ivanovo) <a href="http://www.b-club.ru">www.b-club.ru</a>
4	Arena	Systems Modeling Corp. 504 Beaver St. Sewickley, PA 15143	Interface Ltd. Moscow, Russia <a href="http://www.interface.ru">www.interface.ru</a>

5	GPSS/H	Wolverine Software Corporation 3131 Mount Vernon Avenue Alexandria, VA <a href="http://www.wolverinesoftware.com">www.wolverinesoftware.com</a>	In use. Consultant: Professor G.A. Varzhapyan St Petersburg State University of Aerospace Instrument-Making <a href="mailto:bnm@aanet.ru">bnm@aanet.ru</a>
6	SLX	Wolverine Software Corporation 3131 Mount Vernon Avenue Alexandria, VA <a href="http://www.wolverinesoftware.com">www.wolverinesoftware.com</a>	In use. Consultant: Professor G.A. Varzhapyan St Petersburg State University of Aerospace Instrument-Making <a href="mailto:bnm@aanet.ru">bnm@aanet.ru</a>
7	GPSS World	Minuteman Software Corporation P.O Box 131 Holly Springs NC 27540-0131 USA <a href="http://www.minutemansoftware.com">www.minutemansoftware.com</a>	P.O.B. 577, Kazan 420111 Russia <a href="http://www.elina-computer.ru">www.elina-computer.ru</a> <a href="http://www.gpss.ru">www.gpss.ru</a>

Certainly, the use and quality of support of the above mentioned products is different. In some cases their use is limited to student versions and single commercial applications, as there are no Russian companies that would localize the product and support it.

Arena and GPSS World are the most widely used simulation systems in Russia. They have documentation translated into good Russian, guarantee and technical support. Dozens of enterprises bought licenses for these systems, and they were used in many simulation studies. Besides general-purpose systems, commercial simulators are also used in Russia. Table 3 presents a list of commercial simulators, known to be used in Russia.

Table 3: Commercial simulators in Russia

No.	Name	Manufacturer	Representatives in Russia
1	eM-Plant System of simulation, optimization and visualization of systems and business-processes	TECNOMATIX GmbH Israel <a href="http://www.tecnomatix.com">www.tecnomatix.com</a>	TECNOMATIX GmbH, Moscow Bureau 6-3-56, Vostrukhina st. Moscow 109542, Russian Federation <a href="mailto:vt.tecnomatix@gtel.ru">vt.tecnomatix@gtel.ru</a> <a href="http://23.relcom.ru">23.relcom.ru</a>

2	Catia Solution Program simulation subsystem	Dassault Systems <a href="http://www.catia.com">www.catia.com</a>	Bee Pitron, Ltd. 4, Vilenskiy pereulok St.-Petersburg 191014, Russia <a href="http://www.bee-pitron.com">www.bee-pitron.com</a>
3	NETRAC System of development and research of communications networks	TTI TELECOM M <a href="http://www.tti-telecom.com">www.tti-telecom.com</a> Israel	“Prioritet” Research-and-production centre 15, Sadovnicheskaya st. Moscow 113035 <a href="http://www.priortelecom.ru">www.priortelecom.ru</a>

As a rule, commercial simulators are designed for various branches of industry (eM-Plant – machine-building, Catya – shipbuilding, NETRAC – communications and telecommunications). When using these simulators the user’s attention is not distracted by the peculiarities of this or that simulation system, but he concentrates fully on the application domain. Using opportunities of visual simulation, and modern dialog and animation technologies he can substantially speed up the research process. Commercial simulators are known to be used in Russia at metallurgical enterprises (Severstal-Group JSC), in motor-car construction (Avtovaz, JSC), etc. The main factor, limiting distribution of these systems is their price. For the most part, Russian enterprises cannot pay \$50000 and more for simulation system, though it is usual practice for western companies, which often buy even several different simulation systems.

#### Establishing interaction between simulation experts in Russia

It is well-known, that joint efforts yield greater results. That is why during last three years intense consultation has been carried out and work has been started on establishing Russian Simulation Society. Most likely, that registration of the Society and its practical work will start this year already.

Before that, integration of experts and their communication was done mainly virtually (via the Internet) and at various conferences. The first Internet-resource designed to integrate simulation experts and give them special-purpose information was [www.gpss.ru](http://www.gpss.ru) portal. Due to this portal we managed to start a dialogue between experts, exchange information, publish several books on simulation and discuss issues of uniting experts in a society. And the most important result was the first all-Russian conference IMMOD-2003, held in 2003. More than 200 experts from Russia, CIS countries and Germany took part in the conference. There were many responses to the conference and its

proceedings are the most asked content at [www.gpss.ru](http://www.gpss.ru) portal. The second conference IMMOD-2005 will be held in St Petersburg on 19-21 October 2005.

### Prospects in the foreseeable future

It is obvious, that the potential market for simulation tools in Russia is rather large. But it will take a lot of efforts and money to develop it. The experts are eager to solve these problems, and growth of economy and interest of enterprises in simulation products make the expansion of the market possible. At present, according to the information we have, less than 10% of Russian enterprises use simulation-based research. Thus, one of the most powerful and promising research tools is not used at its full capacity. In view of the fast growth of competition between enterprises, the coming entry of Russia into WTO, the role of research and tools (including simulation) that can allow to reduce production costs and provide competitive advantages, increases.

To make simulation an up-to-date and widely used technology in Russia, the following work should be done:

- finish creation of the Russian Simulation Society and popularize and promote simulation ideas in the society, especially in the real economy and industry;
- improve the quality of teaching simulation by retraining the teachers and improving the material and technical basis of university simulation courses;
- create a working simulation services and means market; give the Russian experts the most complete range of simulation systems, available in the world, provide their distribution and support;
- help home companies, working in this market, to search for the ways of their development.

In conclusion, we can say that simulation in Russia is coming out of the crisis and we are expecting soon important achievements in the application of simulation in different spheres of economy, industry, science and technology.

### CONCLUSIONS

In conclusion, we can say that simulation in Russia is coming out of the crisis. There are important changes in academic sphere: investigation methods are being developed, institutes buy modern licence software, the number of courses, where simulation is used, is increasing. We are expecting soon the appearance in Russia and CIS countries of new theoretical and application developments, as well as the wider introduction of simulation into different spheres of economy, industry, science and technology.

### REFERENCES

- Ryzhikov Yu.I. 2004. "Simulation. Theory and Technology." Altex-A, Moscow; KORONA print, St Petersburg.
- Smirnov V.S.; Vlasov S.A.; Vaulinsky Ye.S.; and B.I. Lebedev. 2001 "Methods and Models of project management in metallurgy". SINTEG, Moscow.

Tomashevsky V. and Ye. Zhdanova. 2003. "Simulation in GPSS environment". Bestseller, Moscow.

### AUTHOR BIOGRAPHIES



**STANISLAV A. VLASOV** was born in Irkutsk, Russia. He entered Moscow Technological University of Light Industry where he studied automation of engineering processes and obtained his degree in 1964.

In 1967-71 he worked at Bardin Central Research Institute of Ferrous Metals. Since 1971 he has worked at the Institute of Control Sciences of RAS. In 1973-74 he studied at engineering department of computational mathematics and cybernetics faculty of Moscow State University. In 1978, at the Institute of Control Sciences of RAS, he defended Ph.D. thesis on the development and application of simulation methods for design of automated processing integrated iron-and-steel works. His email address is: [savlas@yandex.ru](mailto:savlas@yandex.ru).



**VLADIMIR V. DEVIATKOV** was born in Vologodskaya oblast in 1953. In 1976 he graduated from Kazan Aviation Institute. In 1985 he defended Ph.D. thesis on the development of research methods of

interactive simulation system at Bauman Moscow Higher Technical School. In 1976-1992 he worked at Kazan Mainframe Plant and developed simulation software tools. He is the author and developer of PMDS, PMDS 2.0 and DISMA packages. In 1994 he established Elina-Computer Ltd. and became its director. He has published a number of works. At present he manages the localization of GPSS World. His email address is: [vladimir@elina-computer.ru](mailto:vladimir@elina-computer.ru).



# **Tutorial Papers**



# HOW TO SOLVE THE PUZZLE? SIMULATION SUPPORT FOR COMPONENT-BASED PROCESS DESIGN IN LOGISTICS

Gaby Neumann

Logistics Knowledge Management Group  
Otto-von-Guericke-University of Magdeburg  
P.O. Box 4120, 39016 Magdeburg, Germany  
E-mail: [gaby.neumann@mb.uni-magdeburg.de](mailto:gaby.neumann@mb.uni-magdeburg.de)

## KEYWORDS

Logistics, Component-Based Modelling, Discrete Event Simulation.

## ABSTRACT

Logistics process design is one of the major challenges in logistics planning and known as being a complex and complicated problem, since it is subject to a large number of many and diverse influences. The approach of a component-based logistics process design should help to reduce complexity of the design problem and its solution, but especially to overcome the traditional thinking in terms of technical systems instead of focusing on what kind of functionality is really needed to be implemented only. The paper delivers the basic idea of this concept and discusses how simulation and visualization can support this method even though available tools and packages are based on representations of technical components only.

## THE PROBLEM

Logistics planning aims at the planning of logistics processes and systems in general and at both levels, the strategic one and the operational one. In the context of this paper the term logistics planning is strongly related to the strategic design of logistics processes and systems in preparation of investments for building or modifying logistics services. Within this context it must not be understood as operational planning of logistics processes for running them efficiently and effectively. Furthermore, special attention is paid to the design of materials flows instead of the flows of information, money or energy. With this, logistics shouldn't be reduced to materials flow design, management and control only, but this special aspect of logistics is used as an example to figure out main ideas, discuss basic problems and present approaches that seem to be applicable to logistics in its entirety.

The planning of materials flows can be defined as process of purposeful design and needs-related development of materials flow solutions in their entire complexity of materials flow system and the operational processes carried out by it (Neumann 2001). This

complicated structure of potential solutions causes a high degree of complexity not only to the problems to be solved but also to the planning processes for solving those problems. As shown in figure 1 the latter can be characterised as a phased process of loops, as a process developing variants and versions concerning both the materials flow process and the materials flow system. There are analysing steps and creative, evaluating steps for synthesis by turns with partially changing cognitive problems and views. As a rule searching for appropriate and suitable components for solving the problem comes first. After that the defined components must be made consistent with each other to realise the overall functionality as given in the problem specification.

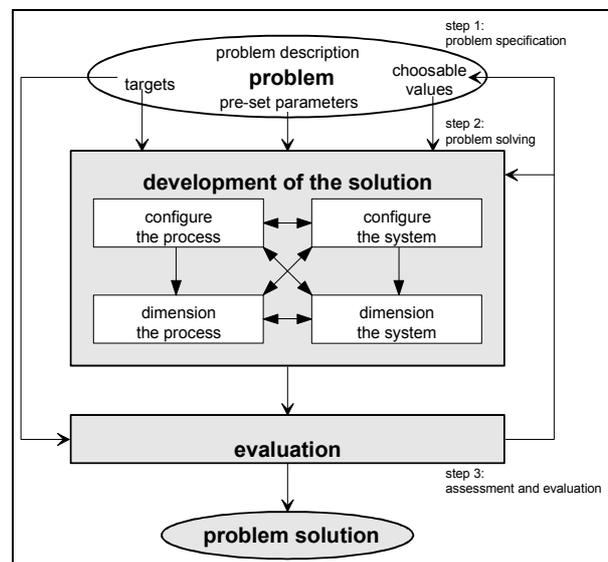


Figure 1: Complex problem solving for planning materials flows

Despite of this holistic approach to materials flow planning in principle, often and in particular planning processes system design dominates the planning process, although process design defines required system functionality any system design should be based upon. As a result materials flow systems use to offer much more functionality than it is needed to fulfil object transformation as required. This too much of system functionality afterwards needs to be reduced by

sophisticated control algorithms requiring a certain amount of control technology and equipment. In the end this leads to a high degree of system complexity and increased implementation costs. Furthermore, this kind of thinking in systems instead of processes reduces very much the chances for real creativity and visionary solutions. To overcome this, a component-based process design is proposed to reduce problem complexity and efforts in the planning process, but also to enable smart materials flow processes and cost-effective materials flow systems behind them.

### THE CONCEPT

The idea of re-usable components for creating new complex solutions in materials flow planning is already well known: CAD packages usually contain catalogues with standard parts such as screws or even more complex technical components. High level simulators defined by Law and Kelton (2000) as “a computer package that allows one to simulate a system contained in a specific class of systems with little or no programming” enable model building from pre-defined sets of components. “The particular system of interest (in the domain of the package) is typically selected for simulation by the use of menus and graphics, without need for programming.” For modelling logistics processes components representing materials flow or information flow operations are used to build up process chains along which objects, i.e. goods or information, are purposefully transformed from a given initial state into the desired final state. The traditional methodology for structuring materials flow processes as described and used in literature (Bolz and Hagemann 1958, Woodley 1964, VDI 3300, VDI 3596) is based upon four categories of processes:

- Operation
- Transportation
- Delay
- Storage

Each of these transformation processes can be characterized by the changes an object experiences with regard to its appearance in space and time. Quantitative characteristics resulting from this are type or sort, quantity, location and time. But this set of characteristics allows to represent selected aspects of object transformation only. For solving problems in logistics process design an extended set of characteristics is required also representing aspects like state of information or quality (Ziems and Neumann 1999). From this, a variety of building blocks for processes, i.e. operations, can be derived (see figure 2):

- Operations modifying an objects appearance according to space, time or structure
- Operations modifying an objects relationship with neighbour objects within the same flow
- Operations effecting several flows

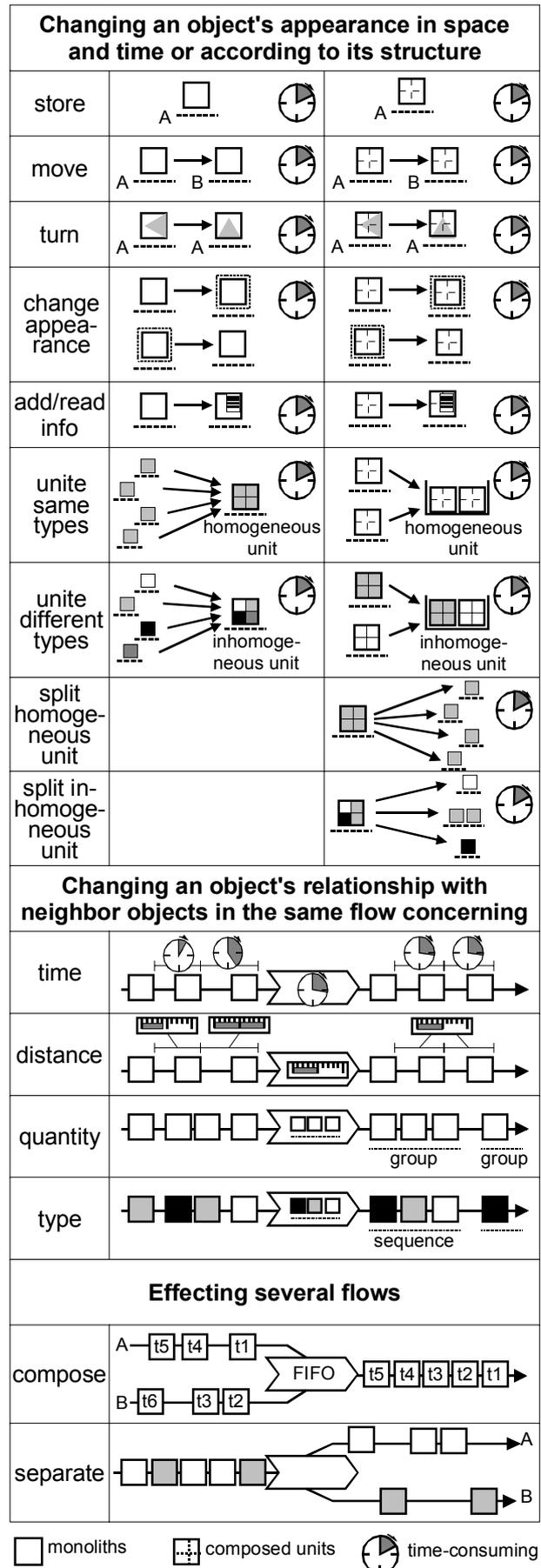


Figure 2: Elementary materials flow operations

These classes of operations are of fundamental nature; by use of them changes with individual goods and units as well as one or more materials flows can be described as transformations of objects and process chains in state models. For defining those elementary operations similarities (and differences) relevant to the design of logistics processes are used to structure the wide variety of phenomenon and to model them along the same lines.

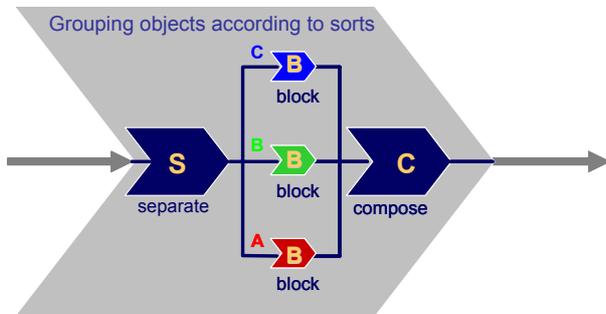
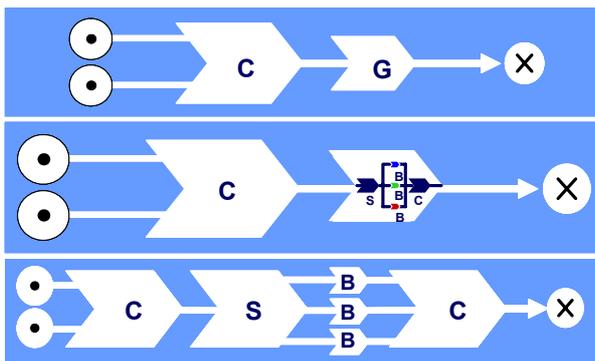


Figure 3: Model building using elementary operations

Elementary operations are the smallest pieces in the puzzle of a logistics process. The entire process can be configured sequencing them alongside the timeline. But, especially in the case of a complex process, it is also possible and might be useful to identify a kind of standard sub-processes of a certain level of complexity (e. g. grouping objects according to sorts) and to model them in detail using elementary operations (see figure 3), but to represent them in the overall process chain by just one component (see figure 4). As a result equivalent models of varying levels of detail of the same object transformation can be built to meet the particular needs of a certain step in the design or analysis process.



C: composing flows G: grouping objects S: separating flows B: blocking objects

Figure 4: Varying levels of detail in the process chain model

## THE METHOD

The approach for component-based process design in logistics is based upon the concept as described in the previous section. In a kind of top-down design a process of complex nature is configured and further specified (see figure 4). In the beginning, building blocks of more complex nature are used to create a first principle

version of the process eventually taking basic variations into consideration. In a second step components representing sub-processes are specified using elementary operations which finally replace the complex component of the first approach. As a result the process chain consisting of elementary operations only represents the minimum functionality required of a system to transform objects as needed. According to this early stage of a planning process the functional specification of the materials flow system is based upon technological and eventually geometric requirements of object transformation only, whereas performance requirements requesting for additional operations such as storing or buffering have not been included yet. To decide about additional functionality eventually to be provided by the system, the process chain as developed so far needs to be analysed using further methods like simulation or visualisation which requires further models of specific types. This way up to four different kinds of process models are built and used in the course of the materials flow design process. For a small illustrating example figure 5 shows these different models:

The *process chain model* (see figure 5b) describes in a standardized way an object's life cycle as sequence of operations in the course of which object characteristics are purposefully changed. A superposition of process chains of different objects forms an abstract model of a company's logistics processes which is used for documentation, visualization, analyzing or planning purposes.

The *principle animation model* (see figure 5c) transfers these static process elements into dynamic operations. These operations are of elementary nature, i.e. they aim at transforming the object with respect to one particular characteristic only. Due to the fact that all of these operations are time-consuming ones an object's time characteristic is changed as a side effect, too. The animation model finally consists of a sequence of small animations each of them representing one operation to transform an object from an initial state starting from the one given with the source into a final state required by the sink. Obviously, the state an object reaches through an operation is also its initial state for the following operation. This equivalence allows to check the sequencing of operations for logical correctness with regard to object transformation. The purpose of using a principle animation model for this is to give visualized support in evaluating the process chain. Consequently there is no possibility for defining or modifying any parameters and this model is to be seen as a qualitative representation of the process chain.

The *simulation model* (see figure 5d) forms the dynamic representation of the process chain modelling not just principal functionality but quantitative parameters as well. With this it allows to investigate flows on the

basis of types of objects and quantities and to analyze flows according to process performance measures like throughput, intensity, or average arrival time and distribution. As a result not only the process' way of working can be proven correct, but also the need for additional buffering or storing functionality and their capacities can be derived.

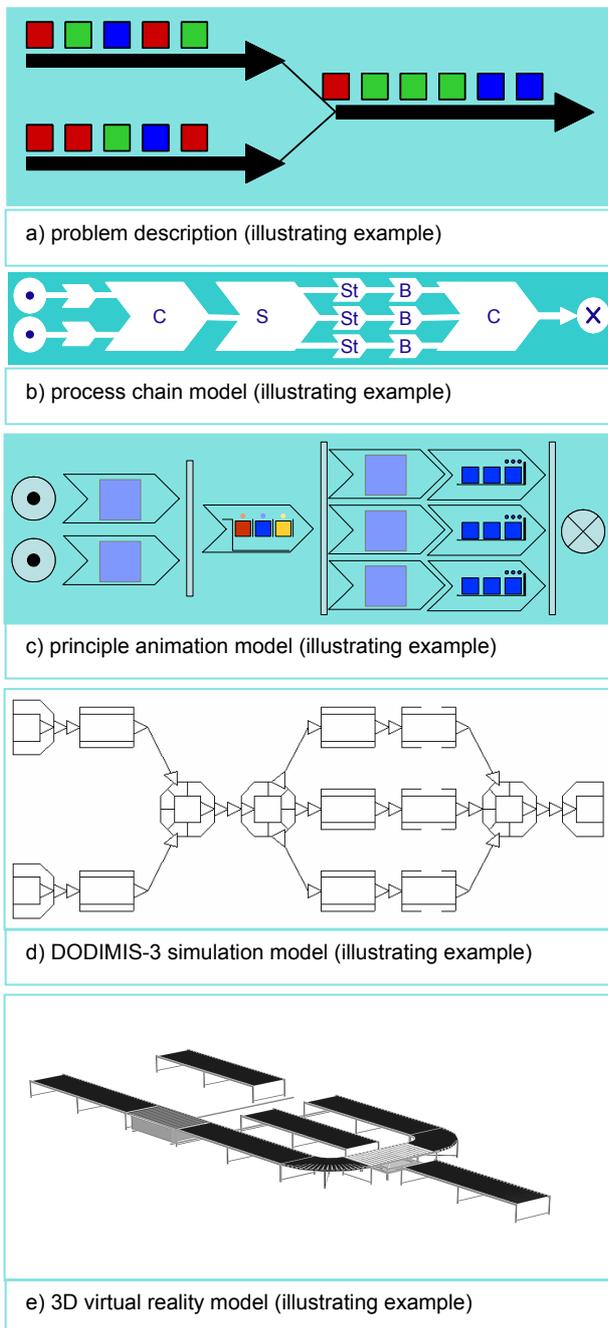


Figure 5: Evolution of the process description

The *3D virtual reality model* (see figure 5e) transfers the outcome of the simulation model, i.e. the trace file, into a dynamic three-dimensional representation of the process chain. The objective of this step is to ensure and validate that the materials flow links sources and sinks correctly also from the geometric point of view. Amongst others this might lead to the introduction of

additional moving, lifting or handling operations which have not been included yet. Furthermore this model also allows producing a nice looking, attractive visualization of the process which can be used for presentation purposes as well.

Although there are four different models to be developed in the course of the design process, the process chain model always forms the basic representation of the materials flow process as developed so far. Since the development and use of the other three models might lead to modifications of the process, the process chain model needs to be adapted and changed respectively after each modification step. In this way, also a correct documentation of the process (as the outcome of the design process) is produced, although focus of the design step, way of modelling and tools used for model building and experimentation do change.

### PROTOTYPE IMPLEMENTATION

For implementing the method of component-based logistics process design, tools from different categories need to be used and available. Currently, there is no automatic crossover from one model to another one. The person designing the process and developing the respective models is changing focus and way of model building as required, but clearly having in mind that there is precisely one process chain, animation and visualisation component for each of the elementary operations. Only for translating elementary operations into an executable simulation model there are alternatives in modelling (see figure 6). This is caused by the simulation package used, DOSIMIS-3, which is specialized to answer questions related to functionality and performance measures of logistics systems and processes and widely deployed in industry as well as logistics education and training in German-speaking countries. (Further information on DOSIMIS-3 in German or English languages can be found on the provider's website [www.sdz.de](http://www.sdz.de).)

operation	process chain	animation	DOSIMIS-3	visualization
moving	T transporting		SST FST UTM	
storing	St storing		LAG SST LFP	
uniting	C collecting		MON	
adding information	L labelling		AST MON	

Figure 6: Mapping list of elementary operations

In typical simulation projects DOSIMIS-3 is mainly used for analysing materials flow systems and their functionality instead of abstractly representing materials flow operations and processes without taking care of

any possible technological implementation. Because of this some of the elementary operations are principally represented within alternative simulation components which gives the modelling person some degree of freedom in designing the simulation model. To not reduce this kind of flexibility too much but at the same time enable automatic generation of the 3D model and its dynamical visualisation through interpreting the DOSIMIS-3 trace file (see also Bernhard and Jessen 2005), a mapping list describes which simulation component is applicable to represent which elementary operation.

Finally, it is necessary to point out that all models represent just the process, but not at all any underlying technical equipment that would make this process become possible. Even the 3D virtual reality model uses principle concepts to show the process in the real layout. If any object movement, for example, is realized in a continuous way by use of any type of conveyor or in a discontinuous way through a vehicle or truck, this is not defined yet. System design would be the next step in the developmental process, although a kind of interrelated design of materials flow process and materials flow system are highly recommended. The proposed method can provide extensive support to such kind of a procedure by guiding through a systematic and structured design process as well as by enabling and encouraging continuous documentation of the results achieved.

Prototype implementation of the method as described aimed at providing a mixture of online and offline services to potential users via the web (Neumann 2003). For this, a web-based learning, information and communication enabling wide-scale and multimedia education in logistics was used as the framework for providing access to the variety of tools and services, but also to knowledge needed for component-based logistics process design. LogEduGate ([www.logedugate.de](http://www.logedugate.de)) is a new kind of a logistics e-learning environment that is tailor-made for the specific needs of this complicated, complex, interdisciplinary field of knowledge. This logistics education gate interlinks a large number of knowledge units covering different aspects of logistics from engineering and business points of view at the same time and especially provides functionality to strengthen competencies in problem solving, decision making, organizing, designing etc. At the moment LogEduGate is developed to support university students in logistics in both modes, face-to-face and distance learning, but at a later stage it wants to meet training needs of logistics professionals, too.

For being integrated into the LogEduGate platform, no major changes to DOSIMIS-3 which runs on PC requiring Windows 95 or higher operating system where required. DOSIMIS-3 is available for download

to all registered users of a particular LogEduGate course. After installing the package on the own computer and without any further add-ons, the simulator automatically comes up with its demo version enabling to build and run small simulation models which do not consist of more than 15 elements. For dealing with more complex problems and building larger models an additional, personalized simulation ID is required that is centrally created and provided on demand. This ID is valid for a certain period of time according to the time scale of the simulation project. During this period the complete functionality of the package is available, whereas after the expiry date has passed by the software automatically returns to demo mode again. In this way, the user needs to be online for downloading the package, requesting for and receiving the simulation ID and submitting the model and trace files for automatic generation of the visualization model only. All time-consuming activities, like model building, validation, running experiments, watching animation or monitoring and analyzing results take place in the offline mode (figure 7). With this, the simulator works and can be used exactly in the same way as it would happen in an industry setting. There is no loss of speed and performance; online costs are reduced to a minimum once the package has been downloaded.

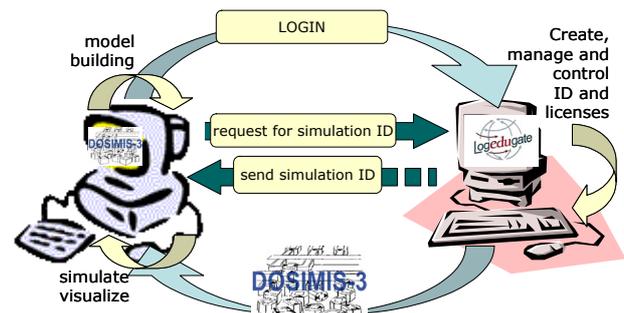


Figure 7: Embedding DOSIMIS-3 into LogEduGate

Further tools used for dynamical representation of the designed process are an animation toolkit and a visualization model generator (see Bernhard and Jessen 2005). Both tools were specially developed to meet the needs of this particular concept and method.

## CONCLUSIONS

As experience from integrating the method of component-based logistics process design into logistics courses at university level have shown this new way of thinking in processes instead of systems needs to be trained intensively, whereas the changes between models and tools did not cause major problems. The challenge consists in overcoming the traditional way of thinking in systems, system components or existing technical solutions at early stages of the design process. Thinking in terms of operations and processes only helps to define what kind of functionality is really needed and allows to play with basic principles instead of combining existing technical components only. With

this truly innovative technical solutions become possible. At the same time this is also the chance this new concept and method offers for reducing functional complexity of today's materials flow and logistics systems and with this the amount of investments into technical solutions as well. Of course the process chain model is not the end of the design process in logistics, but forms an excellent starting point for technical specification of functionality and designing an appropriate materials flow or logistics system afterwards. Here, the models developed in the steps of process design provide serious input. Especially simulation and visualization models and techniques are of tremendous relevance for this kind of a continuous developmental process. They are applicable even at early stages with poor knowledge on how the final solution might eventually look like. And they can be modified and adapted to the progress in a creative problem solving process representing the process first, but then being more and more changed into a representation of the specific system.

#### ACKNOWLEDGEMENTS

The paper is based upon results of a joint project funded by the German Ministry of Education and Research within its "New Media in Education" Program.

#### REFERENCES

- Bernhard, J. and Jessen, U. (2005). Web-based Service fort he Integration of Simulation and Visualization. In: Proceedings of SCS-ESM 2005 – 19th European Simulation Multiconference. (Riga, Latvia, June 1-4, 2005).
- Bolz, H. A. and Hagemann, G. E. (eds., 1958). Materials Handling Handbook. New York: The Ronald Press Company, 1958.
- Law, A. M. and Kelton, W. D. (2000). Simulation Modeling and Analysis. Boston: McGraw-Hill, 2000.
- Neumann, G. (2001). Wissensbasierte Unterstützung des Planers krangestützter Materialflusslösungen; Magdeburg: LOGiSCH 2001.
- Neumann, G. (2003). Simulation in a Problem-Based E-Learning Environment for Logistics. In: Proceedings of Workshop on Harbour, Maritime and Multimodal Logistics Modelling and Simulation. (Riga, Latvia, September 19-20, 2003) Ghent: SCS-Europe 2003.
- VDI-Richtlinie 3300. Materialfluss-Untersuchungen. 08/1973. (German Association of Engineers – Materials Flow Investigation)
- VDI-Richtlinie 3596. Darstellungsmethoden für den Materialfluss. 06/1975. (German Association of Engineers – Methods for Presenting Materials Flows)
- Woodley, D. R. (ed., 1964). Encyclopaedia of Materials Handling. Oxford et al.: Pergamon Press, 1964.
- Ziems, D.; Neumann, G. (1999): Logistikprozesse aus dem Konstruktionsbaukasten? Prozessbausteine helfen planen. In: R. Hossner (ed.); Jahrbuch Logistik; Düsseldorf: Verlag Handelsblatt, 1999, S. 263-267. (Logistics processes from the design box? Process components support planning.)
- Ziems, D.; Neumann, G.; Richter, K. (1998): Ganzheitlichkeit aus Bausteinen? - Lösungsansätze für eine durchgängige Prozeß- und Systemplanung in der Logistik. In: Proceedings of 4. Magdeburger Logistiktagung „Logistikplanung - Methoden, Werkzeuge, Potentiale“, Magdeburg: LOGiSCH 1998, pp. 120-140. (Complete solutions from building blocks? Approaches for a holistic design of logistics processes and systems.)

#### AUTHOR BIOGRAPHY



**GABY NEUMANN** received a Diploma in Materials Handling Technology from the Otto-von-Guericke-University of Technology in Magdeburg and a PhD in Logistics from the University of Magdeburg. Since 2003 she has been Junior Professor in Logistics Knowledge Management there. Since 1991 she has been also working as part-time consultant in logistics simulation. Her current activities and research interests are mainly linked to fields like problem solving and knowledge management in logistics, logistics simulation and planning, and technology-based logistics learning, didactics of teaching logistics as well as logistics competence profiling and assessment. She is author/co-author of two books, one educational multimedia module on warehousing and a series of e-learning modules in logistics as well as of 30 journal publications and 66 papers and presentations at national and international conferences. She co-ordinates the European logistics educators network for providing new technologies for logistics education inside the European Logistics Association (ELA-LogNet) and has been or is being involved in a couple of respective projects.

# INFORMATION ACQUISITION FOR MODEL BASED ANALYSIS OF LARGE LOGISTICS NETWORKS

Jochen Bernhard  
Department of Simulation Concepts and Instruments  
Fraunhofer-Institute for Material Flow and Logistics IML  
44227 Dortmund, Germany  
E-mail: [bernhard@iml.fraunhofer.de](mailto:bernhard@iml.fraunhofer.de)  
Sigrid Wenzel, Professor, Ph.D. Engineering  
University of Kassel, Institute of Production Engineering and Logistics  
Department of Mechanical Engineering  
34125 Kassel, Germany  
E-mail: [s.wenzel@uni-kassel.de](mailto:s.wenzel@uni-kassel.de)

## KEYWORDS

Information Acquisition, Model Based Analysis, Simulation, Logistics Networks.

## ABSTRACT

Bases of each efficient analysis of Large Logistics Networks are valid input data in the right quality, quantity and granularity. Considering these demands a goal and task oriented information acquisition by using several techniques of data acquisition, statistics and visualization is needed. This article gives an overview of current research results including theoretical definitions of terms like information, data, knowledge, a process oriented procedure model for information acquisition as well as different taxonomies of methods from data acquisition, statistics and visualization and their utilization. Finally, the article presents prospects of future research with the focus on the utilization of the information acquisition procedure model in applications.

## INTRODUCTION

Large logistics networks LLN exist wherever a large number of different items are transported via several stations sometimes using alternating transport carriers. In accordance with delimitation and position, these logistics systems are called supplier networks, production networks, distribution networks, freight forwarding networks, container cycles, transportation networks etc. The components of these large networks (e.g. people, companies, resources, orders) are linked to each other by a multitude of various relationships.

The current developments in LLN lead to new structures and rules. Thus, in recent years, a multitude of new network-based enterprise structures such as alliances, joint ventures, supplier networks, fractal factories, segmented factories, virtual enterprises and process-orientated organization forms have been propagated and discussed.

These demands lead to added use of model based analysis and planning methods like simulation and optimization. In this context the input data in an appropriate quality, quantity and granularity are prerequisite for high-value results. Therefore, an interdisciplinary research team with partners from Data Acquisition, Statistics and Visualization decided to develop a methods utilization model for the gathering of qualitatively high-value input data.

In spite of or perhaps because of the current information overload the investment for a goal-oriented information acquisition in simulation projects is much too high with almost 50% of the total project time (Rabe and Hellingrath 2001). Therefore the logistics planner needs a toolbox for an efficient goal-oriented information acquisition, which offers a support of all planning activities.

Therefore, a primary research goal is the development and adaptation of procedure models for the acquisition of context based information for the determination of specific input data for the model based analysis (especially simulation and optimization) of LLN. The results lead to the development of a method toolbox for supporting the process of the information acquisition with respect to target group and task oriented method utilization.

## DATA, INFORMATION, KNOWLEDGE

Any adequate approach to the acquisition of information for the modeling of logistics systems requires an exact differentiation between the terms data, information and knowledge and a distinct use of them. Mainly based on the information theory (Shannon and Weaver 1949; Nonaka and Takeuchi 1997) and an existing DIN standard (DIN 1995) following definitions were fixed:

*Data* consist of analog or digital signals or indications (syntax) and are used for the representation of information for the purpose of further processing.

*Information* covers statements and descriptions of characteristics and structures of certain objects or circumstances (semantic). Information is sent and received and due to this sender-receiver relation it is only valid for a specific purpose. Information is coded by data; it represents communicated and formalized knowledge and serves the increase of knowledge.

*Knowledge* ("mental models", skill, proficiency, know-how, experience, etc.) is bound to the consciousness and arises with the owner of the consciousness (individuals and/or collectives of individuals); knowledge is often difficult to formalize and hard to communicate.

These definitions for data and information built up the bases for all following sections.

### INFORMATION ACQUISITION IN SIMULATION

The utilization of information acquisition in modeling depends basically on the purpose of modeling and the related modeling method. For example, if a purely descriptive model has to be developed, different information will be required than during an optimization task or a simulation. In the following, the information acquisition for simulation is focused.

The basic simulation study process is described in the Part 1 of the VDI-guideline 3633 (VDI 2000). The information acquisition represents a part step of this process and replaces the data acquisition. Input information for the procedure of information acquisition are the project task, the object of study, the determined investigation goal as well as the conceptual model (see Figure 1). The existence of a first conceptual model, which determines the model boundary and the planned abstraction level, is prerequisite to carry out a purposeful and goal-oriented model information acquisition. The information acquisition itself delivers the concrete input data for the simulation model.

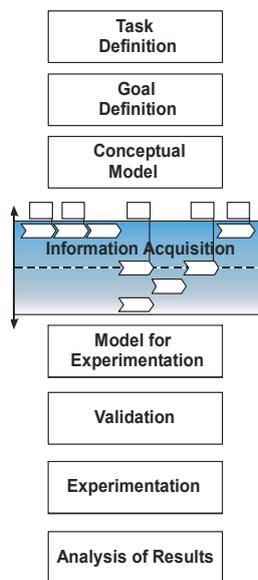


Figure 1: Information acquisition in the context of a simulation study.

### PROCEDURE MODEL OF INFORMATION ACQUISITION

The procedure model of information acquisition links the information view to the data view in a process

model. This includes the process steps of information collection and -estimation as well as the process steps of data management. Figure 2 gives an overview over the single process steps in their temporal sequence and in the respectively focused view of information and data. Where appropriate, iterations are necessary depending on the results of single process steps. For the processing of each process step several specific methods can be used.

The procedure model of information acquisition is based on the analysis of the given problem by considering the object of study, the aims of the study and the selected method of modeling. The *goal setting* process step is aided by taking characteristic information requirements of standard processes in logistics networks into account. After all this process step leads to principal information requirements for the given task and therefore for the model.

The following step *Identification of Information* is divided in three sub-steps: The *Analysis of Information Demand*, which derives the needed information from the principal information requirements, the *Analysis and Evaluation of the Existing Offer of Information*, including the detection and evaluation of all known and potentially useful sources of information, and the third sub-step *Assessment of the Available Information* which covers the comparison of the identified available information with the needed information by regarding aspects of expenses (e.g. availability, usability, costs, etc). This leads to the determination of potentially usable information.

The process step *Preparing the Collection of Information and Data* deals with the *Selection of Adequate Sources* and the *Identification and Selection of Appropriate Collection Methods* as well as the accomplishment of the corresponding preparation arrangements. With the aid of the evaluation of available information sources a decision is made on the sources to be used. Goal-oriented aspects of effort and benefit are the basis for the *Identification and Selection of Appropriate Collection Methods* to obtain an economical and goal-oriented utilization of the acquired information.

The *Collection of Information and Data* is composed of two sub-steps. By using the selected collection methods together with the chosen information sources the needed information is collected. Depending on the used collection method the collected information, for example as plain text, has to be transformed into corresponding data. In the next step the information and data has to be validated by the *Comparison of the Collected Data with the Needed Information*. This process step of information and data collection ensures that usable information with associated data is derived.

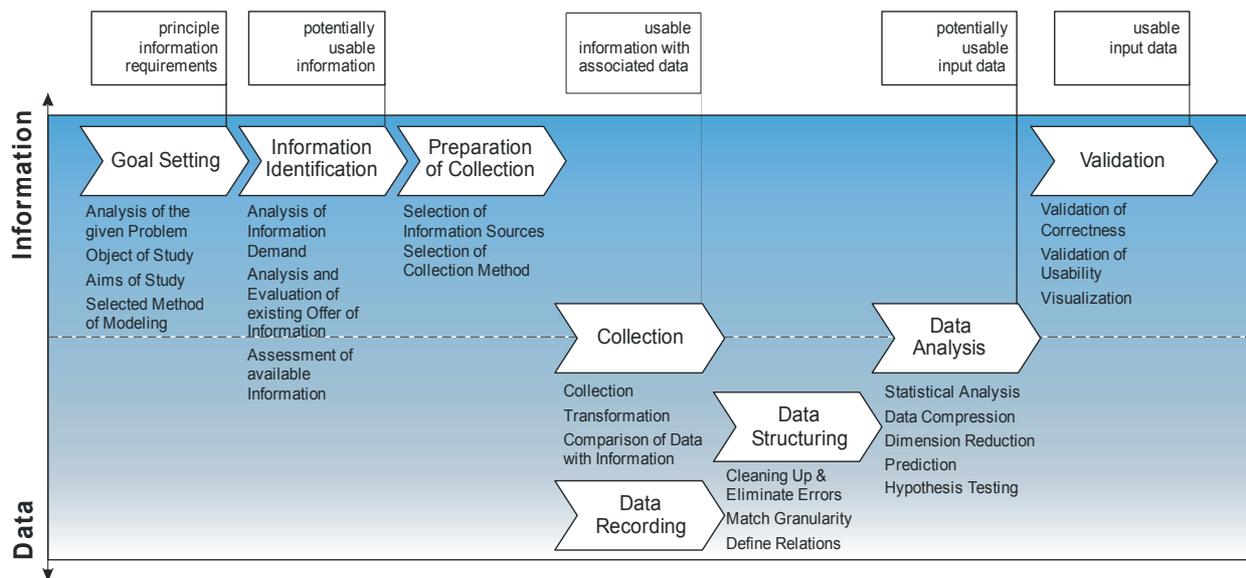


Figure 2: Procedure model of information acquisition.

Normally parallel to the process step described above the process of *Data Recording* includes all manually, semi-automatically or automatically operations to transfer data in a machine-readable structure and store them on a data medium. After this process step the data is available in digital form.

Data Structuring is a preprocessing step for the preparation of the recorded digital data for the following processes. This includes the clearing up, check on inconsistencies, as well as a scale-transformation taking the aims and the granularity level of the model based analysis into account. Also the definition of relations in data base systems by connecting indices and compatible variables leads to new logical rules.

The *Statistical Data Analysis* provides usable input data for the model based analysis of LLN. By using statistics methods for consistence checks (e.g. outlier identification, cross-validation), for reducing dimensions and complexity (e.g. variable selection, principal component analysis, classification methods and pattern recognition) or for a forecast the data are prepared, analyzed and interpreted with regard to quality and plausibility. On the one hand the data analysis works on the data, but on the other hand it depends directly from the previously identified information requirements.

The connected use of different statistical data analysis methods guarantees highly compressed and well prepared potentially usable input data only losing a minimum of information with respect to the collected data.

The *Validation* deals with the sufficient usability of the prepared data regarding the needed information and with its correctness and usage within the implemented simulation model. First, it has to be examined how much of the identified needed information is covered by

the potentially usable input data. This validation process is supported by the usage of different methods of information visualization for the comparative and explorative analysis. In the second stage, the two steps of verification and validation of the potentially usable input data take place by regarding its correctness and usage within the implemented simulation model. These steps are equivalent to parts of the verification and validation process of the simulation model itself (VDI 2000). Finally the validation process yields usable input data for modeling, simulation and optimization of LLN.

### TAXONOMIES FOR USING INFORMATION ACQUISITION METHODS

The utilization of the procedure model seen above necessitates enhancements for the purposive application of methods.

Therefore, parallel to the development of the procedure model different data acquisition, statistical and visualization methods were analyzed and classified regarding their use in the context of the information acquisition process. The worked out classification criteria represent the basis for the method taxonomies using in the different process steps to select a method for the concrete application.

In the area of the data acquisition, a comprehensive method screening was made for this both in internal areas of data acquisition discipline and in external areas, for example market research, development psychology and empirical social research. On basis of a direct cost-benefit-comparison the analyzed data acquisition methods were classified and rated with respect to their applicability for the data acquisition in LLN. Basis for the work within the range of the information and data acquisition is the distinction of the data acquisition methods in two method classes, the primary and the secondary collection. For primary collection

Criteria		Specifications							
Primary Function: Intention		Identification	Localization	Correlation	Association	Comparison	Structures and Patterns	Grouping	Classification
Type of Information to be Represented		qualitative	quantitative	qualitative and quantitative					
Relationship between Information-Objects		independent	relational	circular	hierarchical	network			
Information Encoding	Number of Viewable Information Components	1	2	3	4	5	6	7	n
	Dimension of Dependent Variables	nominal discrete	ordinal discrete	interval discrete	interval continuous	ratio discrete	ratio continuous		

Figure 3: Taxonomy for visualization methods from the information-related point of view

information and data are acquired directly for the investigation purpose, for example by means of questionnaires or interview as well as by self- and foreign-observation. A secondary collection is based on already existing data like external business data, internal business data collected for other purposes, official statistics etc. Known techniques are the inventory, the analysis of documents and the comparison of business companies. These primary and secondary methods have been analyzed by the authors especially in view of their usability to collect information on logistics systems (Hömberg and Jodin 2003). Goal-oriented aspects of effort and benefit are the basis for integrating the collection methods in category schemata. Only by choosing the most capable method, the economical and goal-oriented utilization of the acquired information can be achieved.

Based on multitude of different statistical analysis methods for basically different tasks and objectives, the applicable and useful groups of methods for the context of modeling of LLN were identified (Fender and Kuhnt 2003). The classification of a single method within a group by regarding the use in LLN could be carried out exemplary for the extensive method group Cluster Analysis. At this especially not only the complexity and quality of data, but also the data size becomes main criteria for the taxonomy.

An important step in the information acquisition process is the analysis of usable information and available data, whereby these often originate from different, heterogeneous data sources. The choice of suitable statistical analysis methods depends directly on the given goal and task, the kind of the available information, the structure of data records as well as the necessary granularity of the needed input data. For the multiplicity of different tasks and objectives as well as different data types in LLN usually a group of possible statistical methods exists for each area of application. Within these individual groups, different user-specific sources of error during selection and application of the

suitable analysis method have to be considered, as for example the formulation of wrong assumptions, the neglecting and mistaking of relevant information as well as the deriving of wrong conclusions from an assumption. Due to these errors there also arises strengthened likewise error in input data for the simulation, so that the results of a simulation, based on this input data, have no force of evidence (Gather et al. 2000; Wenzel et al. 2001). Against this background groups of statistic methods could be identified, from which a suitable procedure can be selected, depending on the use within the information acquisition. Only within these selected groups, the taxonomy of the methods specific for the application in LLN can be made. This taxonomy and their criteria have to consider the complexity and quality of the data, the size of the available data sources as well as statistical quality.

For the development of the taxonomy of visualization methods, different visualization techniques used in the area of Scientific Visualization and Information Visualization (e. g. Chi 2000; Harris 1999; Robertson 1991) were identified and analyzed. Furthermore a catalog of criteria together with the criteria specifications was built up by using existing taxonomies. Also research results of Visual Data Mining have been taken into account. But the existing taxonomies only obtain graphical criteria or are built up from the viewpoint of the simulation in production and logistics (VDI 2003). Criteria from the information-related point of view, that offer the utilization of the methods with respect to the task and target group, previously were not considered. The developed taxonomy represents on the one hand the basis for the classification of application relevant visualization methods on the other hand the need inquiry from user view for the method selection (Wenzel et al. 2003).

In addition to the criteria from the graphical point of view the taxonomy for visualization methods is completed by a survey of the criteria from the information-related point of view. These are the primary

functions of the visualization method in terms of its potential use and the description of the information which can be represented and imparted by the chosen visualization method. The taxonomy is shown in figure 3, a detailed description of all criteria and their specifications can be found in (Wenzel et al. 2003).

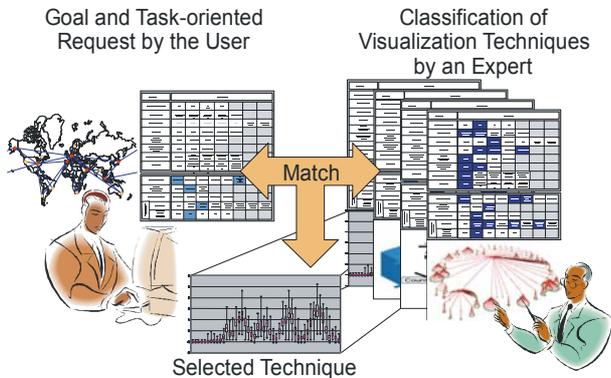


Figure 4: Taxonomy utilization for method selection.

The taxonomies were developed to match the requirements of a user with the specific characteristics of data acquisition, statistical or visualization methods. The precondition is the complete classification of all methods which are relevant for model-based planning of LLN. The user specifies his/her requirements with regard to task and purpose and other requirements, e. g. working field conventions, by means of the same taxonomy. However, this does not require a complete specification. By comparing his/her requirements with the entirety of available methods the user is led to a category of possible methods or, ideally, to just one method (see Figure 4). This comparison could be realized by an appropriate software program. The final selection has to be done manually by the user himself on the basis of extended criteria such as preparation effort and interpretation effort of the target group.

### STANDARD LOGISTICS PROCESSES AND DATA FOR INFORMATION IDENTIFICATION

To assist the process step *Information Identification* of the procedure model a systematic and standardized classification, description and evaluation of the needed input data in the context of the logistics processes are meaningful and necessary. For this reason typical standard processes on different levels of abstraction were and still has to be identified and - on the assumption that the simulation is used as analysis method - characteristic input data for the analysis and planning were assigned to logistics processes. Thereupon the relevant data for the modeling with the identified standard processes were selected and the needed quality, quantity and granularity was defined (e.g. topicality, accuracy, view period etc.). The modeling of the LLN based on the process chain paradigm (Kuhn 1995 and VDI 2001) and the use of these standard processes with the assigned data leads to an aggregating of all necessary input data on the necessary level.

### FUTURE PROSPECTS

The future research activities focus on a comprehensive definition and evaluation of the classification criteria for data acquisition, statistical and visualization methods. These classification criteria serve the development of method categories for simplifying the selection and utilization of specific methods. Also the derivation and definition of information categories, which are directly diverted from identified standard processes of logistics, as well as the generalization of the procedure model for the scope of LLN and not only for single applications are in the focus of the future work. The integrative methods utilization by using the procedure model has to be expanded with respect to a synergetic utilization of the methods. This requires the recognition of use and application specific dependencies of the methods of data acquisition, statistical analysis and visualization among each other. These dependencies have to be formulated as meta-information. In this context, especially several quality criteria are useful to get appropriate information as well as the corresponding input data.

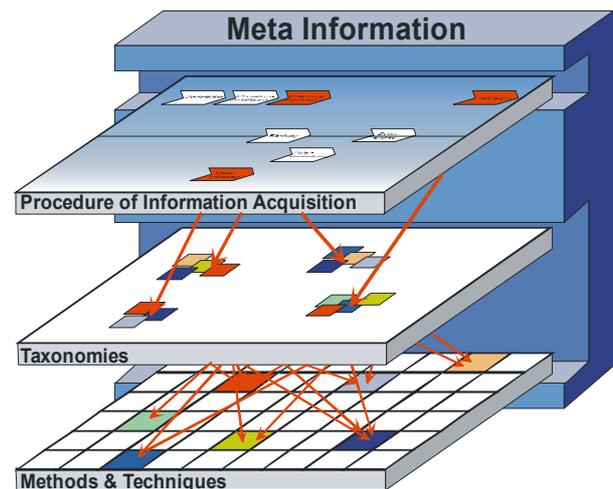


Figure 5: Methods Utilization Model.

The described results complement the methods toolbox and - together with the procedure model - lead to an integrated methods utilization model (see Figure 5), to carry out a purposeful information acquisition based on a given task definition and analysis method.

### ADDITIONAL INFORMATION

Additional information can be found under <http://www.sfb559.uni-dortmund.de>.

### ACKNOWLEDGMENTS

This research was supported by the Deutsche Forschungsgemeinschaft (DFG) as part of the Collaborative Research Center "Modelling of Large Logistics Networks" (559).

## REFERENCES

- Chi, E. H. 2000. A Taxonomy of Visualization Techniques using the Data State Reference Model. In: Proceedings of the IEEE Symposium on Information Visualization. IEEE Computer Society: Washington, DC, USA, 69-75.
- DIN Deutsches Institut für Normung e.V. 1995. Informationsverarbeitung – Part 1. DIN44300-1. Beuth, Berlin.
- Fender, T. and S. Kuhnt. 2003. Statistisches Datenmanagement zur Informationsgewinnung in GNL. Technical Report – Collaborative Research Center 559 „Modellierung großer Netze in der Logistik“ 03006. University of Dortmund, ISSN 1612-1376.
- Gather, U.; C. Becker and S. Kuhnt. 2000. Robust Methods for Complex Data Structures. In: Kiers, H. A. L.; J.-P. Rassin; P. J. F. Groenen and M. Schader. (Eds.). Data Analysis, Classification and Related Methods. Springer, Berlin Heidelberg New York, 315-320.
- Harris, R. L. 1999. *Information Graphics. A Comprehensive Illustrated Reference. Visual Tools for Analyzing, Managing, and Communicating.* Oxford University Press.
- Hömberg, K. and D. Jodin. 2003. Methodische Datenerhebung für effiziente Planungs- und Simulationsprojekte in der Logistik. In: Material Management (2003) 5. Grütter, Coburg/Paderborn. 20-25.
- Kuhn, A. 1995. Prozessketten in der Logistik – Entwicklungstrends und Umsetzungsstrategien. Verlag Praxiswissen, Dortmund.
- Nonaka, I. and H. Takeuchi. 1997. Die Organisation des Wissens. Campus, Frankfurt, New York.
- Rabe, M. and B. Hellingrath. 2001. Handlungsanleitung Simulation in Produktion und Logistik. SCS. San Diego.
- Robertson, P. K. 1991. A methodology for choosing data representations. *IEEE Computer Graphics & Applications* 11 (1991) 3, 56-67.
- Shannon, C. E. and W. Weaver. 1949/1976. Mathematische Grundlagen der Informationstheorie. Oldenbourg Wirtschaftsverlag GmbH, München.
- VDI Verein Deutscher Ingenieure. 2000. Simulation von Logistik-, Materialfluss- und Produktionssystemen. Guideline 3633 Page 1. Beuth, Berlin.
- VDI Verein Deutscher Ingenieure. 2001. Processes and process orientation in production logistics Example: Automotive Industry. Guideline 3600. Beuth, Berlin.
- VDI Verein Deutscher Ingenieure. 2003. Simulation und Visualisierung. Guideline 3633 Page 11. Beuth, Berlin.
- Wenzel, S.; C. Becker and D. Jodin. 2001. Methoden der Informationsgewinnung zur Bestimmung des Eingangsdatenraums für Simulationsmodelle in der Logistik. In: Panreck, K.; and F. Dörrscheidt (Eds.). Simulationstechnik. Proceedings of the 15th Symposium ASIM 2001. SCS-Europe BVBA, Ghent. 417-422.
- Wenzel, S.; J. Bernhard and U. Jessen. 2003. A Taxonomy of Visualization Techniques for Simulation in Production and Logistics. In: Chick, S.; P.J. Sanchez; D. Ferrin and D.J. Morrice (Eds.). Proceedings of the 2003 Winter Simulation Conference. IEEE, New Orleans, Louisiana, USA. 729-736.

## AUTHOR BIOGRAPHIES



**JOCHEN BERNHARD** was born in 1970 and studied mechanical engineering at the University of Kassel, Germany. After his graduation he worked at the Volkswagen AG and became an employee at the Fraunhofer Institute for Material Flow and Logistics in 2001. Since May 2004 he is leading the team Simulation Concepts and Tools. His main activities include the development and realization of concepts as well as the practical application of systems in simulation, planning and training. Emphases of his work are elaborating concepts for the coupling of simulation systems, the use of information visualization methods in simulation, collaborative engineering and internet-based learning. In addition to the development and management of national research and industry projects, he is an active member of the ASIM (Arbeitsgemeinschaft Simulation) group Simulation in Production and Logistics and the technical committee for Simulation and Visualization, a section of Committee A5 Modeling and Simulation of the Association of Engineers Society for Materials Handling, Materials Flow and Logistics Engineering (VDI-FML). You can reach him by e-mail at <bernhard@iml.fraunhofer.de>.



Prof. Dr. **SIGRID WENZEL** studied computer science at the University of Dortmund. She was employed at the Department for Material Handling and Warehousing at the University of Dortmund from 1986 to 1989. Since 1990, she has been employed at the Fraunhofer Institute for Material Flow and Logistics in the area of simulation concepts and tools as well as in the area of visualization techniques for production and logistics. She was head of the department from 1992 to 2004, acting department head of the division of enterprise modeling from 1995 to 2004 and additionally manager of the Collaborative Research Center 559 - Modeling of Large Logistics Networks at the University of Dortmund from 2001 to 2004. Since May 2004 she is professor and head of the Department of Mechanical Engineering, University of Kassel, Institute for Production Engineering and Logistics. In addition to this, she is a board member of the Arbeitsgemeinschaft Simulation (ASIM), spokesperson for the ASIM group Simulation in Production and Logistics, head of the technical committee for Simulation and Visualization and an active member of the technical committee for Modeling Processes and for Digital Factory, sections of Committee A5 Modeling and Simulation of the Association of Engineers Society for Materials Handling, Materials Flow and Logistics Engineering (VDI-FML). You can reach her by e-mail at <s.wenzel@uni-kassel.de>.

# WEB-BASED SERVICE FOR THE INTEGRATION OF SIMULATION AND VISUALIZATION

Jochen Bernhard and Ulrich Jessen  
Department of Simulation Concepts and Instruments  
Fraunhofer-Institute of Material Flow and Logistics  
44227, Dortmund, Germany  
E-mail: [bernhard@iml.fraunhofer.de](mailto:bernhard@iml.fraunhofer.de), [jessen@iml.fraunhofer.de](mailto:jessen@iml.fraunhofer.de)

## KEYWORDS

Simulation, Visualization, Logistics, Web-Service, e-Learning

## ABSTRACT

This article demonstrates the implementation and integration of a web-based service for the visualization of simulation models and their dynamics within the scope of logistics education at universities. Therefore, a survey of applications of visualization in simulation-based analysis of logistic systems as well as the utilization of web-based services in logistics planning is given. Both the technical realization and the definition of semantic dependencies as mapping rules or restrictions are discussed in detail. Finally, the results of a first validation and the future development are presented.

## INTRODUCTION

The tasks of logisticians deal with the various problems of planning, control and monitoring of worldwide linked supply and production networks. The tasks touch all levels of enterprise hierarchy, from strategic level comprising location planning, organization or logistics controlling to operational level of material flow functionality, e.g. packing, materials handling, warehousing, transshipment. The logistician is not only challenged to operate logistics systems but also to analyze, to plan and design as well as to improve them continually. Therefore, in particular model-based analytical methods as process analysis and simulation are playing a decisive role. On the one hand these methods based on their abstraction allow for focusing on the crucial tasks of planning, on the other hand for experiments of dynamics of great logistic networks as well as for comparison of different planning variants.

To prepare logisticians for the increasing use of IT technology in industrial practice, consequent integration of web-based e-Learning including corresponding multi-medial elements in education is necessary. Thus, discipline spanning competences are supported by new forms of communication. By strong linkage of interdisciplinary areas the understanding of various dependencies and interrelations is raised and the previous knowledge range is expanded to a location and time independent usable knowledge pool of logistics.

Due to limited time and funding academic education and training is mainly performed as lecture and/or seminar. These events only allow for a description and short demonstration of modelling and simulation tools. The complex tool functionality cannot completely be demonstrated and communicated. In this context the application of new media has a lasting effect. Besides the utilization of multi-media techniques for a more appealing presentation and intuitive imparting of knowledge the utilization of the World Wide Web (WWW) is playing a more and more important role for the education and training at universities. Dealing with individual web-based courses students can learn independently from time restrictions and location. Additionally, students can complete and extend their knowledge to their own interests browsing the complete knowledge database of the web-based education and training system. Relating to simulation students can find out their own way to apply the basic methodology and handle complex tools.

The inter-disciplinary approach of simulation methodology and the corresponding utilization of different methods as data acquisition and static analysis to 3-D visualization make the focusing on primary creative planning process necessary. This includes the analysis of the problem with implicit derivation of research goals, the modelling and validation of planned processes as well as experiments and result analysis on basis of simulation models (VDI 2000). In particular within the scope of model validation and the evaluation of dynamics the visualization (3-D animation, charts and diagrams) keeps on representing a crucial part for the creation of modules for education and training for simulation within logistics (VDI 2003).

Considering the mentioned conditions the development of an internet-based service for the automatic creation of animated Virtual Reality (VR) scenes on basis of simulation models and their dynamics description represents a reasonable solution. This enables the student to utilize a high-sophisticated realistic representation of the simulated system without specific knowledge about computer-based 3-D design. Requirements to be derived for the utilization process, transformation rules and the technical realization are discussed in the following two chapters. Finally, the

concrete realization of the web-based service to provide 3-D visualization in academic education is described.

### **FROM PROCESS DEFINITION TO SIMULATION AND VISUALIZATION**

The utilization of visualization in simulation includes both static and dynamic form. For example, static visualization shows content of simulation database as well as status and results of the experiment process as tables or graphs. The student gets information about structure of the model and key performance indicators but no information about the dynamics of the model. In particular the knowledge of dynamic coherences within simulation models lets the students recognize and understand occurring effects. Additionally, dynamic visualization gives students an insight in the dynamics of the process by animation and thus makes complex issues much easier to understand.

Generally, visualization improves the understanding of the model and sets the basis for a common communication between all persons involved (Wenzel et al. 2004). The goals of visualization differ according to the phase of the simulation study as well as the target group and focus on the one hand on the cognition enhancement (e.g. during validation and analysis) and on the other hand on the knowledge mediation (e.g. for presentation).

The automatic creation of a dynamic 3-D scene includes the transition from process modelling over simulation to 3-D visualization and the implicitly changing of modelling paradigms (process-oriented model, structure-oriented simulation, close to reality visualization). For the realization several requirements and restrictions have to be taken under consideration.

In this context the descriptive process model is based on the specification of parameterisable elementary operations. By arranging and parameterising these elementary operations the logistics process could be described (Neumann 2004). In this concrete application the transition into a simulation model is not supported in a technical way because the utilization of a common commercial software tool was intended. To ensure an appropriate transformation of the process description into a simulation model specific mapping rules and restrictions can be used. The transition of the simulation model into an animated Virtual Reality scene is supported per automatism. The automatic transformation of the simulation model into a VR scene and the transmission of the dynamic flows into animation of the scene have special influence on the technical basis. Based on the selected simulation tool and its non-proportional scaling modelling technique as well as the use of abstractions, mapping rules and restrictions as well as the specification of a specific system technology for the close to reality visualization are necessary for an appropriate transformation. The

concrete measures for this are described in chapter Web-based SimVis Service.

### **WEB-SERVICES**

On the field of mechanical and plant engineering apart from their technical features capital goods are increasingly assessed according to their contribution to the solution of the problem. Thus, manufacturers of production and logistic systems expand their technically orientated product spectrum with additional (value-added) services, which increase the efficiency during the ramp-up and the operation processes of complex systems significantly (Hellmann et al. 2003). The World Wide Web is the ideal platform to implement a service process and to make this service available to customers.

For example, the web-based use of simulation and 3D-visualization as a value-added service of logistics system planning and tender generation forms the common, consistent and confidence-building discussion basis for all people involved in the planning process (Hellmann, Jessen and Wenzel 2003). It prevents possible deficits and misinterpretations and allows the integration of system planner as well as customers during an early project phase. Herewith the users are offered a convenient access to advanced planning techniques. Users don't have to invest in expensive simulation and visualization tools since the tools are utilized as part of the web-based service. Additionally, a well-defined process of the individual service simplifies the usage of simulation and visualization techniques by focussing on fundamental information and reduces effort for planning and tool-specific staff training and education.

Students are the future user of advanced simulation and visualization techniques. Thus, web-based services also lend themselves to students' training and education at universities (Neumann 2005). Lecturers expand their spectrum of lectures and seminars with additional (value-added) services, which increase the efficiency and effectiveness of learning about complex issues significantly. Students are enabled by the services to design their own (additional) training program usable at any time at any location.

Besides creation of organizational conditions and definition of service processes, a suitable IT infrastructure is needed to implement web-based services for academic training and education. The basic technology to set up web-based services is available and worldwide utilized, standards for services are defined. For example, IBM, BEA and Oracle provide comprehensive IT platforms including specific functionality for authentication, authorisation, security; web-portals etc. and – last but not least – to set up web-based services.

## WEB-BASED SIMVIS SERVICE

From technical point of view the SimVis service is divided into three main components: the user interface, the service platform and the processing kernel (see Figure 1). The user interface was realized based on common web technology, so that the user does not need additional software. To upload the necessary input data, a model file and a so-called trace file, the user is provided with a convenient web-interface. The model provides the necessary information about the model structure (principle component layout, technical parameters, topology, etc.) for the automatic generation of the VR scene. The trace file provides a summary of all state changes which have occurred during simulation run and is used for adding dynamics of simulation into the VR scene. Furthermore, a valid email address for dispatching different status information and the transmission of the result has to be specified. The result, an animated Virtual Reality Modelling Language (VRML) scene, can be viewed with a generic internet browser after the installation of a free VRML plug-in.

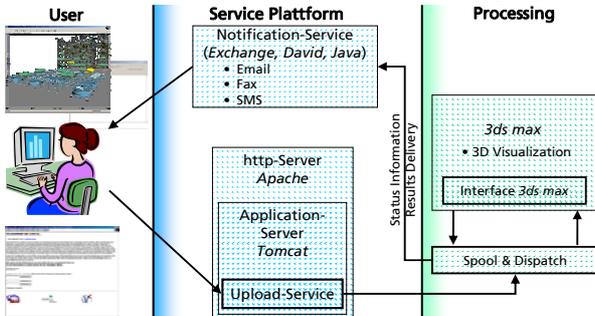


Figure 1: SimVis-Process and Technical Details.

The central service platform consists of application server (including web server) and notification component. The application server provides an upload service for the transfer of necessary input files from the user to the processing kernel as well as a verification of the email address which defines the target for sending the service results. To obtain an implicit, but very rudimentary access control, the web service checks authentication and authorisation by the unique IP-Address of the calling server of the e-Learning-environment. Moreover, the service platform comprises a notification service, which uses existing infrastructure as MS-Exchange, DAVID or any POP email server for messaging. This service is used for the transmission of status information and the delivery of the results.

The processing kernel consists of a spool and dispatch component and an intrinsic processing unit. The spool and dispatch component receives the task, puts it into a queue, and delivers status information as well as results of the task. The intrinsic processing component is realized as extension of a professional 3D modelling and visualization tool. The selected tool 3ds max by Discreet provides a script language which has been used for the development of the automatism for transformation of simulation models into 3-D scenes.

The transformation process is supported by a component library, which has been developed by Fraunhofer IML. It consists of parametrisable graphical elements representing typical material handling elements. In addition, the dynamics of simulation defined as trace file is automatically transformed into the animation of the scene. At last the animated scene is exported format-optimized into a VRML file and delivered to the user.

To ensure an adequate, close-to-reality visualization of the simulation model several restrictions and implicit mapping rules have to be specified. Basically, these rules define a pre-selection of graphical elements as representatives of simulation elements and their positioning in animation scene as well as standard behaviour of the graphical elements based on simulation events. In case of the selected discrete-event simulation tool DOSIMIS-3, additional mapping rules have to be defined, which concerned the avoidance of typical abstractions in the non-proportional scaled simulation modelling. For example non-straight-line transportation processes (curves, turn-table, etc.) must be explicitly modelled (see Figure 2). This could be performed either by using several elements or by an explicitly curve modelling using the conveyer element. A purely graphic modelling by using the logical building blocks linkage could not accordingly be visualized, because this information is not specified in the model file.

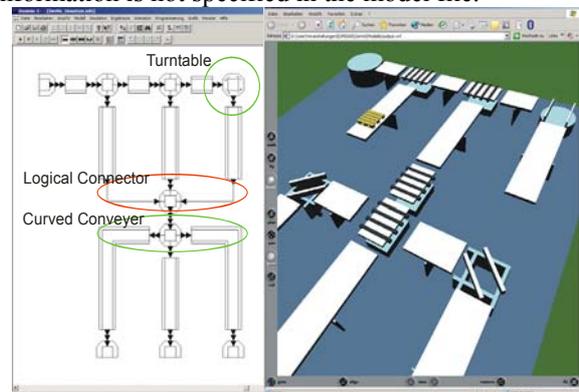


Figure 2: Example for Restrictions in DOSIMIS-3.

In the sense of documentation and user guidance both the mapping rules for the manual transition between process and structure modelling as well as the automatic transition between structure modelling and visualization are described in a table (see Figure 3).

Logistics Operations	Process	DOSIMIS-3	Visualization
Change Location		SST FST UTH	
Keeping		LAG SST LFP	
Combining		MON	
Add Information		RST MON	

Figure 3: Mapping Table for Transformations.

## UTILIZATION OF THE SIMVIS SERVICE

In addition to the description of the component-based process design (Neumann 2005) this chapter shows an illustrative example for the utilization of the SimVis service. Starting with a task description and the process model on the basis of elementary material flow elements this section focuses on the modelling of a suitable simulation model by concerning restrictions and mappings as well as the automatic transformation into a dynamic VR-Scene by the SimVis service.

### Task Description:

As an example for a component-based process design following task should be given: to define an efficient material flow for the production of a merchandising product, a football with the print of an emblem of one of three different soccer-clubs. For this, on average every 20 s with a deviation of 3 s the system source produces 4 plain footballs as batch entering the system on a loading aid. The footballs need 40 s to be printed, 40 percent of the balls with the emblem of Club A, 30 percent with Club B and 30 percent with Club C emblem. 10 percent of the printed balls are checked, 3 percent of them again leave the system as rejections. The subsequent packaging needs 40 s. At this stage the balls get unique identifier. In the shipping area batches of 4 balls of Club A, 3 balls of Club B and 3 balls of Club C each have to be assorted which finally leave the system.

### Process Model:

First, starting with the task description the central useful and supporting processes are identified and described within an approximate process chain model. The *useful processes*, planned processes directly participating in the creation of value for the customer, comprise accordingly to VDI 3600 (VDI 2001) processing, assembly, packing, development, conditioning, designing et al. The *supporting processes* are planned processes which support the execution of the useful processes as transporting, storing, checking, order placement and acceptance, buffering and transhipping.

In contrast *blind processes* are processes like checking back, waiting times, searching for parts, etc., and *error processes*, reject production, over-storage of goods, faulty commissioning, delivery to wrong consignee, etc. Main goal of a planning process is to avoid blind and error processes as well as to minimize the supporting processes. As second step the process chain model is detailed and the single process elements are parameterised accordingly to the task description (see Figure 4). For this, different kinds of influencing the process chain or the parameters like omission, combination, replacement or paralleling of process steps are possible. The strategic goal is the increase of value-creating share while minimising the process costs (Kuhn 1995).

### Simulation Model:

As next step based on process description using mapping rules and taking defined restrictions into account the process chain model is manually transformed into a simulation model which allows the investigation of dynamic system behaviour as simulation experiment. For modelling, the simulation tool DOSIMIS-3 (DS3) was utilised. Here, defined mappings and restrictions have to be considered by the modeller in terms of subsequent visualisation (see mappings in table 1 and simulation model in figure 5).

Process	DOSIMIS-3 Element
Source	Source
Opening OR-Connector	Distributor
Printing	3 Conveyers, 3 Workstations
Closing OR-Connector	3 Conveyers, 1 Combining Station

Table 1: Mapping Table Process–DOSIMIS-3.

The parameterisation of simulation elements is done accordingly to the process chain modelling. Fixed data of the process model could be used, e.g. inter-arrival time, process time and distributions. Additionally, 2 m/s are defined as standard speed of transportation and 1.2 m as standard length of a pallet place (length of euro-pallet).

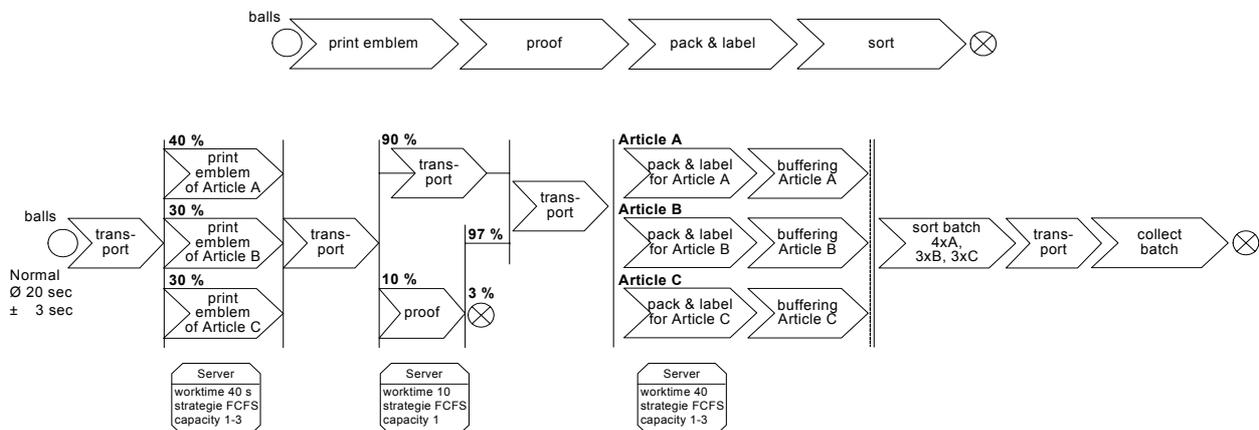


Figure 4: Process-Chain model.

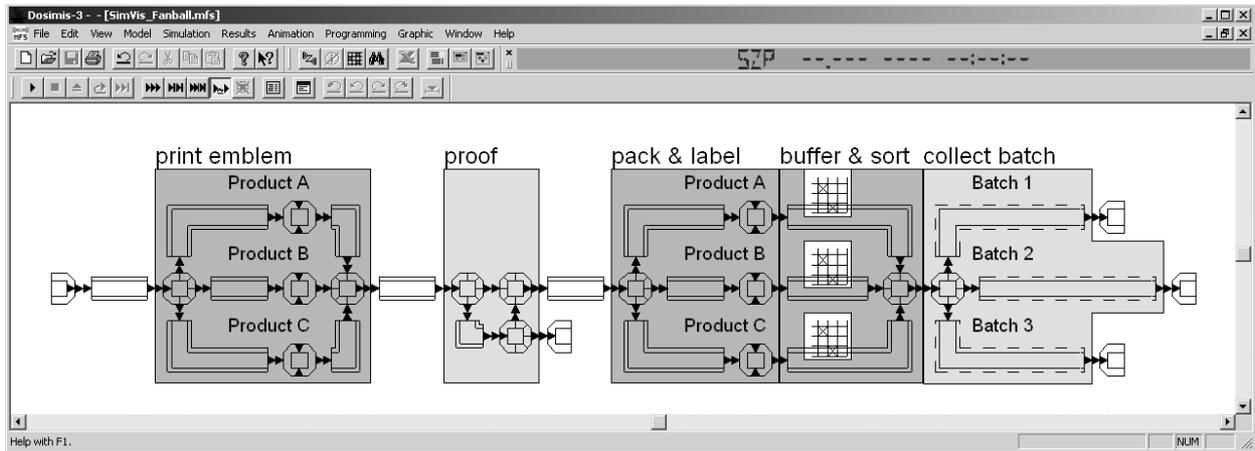


Figure 5: DOSIMIS-3 simulation model.

#### Visualization Model:

The visualisation model is automatically created by the SimVis service (see previous chapter for description). For this, accordingly to mappings and restrictions a static visualisation model is generated from the DS3 model. Subsequently, the dynamics information is extracted from simulation model and added to the animation scene. Finally, the animated scene is exported as VRML file (Virtual-Reality-Modelling-Language according to ISO/IEC 14772-1:1997 Standard) to show it on standard (mostly freely available) VRML viewer.

The intrinsic processing unit of the SimVis service generates 3-D scenes based on completely defined error-free DS3 models considering implicit mappings and assumptions. In this example material flow using euro-pallets is supposed, i.e. sizes (length, width) of all graphical elements are related to standard size of euro-pallet (1.2m x 0.8m). For this, a library of parametrizable, adaptable and re-usable graphical animation elements for the professional modelling and animation tool 3ds max by Discreet has been built up. The elements represent real system components based on geometry and behaviour model and are used for quick and flexible 3-D modelling and animation of logistics and production systems. Using information of the DS3 model file as simulation element positions and directions as well as length and specific element type, the processing unit selects animation elements from the

library and arranges them within the 3-D scene. Because of the non-proportional scaled DS3 models a grid (1.2m x 1.2m) is defined for the generation of the 3-D scene. The animation elements as representations of each DS3 element are selected from the library, positioned considering the grid structure and parameterised accordingly to an implicit mapping table and possibly on basis of technical attributes, e.g. a conveyor with its exact conveying distance.

The *transfer of dynamics* from simulation model to animation utilises the information of the event list represented by the trace file. It has to be pointed out that there is a fundamental difference between the discrete-event-based descriptions of dynamics of simulation and the activity-based description of animation. Events only describe single points in time. Activities define a period with starting point and duration. Thus, a translation process has to be defined, which based on event information generates a dynamics description, e.g. creation of pallet movement activity based on start and end event of a conveyor in simulation. Within the scope of the SimVis service this task is performed by a translator individually configured for DS3 and 3ds max.

When the dynamics are transferred from simulation to animation the completely animated scene is available in 3ds max. Potential export formats are picture, movie and accessible Virtual-Reality scene. Within the scope

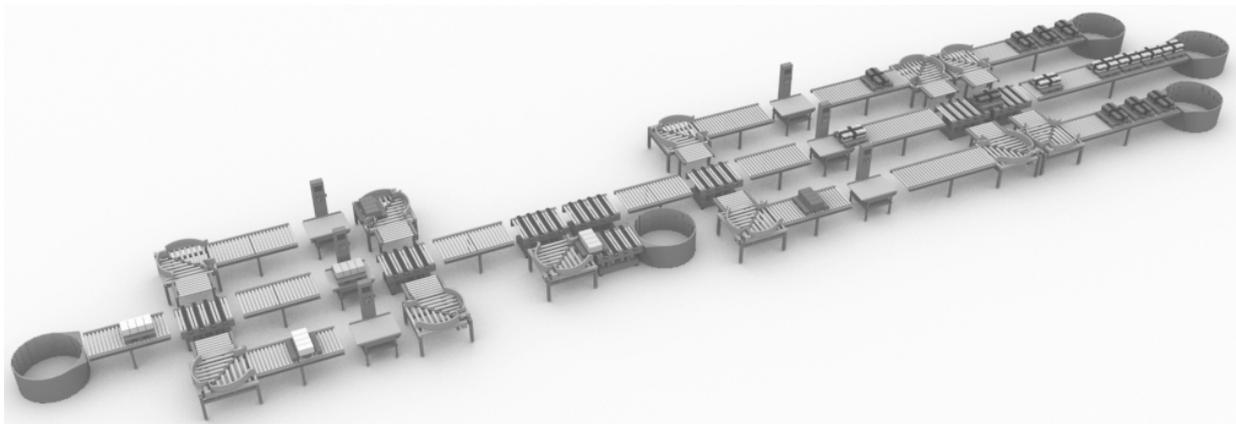


Figure 6: Visualization model.

of the SimVis service an animated *VRML file* is generated. Several measures improve the quality of the resulting file as polygon reduction, element referencing, file size compression etc. Finally, this file is sent to the user by email.

## VALIDATION AND FUTURE PROSPECTS

The application of web-based services for the visualization of simulation models within the scope of academic education has demonstrated that particularly the application of the service for model validation and presentation of simulation results has been highly accepted by the students. However, default mapping rules and restrictions limited the work of the students with the service. Disregarding the mapping rules, transferring incomplete models by the students and a few deficiencies in error handling causes breakdowns of the web-based service. To fix the deficiencies, additional technical measures making the service stable regarding hardware, software bugs as well as handling errors including bug-reporting, have been taken. Additionally, the defined restrictions and mapping rules are permanently improved in order to increase the modelling flexibility. For further validation the utilization of the service is planned within the scope of other tutorials and workshops. By doing this, the service is evaluated within a different context and proved for other planning tasks. The goal is a generalized service.

## ADDITIONAL INFORMATION

Additional information can be found under <<http://www.logedugate.de>>.

## REFERENCES

- Hellmann, A. and U. Jessen. 2002. Web-based planning and operation support using e-services. In: 5th international Conference on Managing Innovations in Manufacturing (MIM) 2002, Proceedings Milwaukee, Wisconsin, USA.
- Hellmann A.; U. Jessen and S. Wenzel. 2003. e-Services - A Part of the "Digital Factory". In: Bley H. (Eds.). Proceedings of the 36th CIRP International Seminar on Manufacturing Systems. Schriftenreihe Produktionstechnik. Vol. 29. Saarbrücken, Germany. 199-203.
- Neumann, G. 2004. Problembasiertes e-Learning in der Logistik: Die Integration von Simulation und Visualisierung in die virtuelle Lernumgebung LogEduGate. In: Schulze, T.; S. Schlechtweg and V. Hinz (Eds.). Proceedings of the Conference "Simulation und Visualisierung 2004". Magdeburg, Germany. 39-51.
- Neumann, G. 2005. How to Solve the Puzzle? Simulation Support for Component-Based Process Design in Logistics. Accepted for the Proceedings of the Conference "9<sup>th</sup> European Conference on Modelling and Simulation". Riga, Latvia.
- Kuhn, A. 1995. Prozessketten in der Logistik – Entwicklungstrends und Umsetzungsstrategien. Verlag Praxiswissen, Dortmund.
- VDI Verein Deutscher Ingenieure. 2000. Simulation von Logistik-, Materialfluss- und Produktionssystemen. Guideline 3633 Page 1. Beuth, Berlin.

- VDI Verein Deutscher Ingenieure. 2001. Processes and process orientation in production logistics Example: Automotive Industry. Guideline 3600. Beuth, Berlin.
- VDI Verein Deutscher Ingenieure. 2003. Simulation und Visualisierung. Guideline 3633 Page 11. Beuth, Berlin.
- Wenzel, S.; J. Bernhard and U. Jessen. 2003. A Taxonomy of Visualization Techniques for Simulation in Production and Logistics. In: Chick, S.; P.J. Sanchez; D. Ferrin and D.J. Morrice (Eds.). Proceedings of the 2003 Winter Simulation Conference. IEEE, New Orleans, Louisiana, USA. 729-736.

## AUTHOR BIOGRAPHIES



**JOCHEN BERNHARD** was born in 1970 and studied mechanical engineering at the University of Kassel, Germany. After his graduation he worked at the Volkswagen AG. He has become an employee at the Fraunhofer Institute for

Material Flow and Logistics in 2001. Since May 2004 he has led the team Simulation Concepts and Tools. His main activities include the development and realization of concepts as well as the practical application of systems in simulation, planning and training. Emphases of his work are elaborating concepts for the coupling of simulation systems, the use of information visualization methods in simulation, collaborative engineering and internet-based learning. In addition to the development and management of national research and industry projects, he is an active member of the ASIM (Arbeitsgemeinschaft Simulation) group Simulation in Production and Logistics and the technical committee for Simulation and Visualization, a section of Committee A5 Modeling and Simulation of the Association of Engineers Society for Materials Handling, Materials Flow and Logistics Engineering (VDI-FML). You can reach him by e-mail at <[bernhard@iml.fraunhofer.de](mailto:bernhard@iml.fraunhofer.de)>.



**ULRICH JESSEN** was born in 1958 and studied computer sciences at the University of Dortmund. He has been employed at the Fraunhofer Institute for Material Flow and Logistics since 1990.

His main activities include the development and realization of simulation, planning, and assistance systems, with an emphasis on elaborating concepts for the application and implementation of visualization and coupling techniques. In addition to the development and management of national and international research and industry projects, he is an active member of the technical committee for Simulation and Visualization as well as the Digital Factory, sections of Committee A5 Modeling and Simulation of the Association of Engineers Society for Materials Handling, Materials Flow and Logistics Engineering (VDI-FML). You can reach him by e-mail at <[jessen@iml.fraunhofer.de](mailto:jessen@iml.fraunhofer.de)>.

# **Simulation of Intelligent Systems**



# Investigation on Evolutionary Deterministic Chaos Control – Extended Study

Ivan Zelinka  
Institute of Proces Control and Information Technologies  
Faculty of Technology  
Tomas Bata University in Zlin  
Mostni 5139 Zlin, Czech Republic  
zelinka@ft.utb.cz

## KEYWORDS

spatiotemporal chaos, coupled map lattices, evolution, optimisation, SOMA, differential evolution, genetic algorithm

## ABSTRACT

This contribution presents the results of an investigation on deterministic spatiotemporal chaos control by means of evolutionary algorithms. Three evolutionary algorithms are used for chaos control: differential evolution, self-organizing migrating algorithm and genetic algorithm. Models of spatiotemporal chaos, so called coupled map lattices, are used. The main aim of this investigation was to show that evolutionary algorithms are capable of deterministic chaos control when the cost function is properly defined. The investigation consists of four different case studies with increasing calculation complexity. For each algorithm 50 simulations were carried for each problem to verify and to demonstrate the robustness of the methods used.

## 1 INTRODUCTION

The term *deterministic chaos control* (DCC) was first coined by Ott E., Greboki C., Yorke J.A. in (Ott E., Greboki C., Yorke J.A., 1990). It is the process of deriving and applying a control law, so that the originally chaotic process would stabilize itself either on a constant level of output values, or in an  $n$ -periodic cycle. Since the introduction of DCC, many methods for deriving control laws were developed, based on the original method (Ott E., Greboki C., Yorke J.A., 1990), for example pole placement (Greboki C., Lai Y.C. 1999) and delay feedback (Just W., 1999). Many of the published methods, which were originally developed for the classic DCC, were adapted for the so called spatiotemporal chaos, which is represented by coupled map lattices (CML), given by (1). Models of this kind are based on sets of spatiotemporal (for 1D, Figure 1) or spatial (for 2D, Figure 2) cells, which represent appropriate states of system elements. A typical example is CML based on the logistic equation, (Hilborn R.C.1994), (Guanrong Chen, 2000) which is used to simulate the behaviour of systems which consist of  $n$  mutually joined cells – logistic equations.

Control laws derived for CML controlling usually rely on existing knowledge about the system structure (Schuster H.G., 1999), or on the use of an external observer (Guanrong Chen, 2000).

The main aim of this research was to show that evolutionary algorithms (EAs) are capable of controlling CML (as it was previously shown for temporal DCC in (Hendrik Richter & Kurt J. Reinschke, 2000), (Hendrik Richter, 2002), (Hendrik Richter & Kurt J. Reinschke 2000a)) as well as deterministic methods, without the need of internal system knowledge, and that operates with CML as with a black box. The ability of EAs to successfully work with black box problems was demonstrated many times, for example for real-time control of plasma reactors (Zelinka I., Nolle L., 2005).

$$x_{n+1}(i) = (1 - \varepsilon)f(x_n(i)) + \frac{\varepsilon}{2}(f(x_n(i-1)) + f(x_n(i+1))) \quad (1)$$

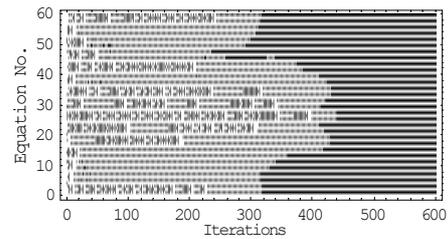


Figure 1 1D CML with pattern T1S2

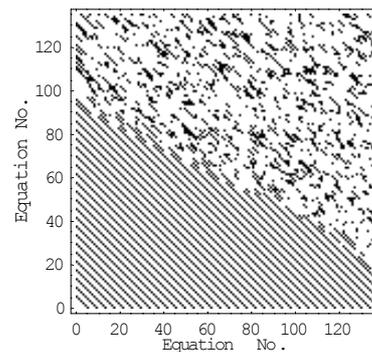


Figure 2 2D CML with pinning imported through lattice on position (0,0). The resulting control pattern (left) is visible as well as spatiotemporal chaos (right)

## 2 PROBLEM DESIGN

### 2.1 Problem selection and case studies

The class of CML problems chosen for this comparative study was based on case studies reported in (Schuster H.G., 1999). In general, CML control means setting of such pinnings (control CML sites) and their pinning values (control values) so that a system stabilizes itself

on expected spatiotemporal patterns. CML as an object of study was chosen because it shows chaotic behavior and its level of complexity can be quite rich. This research consists of four parts, presented in increasing order, from the calculational complexity point of view, and was based on the work of (Hu G., Xie F., Xiao J., Yang J., Qu Z., 1999) and (Zelinka Ivan, 2005). The first one is focused on pinning values estimation for a priori given pinning sites. In the second one pinning sites with a priori given pinning values were estimated by EA. The third simulation was an enlargement of the previous simulation – EA was used to find the minimal number of pinning sites and the fourth simulation was focused on mutual estimation of pinning sites and values, i.e. EA was searching for the minimal number of pinning sites and optimal (i.e. as many as possible) pinning values. All simulations were based on the same CML model and were repeated 50 times for each EA with new initial conditions for each simulation. In total there 600 independent simulations of spatiotemporal DCC were carried out.

## 2.2 The Cost Function

The fitness (cost function) has been calculated using the distance between the desired CML state and the actual CML output (2). The minimal value of this cost function, representing the best solution, is 0. The aim of all the simulations based on (2) was to find the best solution, i.e. a solution that returns the cost value 0. This cost function was used for the first two case studies (pinning values setting, pinning sites setting). In the remaining two case studies the cost function (3) was used. It is synthesised from the cost function (2) so that two terms are added. The first one (“*p1*”) represents the number of pinning sites in CML. The second one (“*p2*”) is added here to “*attract attention*” of the evolutionary process on the main part of the cost function. It is in fact the number of used pinning sites, i.e. in the optimal case cost function (3) should return *p1*. If the second part of (3) would not be present, then mainly *p1* would be optimised, so the results would not be acceptable (proved by simulations). Indexes *i* and *j* are coordinates of lattice elements, i.e.  $CML_{i,j}$  is *i*<sup>th</sup> site (equation) in *j*<sup>th</sup> iteration. In all simulations for the target of control,  $TS_{i,j}$  was set to 0.75, i.e. CML behavior was controlled to this simplest state.

$$f_{cost} = \sum_{i=1}^{10} \sum_{j=80}^{100} |TS_{i,j} - CML_{i,j}|^2 \quad (2)$$

$TS_{i,j}$  - target state of CML

$CML_{i,j}$  - actual state of controlled CML

$$f_{cost} = p1 + \left( p2 \sum_{i=1}^{10} \sum_{j=80}^{100} |TS_{i,j} - CML_{i,j}| \right)^2 \quad (3)$$

$TS_{i,j}$  - target state of CML

$CML_{i,j}$  - actual state of controlled CML

*p1* - number of actually selected pinning sites

*p2* - 1000, heuristically set weight constant

## 2.3 Optimisation Algorithm and Parameter Setting

For the experiments described here, stochastic optimisation algorithms, such as Differential Evolution (DE) (Price K. 1999), SelfOrganizing Migrating Algorithm (SOMA) (Zelinka Ivan, 2004), and Genetic Algorithm had been used. Alternative algorithms Simulated Annealing (SA), are now in process, and results are hoped to be presented soon.

Differential Evolution is a population-based optimization method that works on real-number coded individuals. For each individual  $\vec{x}_{i,G}$  in the current generation *G*, DE generates a new trial individual  $\vec{x}'_{i,G}$  by adding the weighted difference between two randomly selected individuals  $\vec{x}_{r1,G}$  and  $\vec{x}_{r2,G}$  to a third randomly selected individual  $\vec{x}_{r3,G}$ . The resulting individual  $\vec{x}'_{i,G}$  is crossed-over with the original individual  $\vec{x}_{i,G}$ . The fitness of the resulting individual, referred to as perturbed vector  $\vec{u}_{i,G+1}$ , is then compared with the fitness of  $\vec{x}_{i,G}$ . If the fitness of  $\vec{u}_{i,G+1}$  is greater than the fitness of  $\vec{x}_{i,G}$ ,  $\vec{x}_{i,G}$  is replaced with  $\vec{u}_{i,G+1}$ , otherwise  $\vec{x}_{i,G}$  remains in the population as  $\vec{x}_{i,G+1}$ .

Differential Evolution is robust, fast, and effective with global optimization ability. It does not require that the objective function is differentiable, and it works with noisy, epistatic and time-dependent objective functions.

SOMA is a stochastic optimization algorithm that is modelled on the social behaviour of cooperating individuals (Zelinka Ivan, 2004). It was chosen because it has been proven that the algorithm has the ability to converge towards the global optimum (Zelinka Ivan, 2004). SOMA works on a population of candidate solutions in loops called *migration loops*. The population is initialized randomly distributed over the search space at the beginning of the search. In each loop, the population is evaluated and the solution with the highest fitness becomes the leader *L*. Apart from the leader, in one migration loop, all individuals will traverse the input space in the direction of the leader. Mutation, the random perturbation of individuals, is an important operation for evolutionary strategies (ES). It ensures the diversity amongst the individuals and it also provides the means to restore lost information in a

population. Mutation is different in SOMA compared with other ES strategies. SOMA uses a parameter called PRT to achieve perturbation. This parameter has the same effect for SOMA as mutation has for GA.

The novelty of this approach is that the PRT Vector is created before an individual starts its journey over the search space. The PRT Vector defines the final movement of an active individual in search space.

The randomly generated binary perturbation vector controls the allowed dimensions for an individual. If an element of the perturbation vector is set to zero, then the individual is not allowed to change its position in the corresponding dimension.

An individual will travel a certain distance (called the path length) towards the leader in  $n$  steps of defined length. If the path length is chosen to be greater than one, then the individual will overshoot the leader. This path is perturbed randomly.

For an exact description of the algorithms, see (Price K. 1999) for DE and (Zelinka Ivan, 2004) for SOMA.

The control parameter settings have been found empirically and are given in Table 1 (SOMA), Table 2 (DE) and Table 3 (GA). The main criterion for this setting was to keep the same setting of parameters as much as possible for all simulations and of course the same number of cost function evaluations as well as population size (parameter PopSize for SOMA and GA, NP for DE). Individual length represents number of optimised parameters (number of pinning sites, values...).

Table 1: SOMA setting for case studies A, B, C and D

	A	B	C	D
PathLength	3	3	3	3
Step	3	3	3	3
PRT	0.1	0.1	0.1	0.1
PopSize	20	20	20	20
Migrations	10	10	10	10
MinDiv	0.1	0.1	0.1	0.1
Individual Length	1	10	10	20
CF Evaluations	1900	1900	1900	1900

Table 2: DE setting for case studies A, B, C and D

	A	B	C	D
NP	20	20	20	20
F	0.8	0.8	0.8	0.8
CR	0.2	0.2	0.2	0.2
Generations	100	100	100	100
Individual Length	1	10	10	20
CF Evaluations	2000	2000	2000	2000

Table 3: GA setting for case studies A, B, C and D

	A	B	C	D
PopSize	20	20	20	20
Mutation	0.4	0.4	0.4	0.4
Generations	100	100	100	100
Individual Length	1	10	10	20
CF Evaluations	2000	2000	2000	2000

### 3 EXPERIMENTAL RESULTS

All three algorithms (SOMA, DE, GA) have been applied 50 times in order to find the optimum of all CML DCC problems. The primary aim of this comparative study is not to show which algorithm is better and worst, but to show that evolutionary DCC (EDCC) can be used for different problems of spatiotemporal chaos control based at least on CML.

The outputs of all simulations are depicted in Figures 4-31. Figures 4-24 show the results of all 50 simulations for each case study. Figures 25-27 show a mutual comparison of algorithm performance in the point of view of the number of estimated pinning sites.

#### 3.1 Case study A - Pinning Value Estimation

In this case study SOMA, DE and GA were used to estimate the pinning value for CML. Pinning sites were a priori set according to (Hu G., Xie F., Xiao J., Yang J., Qu Z., 1999). Estimated pinning value was used for all a priori defined pinning sites (each odd). The simulations were repeated 50 times and from the last population in each simulation the best was recorded, together with the worst and the average result (individual). All fifty triplets (best, worst and average) were used to create Figures 4 - 6. For the verification of the results, the dependency of cost values (according to (2)) on pinning values was calculated and is depicted in Figure 3. Optimal pinning values are in the interval 2.1 – 3.6 (cost value is 0, i.e. minimal difference between CML behavior and desired behavior). Based on Figures 4, 5 and 6 it can be stated that in all simulations suitable pinning values were estimated because according to (Hu G., Xie F., Xiao J., Yang J., Qu Z., 1999) a suitable pinning value (equal to 2) was used and here in each simulation the best values are around 2.5 and the average values around 2.9.

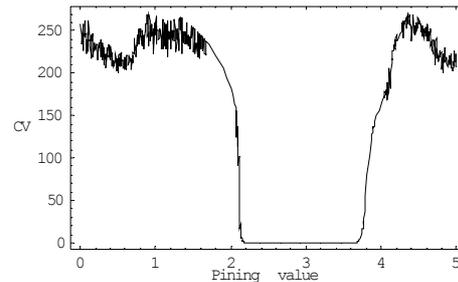


Figure 3 Dependence of costvalue on pinning values

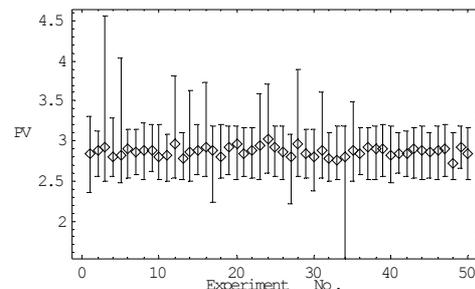


Figure 4 Estimated pinning values by SOMA

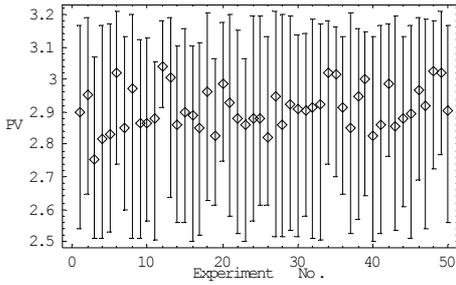


Figure 5 Estimated pinning values by DE

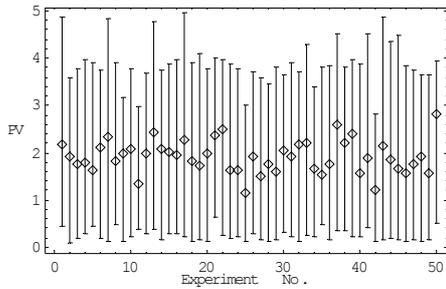


Figure 6 Estimated pinning values by GA

### 3.2 Case study B - Pinning Sites Position Estimation

Case study B was designed based on results from the previous case study. SOMA, DE and GA were used to estimate the pinning sites for CML. The pinning values were a priori set equal to 2 for all estimated sites according to (Hu G., Xie F., Xiao J., Yang J., Qu Z., 1999). The simulations were again repeated 50 times and the best solution (pinning sites) from each simulation was used to create Figure 7 - 12. Columns on Figures 7, 9 and 11 represent the best solution from an actual simulation and the black squares in the columns represent active input for pinning (white squares represent unused inputs, i.e. inputs without pinings).

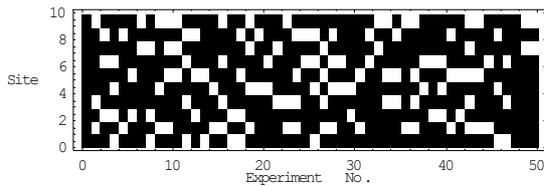


Figure 7 Estimated pinning sites by SOMA

For better visibility of the results achieved, histograms (Figure 8, 10 and 12) were also created, showing the frequency of estimated pinning sites. As the figures show, it can be stated that in two cases (SOMA, DE) there were redundant pinning sites, because according to (Hu G., Xie F., Xiao J., Yang J., Qu Z., 1999) certainly it is enough if pinings are at each odd (or even) site (equation, pinning input). GA has shown better results comparing to SOMA and DE. In order to improve SOMA and DE results, case study C was designed.

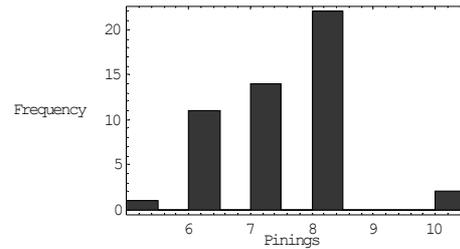


Figure 8 Histogram of estimated pinning sites by SOMA

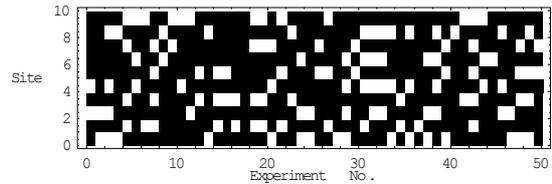


Figure 9 Estimated pinning sites by DE

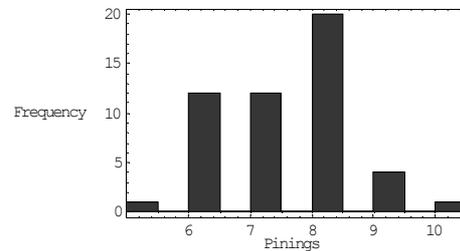


Figure 10 Histogram of estimated pinning sites by DE

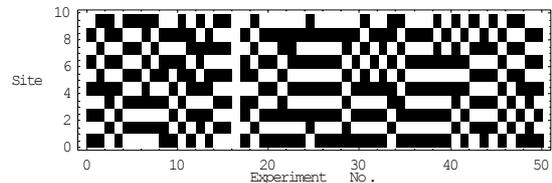


Figure 11 Estimated pinning sites by GA

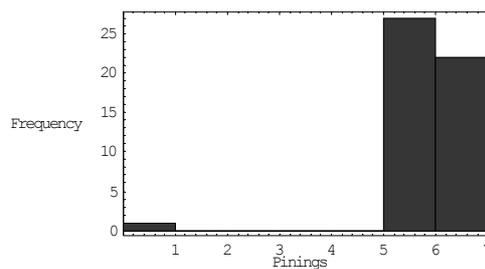


Figure 12 Histogram of estimated pinning sites by GA

### 3.3 Case study C - Minimal Pinning Sites Position Estimation

This case study was designed to improve the previous results from case study B. Cost function (2) was modified to (3) as described on section 2.2. All other conditions were kept the same. Again, the same types figures were created (pinning sites - Figure 13, 15 and 17, histograms - Figure 14, 16 and 18). Figures 13, 15 and 17 show surprisingly nice structures of pinning sites. SOMA and DE had found in all 50 simulations pinning

sites, which are on odd sites or on even sites, which correspond with the results from (Hu G., Xie F., Xiao J., Yang J., Qu Z., 1999). Because CML used here had so called cyclic boundary ((Hu G., Xie F., Xiao J., Yang J., Qu Z., 1999),  $x(L+1) = x(1)$ ), then it can be stated that all these solutions are equal. Only GA has failed in two cases (see two white columns - experiments (12,14) on Figure 17) Based on the histograms in Figures 14, 16 and 18, it can also be concluded that all algorithms demonstrated the same level of performance.

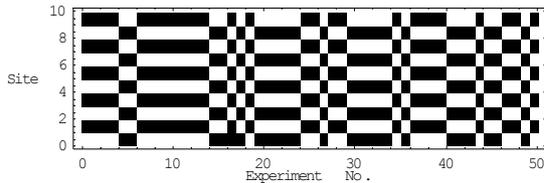


Figure 13 Estimated pinning sites by SOMA

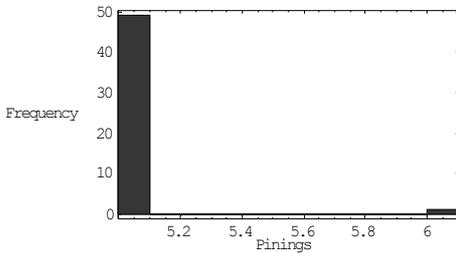


Figure 14 Histogram of estimated pinning sites by SOMA

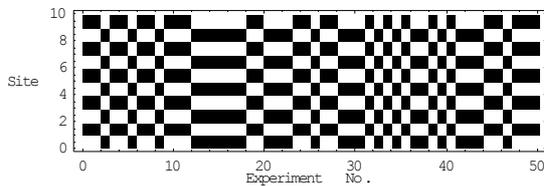


Figure 15 Estimated pinning sites by DE

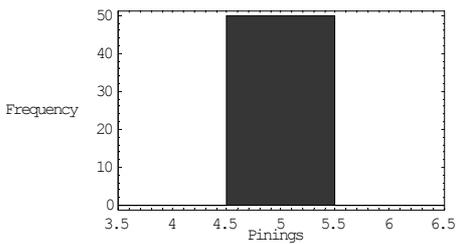


Figure 16 Histogram of estimated pinning sites by DE

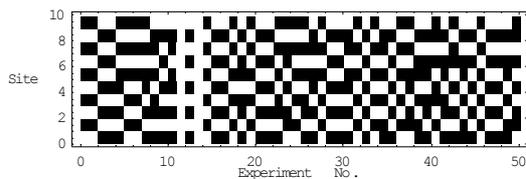


Figure 17 Estimated pinning sites by GA

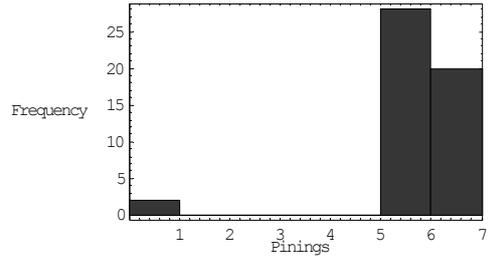


Figure 18 Histogram of estimated pinning sites by GA

### 3.4 Case study D - Minimal Pinning Values and Sites Position Estimation

The last case study was dedicated to the estimation of minimal numbers of pinning sites and different pinning values. In contrast to the previous case studies, for each estimated pinning site a unique pinning value was estimated here. All simulations were repeated under the same conditions as in the case study C and the same kind of figures (Figure 19, 21 and 23, Figure 20, 22 and 24) was created. Figures 19, 21 show again pinning patterns demonstrating that SOMA and DE had found more times the same solution. Unexpectedly GA (Figure 23) has failed from pinning patterns point of view as well as in the four cases (experiment 5, 6, 14, 29).

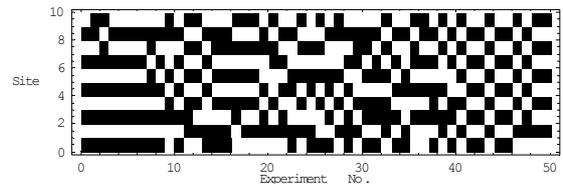


Figure 19 Estimated pinning sites by SOMA

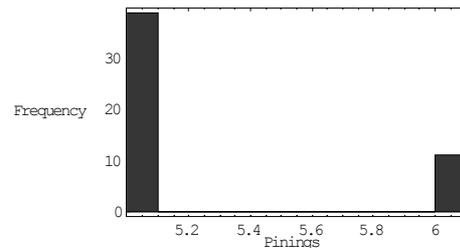


Figure 20 Histogram of estimated pinning sites by SOMA

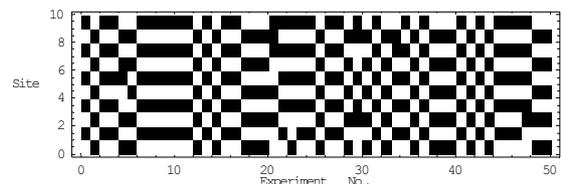


Figure 21 Estimated pinning sites by DE

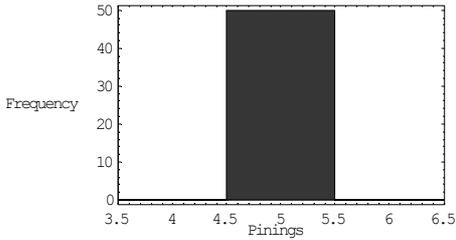


Figure 22 Histogram of estimated pinning sites by DE

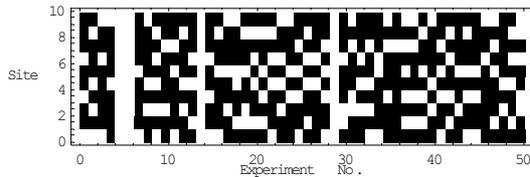


Figure 23 Estimated pinning sites by GA

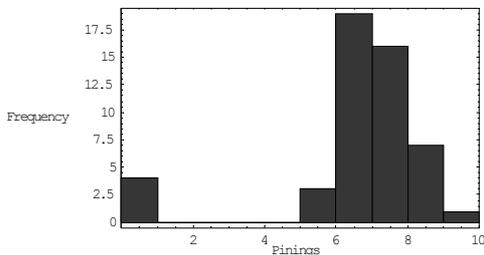


Figure 24 Histogram of estimated pinning sites by GA

#### 4 Selected mutual comparison

The problem complexity has increased in all four case studies. Based on data from all simulations a comparison can be done from pinning sites point of view. As is depicted in Figure 25 - 27 all three algorithms are comparable in performance (with small deviations). It is also visible that changes in the cost functions can significantly improve estimated solutions (case C / D).

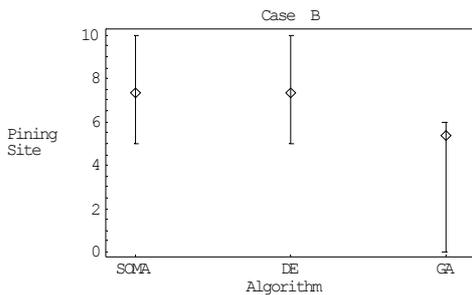


Figure 25 Mutual comparison of pinning sites - case B

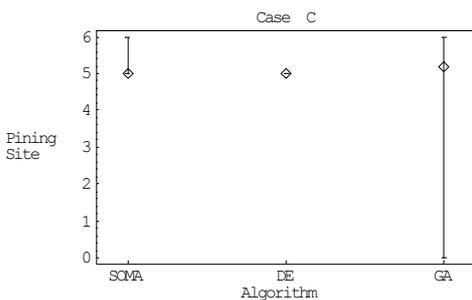


Figure 26 Mutual comparison of pinning sites - case C

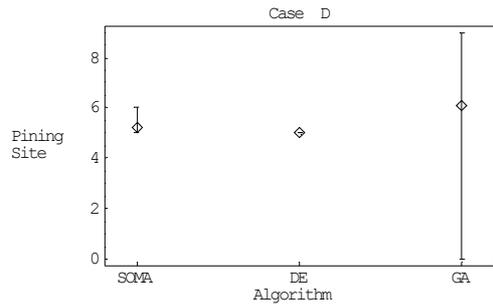


Figure 27 Mutual comparison of pinning sites - case D

In order to verify that the estimated pinning sites and pinning values really stabilize behaviour of CML, 600 figures was generated (4 cases  $\times$  (50 + 50 + 50 simulations)) on data from all simulations (including a few above mentioned wrong solutions). In all 594 = 600 - 6 simulations CML was stabilised on the desired behavior. For comparison with deterministic CML control (Hu G., Xie F., Xiao J., Yang J., Qu Z., 1999) Figure 28 was created and as a typical example of DCC and examples of ECDCC are also depicted here in Figures 29 - 31.

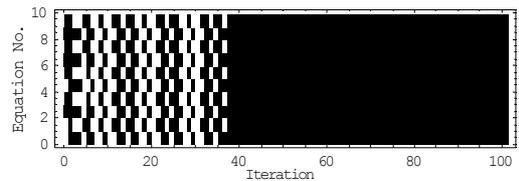


Figure 28 Control by deterministic control law

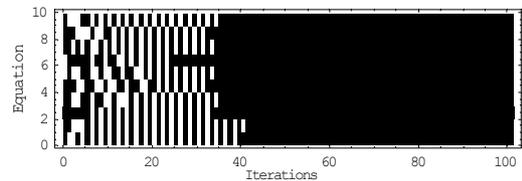


Figure 29 Control by SOMA

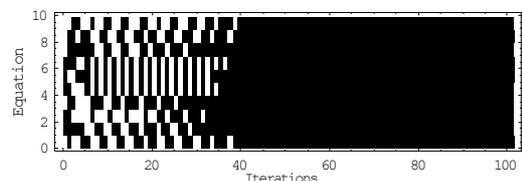


Figure 30 Control by DE

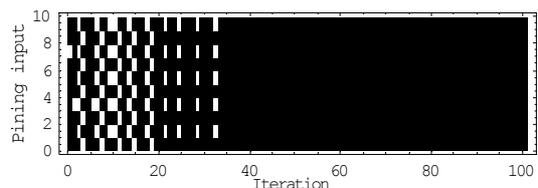


Figure 31 Control by GA

## 5 CONCLUSIONS

The method of evolutionary deterministic chaos control described here is relatively simple, easy to implement and easy to use. Based on its principles and its possible universality (it was tested with 3 evolutionary algorithms – SOMA, DE and GA) it can be stated that evolutionary deterministic chaos control is capable to solve 1D class CML deterministic chaos control problems.

The main aim of this paper was to show how various CML control problems can be solved by means of evolutionary algorithms. Evolutionary deterministic chaos control was used here in four basic comparative simulations. Each comparative simulation was 50 times repeated and all 600 results (50 simulations for each algorithm and for each problem) were used to create figures for performance evaluation of evolutionary deterministic chaos control.

For the comparative study three algorithms were used - DE (Price K. 1999), SOMA (Zelinka Ivan, 2004) and GA. They were chosen to show that evolutionary deterministic chaos control can be regarded as a “blackbox” method and that it can be implemented using arbitrary evolutionary algorithms. As a conclusion the following statements are presented:

1. **Reached results.** Based on results reported in Figures 4 - 31 it can be stated that all simulations give satisfactory results and thus evolutionary deterministic chaos control is capable of solving this class of problems.
2. **Anomaly.** In all cases the quality of reached results has increased. Only in the case of GA an anomaly was observed, in case C and D. In case D the worst quality of pinning site “patterns” was observed comparing to case C. Also 2 simulations (C) and 4 simulations (D) have totally failed. It was probably caused by the fact that a low amount of cost function evaluations was used for EDCC and GA probably needs more Generations or / and bigger population size.
3. **Mutual comparison.** When comparing all algorithms, then it is visible that all three algorithms give good results. Parameter settings for both algorithms were based on a heuristically approach and thus there is a possibility that better settings can be found there.

Future research is one of the key activities in the frame of evolutionary deterministic chaos control. According to all results obtained during time it is planned that the main activities would be focused expanding of this comparative study for simulated annealing and 2D CML cases. More complicated patterns are also planned to be controlled like T1S2, etc.

## ACKNOWLEDGEMENT

This work was supported by grant No. MSM 7088352101 of the Ministry of Education of the Czech Republic and by grants of the Grant Agency of the Czech Republic GACR 102/03/0070.

## REFERENCES

- Greboki C., Lai Y.C. 1999, Controlling Chaos, In: Schuster H.G., Handbook of Chaos Control, Wiley-Vch, ISBN 3-527-29436-8, 1999
- Guanrong Chen, 2000, Controlling Chaos and Bifurcations in Engineering Systems, CRC Press, ISBN 0-8493-0579-9, 2000
- Hendrik Richter & Kurt J. Reinschke, 2000: Optimization of local control of chaos by an evolutionary algorithm. *Physica D144* (2000), 309-334.
- Hendrik Richter & Kurt J. Reinschke 2000a: Optimization of local control of chaos by an evolutionary algorithm. *Physica D144* (2000), 309-334.
- Hendrik Richter, 2002: An evolutionary algorithm for controlling chaos: The use of multi-objective fitness functions. In: *Parallel Problem Solving from Nature-PPSN VII*. (Eds.: Merelo Guervós, J.J.; Panagiotis, A.; Beyer, H.G.; Fernández Villacanas, J.L.; Schwefel, H.P.), Lecture Notes in Computer Science, Vol. 2439, Springer-Verlag, Berlin Heidelberg New York, 2002, 308-317
- Hilborn R.C.1994, Chaos and Nonlinear Dynamics, Oxford University Press, ISBN 0-19-508816-8, 1994
- Hu G., Xie F., Xiao J., Yang J., Qu Z., 1999, Control of Patterns and Spatiotemporal Chaos and its Application, In: Schuster H.G., Handbook of Chaos Control, Wiley-Vch, ISBN 3-527-29436-8, 1999
- Just W., 1999, Principles of Time Delayed Feedback Control, In: Schuster H.G., Handbook of Chaos Control, Wiley-Vch, ISBN 3-527-29436-8, 1999
- Ott E., Greboki C., Yorke J.A., 1990, Controlling Chaos, *Phys. Rev. Lett.* 64, 1196, (1990)
- Price K. 1999, An Introduction to Differential Evolution, in *New Ideas in Optimization*, D. Corne, M. Dorigo and F. Glover, Eds., s. 79–108, McGraw-Hill, London, UK, 1999. ISBN 007-709506-5
- Schuster H.G., 1999, Handbook of Chaos Control, Wiley-Vch, ISBN 3-527-29436-8, 1999
- Zelinka I., Nolle L., 2005, „Plasma Reactor Optimizing Using Differential Evolution“, In: Price K.V., Lampinen J., Storn R., *Differential Evolution* :

Global Optimization for Scientists and Engineers,  
Springer-Verlag, 2005

Zelinka Ivan, 2004, SOMA – Self Organizing Migrating  
Algorithm“,Chapter 7, 33 p. in: B.V. Babu, G.  
Onwubolu (eds), New Optimization Techniques in  
Engineering, Springer-Verlag, ISBN 3-540-20167X

Zelinka Ivan, 2005, Investigation on Evolutionary  
Deterministic Chaos Control, IFAC, Prague 2005, in  
print

## **AUTHOR BIOGRAPHIES**



**IVAN ZELINKA** was born in Czech Republic, and went to the Technical University of Brno, where he studied technical cybernetics and obtained his degree in 1995. He obtained Ph.D. degree in technical cybernetics in 2001 at Tomas Bata University in Zlin. Now he is an associated profesor (artificial intelligence, theory of information) and head of department. His e-mail address is: [zelinka@ft.utb.cz](mailto:zelinka@ft.utb.cz) and his Web-page can be found at <http://www.ft.utb.cz/people/zelinka>.

# OPTIMIZATION AND CONTROL OF BATCH REACTOR BY EVOLUTIONARY ALGORITHMS

Roman Senkerik

Ivan Zelinka

Department of Applied Informatics

Tomas Bata University

Nad Stranemi 4511, Zlin 760 05, Czech Republic

E-mail: [zelinka@ft.utb.cz](mailto:zelinka@ft.utb.cz) , [senkerik@ft.utb.cz](mailto:senkerik@ft.utb.cz)

## ABSTRACT

This work deals with using a method of artificial intelligence, namely the evolutionary algorithm SOMA, for static optimization of a chemical batch reactor, for the purpose to improve its behavior in uncontrolled state and predictive control. The importance of this problem is increasing with growing demand for special products made in batch reactors. The optimizations has been performed in several ways, each one for another set of reactor parameters or another cost function. The optimization gave the best solution, which improved the performance of the reactor. All the important results of each optimization are discussed continuously and at the end of the paper too. The optimized reactor was used in a simulation with predictive control by the evolutionary algorithms and excellent results were achieved.

## INTRODUCTION

Chemical industry produces a whole range of products through physical and chemical reactions. Successful mastering of the chemico-technological process requires its quantitative and qualitative assessment. This is particularly necessary in the cases where an introduction of automatic control systems of technological procedures is under consideration. In order to prevent computer-control modernization of existing and future processes becoming an end in itself, it is necessary to proceed in individual stages.

The most important stage is an analysis of the manufacturing system, which in most cases comprises simulating calculations based on a realistic understanding of physico-chemical mechanisms through which initial raw materials are converted into the required product. These calculations then reveal key points of the technological process which by their nature enforce changes and adjustments of the technological procedure so that potential automation produces the required effect, i.e. a quality product at minimal production costs.

Generally it may be stated that key technological points are chemical reactors. Designing optimal reactor parameters including control constitutes one of the most complex tasks in process engineering. The situation is particularly complicated by the fact that the precise mechanism of chemical reaction kinetics is very often unknown. For this reason it is necessary to carry out

extensive measurements of input and output concentration dependencies of components on time, temperature, etc. These quite complex kinetic models, verified of course by experimental measurements, are subsequently linearized employing various methods, which enables to apply already developed control methods to these models.

Disregarding the petrochemical field, attention of chemical industry is mostly given to raw materials for producing macromolecular materials, i.e. plastics. High-molecular compounds arise through two types of reactions, polymerization and polycondensation. What is essential for designing algorithms controlling such reactions is the fact that the majority of these reactions are exothermic, i.e. releasing heat. From the viewpoint of economy, productivity of the chemical reactor is usually demanded to be in a certain respect maximal, and final-product quality to be as required. Reactor productivity depends on reaction rate, which usually increases exponentially with increasing temperature.

At first sight it seems that an exothermal character of a reaction is very advantageous. This conclusion is true to a certain extent, nevertheless, as with all, it also has limitations regarding both safety and quality (e.g. product quality may be degraded with increased temperature).

This is particularly true if the main reaction is accompanied with undesired side reactions whose rates also increase exponentially with temperature. Hence, it is apparent that the principal controlled quantity in exothermal reactions is temperature of reaction mixture. Therefore, models presented in this work are based on enthalpic balances enabling relevant simulations.

## Characteristics of batch processes

The optimization of batch processes has attracted attention in recent years (Aziz et al. 2000), (Silva et al. 2003) because, in the face of growing competition, it is a natural choice for reducing production costs, improving product quality, meeting safety requirements and environmental regulations. Batch and semi-batch processes are of considerable importance in the fine chemicals industry. A wide variety of special chemicals, pharmaceutical products, and certain types of polymers are manufactured in batch operations. Batch processes are typically used when the production volumes are

low, when isolation is required for reasons of sterility or safety, and when the materials involved are difficult to handle. In batch operations, all the reactants are charged in a tank initially and processed according to a pre-determined course of action during which no material is added or removed. In semi-batch operations, a reactant may be added with no product removal, or a product may be removed with no reactant addition, or a combination of both. From a process systems point of view, the key feature that differentiates continuous processes from batch and semi-batch processes is that continuous processes have a steady state, whereas batch and semi-batch processes do not (Srinivasan 2000 et al. 2002a and 2000b)

## DESCRIPTION OF REACTOR

This work used a mathematical model of reactor shown in Fig.1. This chemical reactor was designed for chemico-technological process of leather manufacturing waste processing, which is a huge ecologic problem world-wide. Successful determination of the reactor optimal parameters and improved control of the whole process will help to elaborate a propose of technological plant, which will be able to satisfy economical and quality requirements.

The reactor has two physical inputs (one for chemical substances and one for cooling medium) and one output (cooling medium). Equations (1) and (2) represent the mathematical model of the reactor.

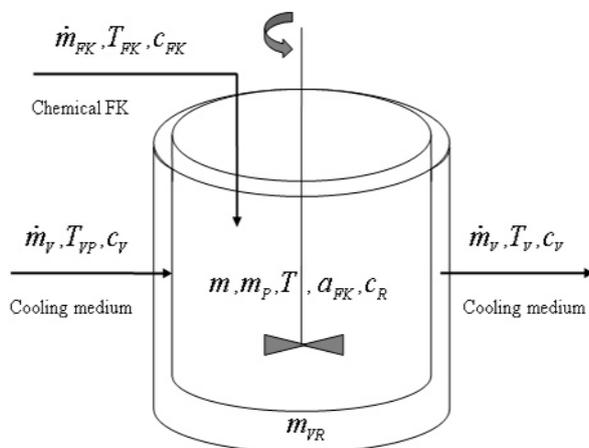


Fig.1 Scheme of reactor

Chemical FK flows into the reactor through the input denoted “Chemical FK”, with parameters temperature  $T_{FK}$ , mass flow rate  $\dot{m}_{FK}$  and specific heat  $c_{FK}$ . The coolant flows into reactor through the second input denoted “Cooling medium”, which is usually water of temperature  $T_{VP}$ , mass flow rate  $\dot{m}_V$  and specific heat  $c_V$ .

Cooling medium flows through jacket inner space of reactor, with volume related to mass flow rate  $\dot{m}_{VR}$ , and

flows out through second output, with parameters mass flow rate  $\dot{m}_V$ , temperature  $T_V$  and specific heat  $c_V$ .

At the beginning of the process there is an initial batch inside the reactor with parameter mass  $m_p$ . The chemical FK is then added to this initial batch, so the reaction mixture inside the reactor has total mass  $m$ , temperature  $T$  and specific heat  $c_R$  and also contains partially unreacted portions of chemical FK described by parameter concentration  $a_{FK}$ .

This technique partially allows controlling the temperature of reaction mixture by the controlled feeding of the input chemical FK.

The main objective of optimization is to achieve the processing of large amount of chemical FK in very short time. An exothermal reaction described by relationships (1) – (3) takes place in the reactor.

In general, this reaction is highly exothermal. Hence, the most important parameter is the temperature of the reaction mixture. **This temperature must not exceed 100°C** because of safety aspects and quality of processing.

Designing the reactor was based on standard chemical-technological methods and gives a proposal of reactor physical dimensions and parameters of chemical substances. These values are called in this work **expert parameters**. The objective of this part of the work was to perform a simulation and optimization of the given reactor.

## Non-linear model of reactor

Description of the reactor applied a system of four balance equations (2). The first expresses a mass balance of reaction mixture inside the reactor, the second a mass balance of the chemical FK, and the last two formulate enthalpic balances, namely balances of reaction mixture and cooling medium. Equation (1), in which (2) is represented by term “k”, is written out here for simplified notation of basic equations.

$$\dot{m}_{FK} = m'[t] \quad (1)$$

$$\dot{m}_{FK} = m[t] a'_{FK}[t] + k m[t] a_{FK}[t]$$

$$\begin{aligned} \dot{m}_{FK} c_{FK} T_{FK} + \Delta H_r k m[t] a_{FK}[t] = \\ = K S (T[t] - T_V[t]) + m[t] c_R T'[t] \end{aligned}$$

$$\dot{m}_V c_V T_{VP} + K S (T[t] - T_V[t]) = \dot{m}_V c_V T_V[t] + \dot{m}_{VR} c_V T'_V[t]$$

$$k = A e^{-\frac{E}{RT[t]}} \quad (2)$$

After modification into standard form, balance equations are obtained in form (3)

$$m'[t] = \dot{m}_{FK} \quad (3)$$

$$a'_{FK}[t] = \frac{\dot{m}_{FK}}{m[t]} - A e^{-\frac{E}{RT[t]}} a_{FK}[t]$$

$$T''[t] = \frac{\dot{m}_{FK} c_{FK} T_{FK}}{m[t] c_R} + \frac{A e^{-\frac{E}{RT[t]}} \Delta H_r a_A[t]}{c_R} - \frac{K S T[t]}{m[t] c_R} + \frac{K S T_V[t]}{m[t] c_R}$$

$$T'_V[t] = \frac{\dot{m}_V T_{VP}}{m_{VR}} + \frac{K S T[t]}{m_{VR} c_V} - \frac{K S T_V[t]}{m_{VR} c_V} - \frac{\dot{m}_V T_V[t]}{m_{VR}}$$

Parameters for this reactor and initial conditions ( $a_{FK0}$ ,  $T_{V0}$ ,  $T_0$ ,  $m_0, \dots$ ) have been specified by expert, giving both physical dimensions as well as parameters of individual chemical substances. These were used to simulate the behavior of this reactor. It is evident from Fig.2, that reactor behavior in compliance with parameters thus specified by expert is not quite satisfactory during the processing of one batch. The duration of the processing of one batch is given by the time from the start of the process till the reaction mixture cools down to its initial temperature  $T_0$ . The behavior of the reactor was simulated using five different values of batching the chemical FK into the reaction mixture (from the range of 0 – 3 kg.s<sup>-1</sup>).  $\dot{m}_{FK} = (0,05$  (violet) ; 0,1 (green) ; 0,5 (blue) ; 1 (red) ; 3 (black)) kg.s<sup>-1</sup>.

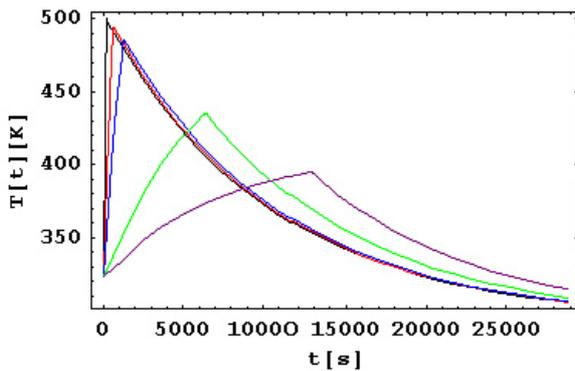


Fig.2 Behavior of the reactor set up by means of expert parameters

Facts that follow from the simulation of this strongly exothermal reaction are as follows:

1. Duration of one batch cycle is approx 25 000 sec
2. Temperature of reaction mixture  $T$  markedly exceeds critical limit of 100°C (373,15 K), even in case of simulation with lowest value of batching chemical FK into the reactor.

As these are very serious, multifold repeated static optimizations by means of SOMA algorithm has been performed, and this led to finding the reactor parameters with which the reactor behavior is much more acceptable.

### Optimizing algorithm

Optimization algorithms are a powerful tool for solving many problems in engineering practice. They are usually applied when solving a given problem analytically is impossible or unrealistic. When properly implemented, they can be employed in such manner that even frequent user's intervention in work of the respective plant where they are applied is unnecessary. This contribution consider the use of optimization algorithm SOMA. Numerous results have shown SOMA is a viable algorithm that can be used to solve many practical as well as theoretical problems. (Zelinka 2004)

### STATIC OPTIMIZATION OF REACTOR

The reactor described above, in the original set-up, gave unsatisfactory results. To improve reactor behavior, static optimization was performed using the algorithm SOMA. The optimization was performed by the following three basic methods.

1. Optimization of batching value  $\dot{m}_{FK}$
2. Optimization of batching value  $\dot{m}_{FK}$  together with process parameters of the cooling medium
3. Optimization that covered all previous optimized parameters, and including also optimization of reactor geometry and cooling area.

Each optimization was repeated ten times just in case, and separate figures present time plots of all important variables collected together in order to emphasize optimizing robustness, including courses of time evolutions from all ten simulations. The behavior of the best reactor was shown with the last optimization. A discussion on particular optimization cases follows in the sections below.

### Optimization of batching value

In this optimization the point was to minimize the area arising as a difference between the required and real temperature profile of the reaction mixture in a selected time interval, which was the duration of a batch cycle. The required temperature was 97°C (370,15 K). The cost function that was minimized is given in (5), minimizing term which limits the maximum mass of one batch in (6)

$$f_{\text{cost}} = \sum_{t=0}^t |w - T[t]| \quad (5)$$

$$m[t] \leq m_{\text{max}} \quad (6)$$

The simulation results are presented in Fig.3. The optimization algorithm obviously found the best value for batching so that the critical temperature was not exceeded. Nevertheless, the duration of the batch cycle was not shortened so the performance of the reactor could not be improved without more effective cooling of the reaction mixture. Thus the next step was the second optimization covering process parameters of the cooling medium, but preserving the geometry of the reactor.

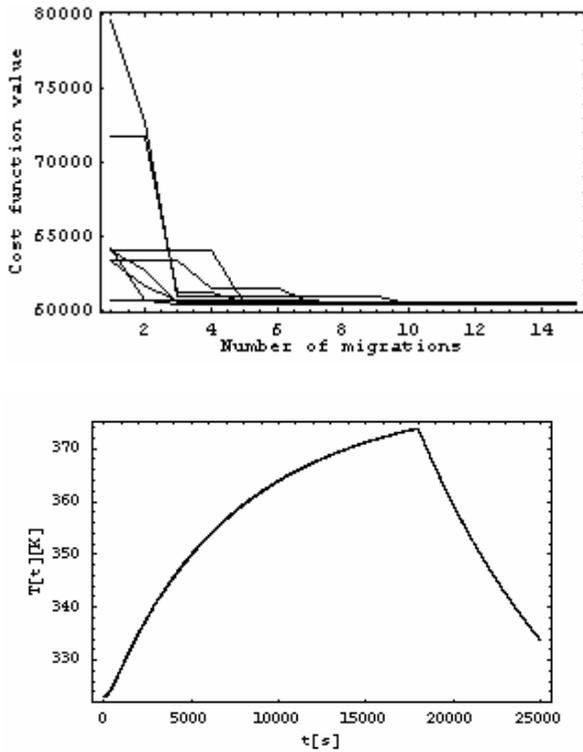


Fig.3 Results of optimization of batching value

### Optimization of batching value and process parameters of cooling medium

Optimization proceeded in the second case with the parameters that are shown in Tab.1.

The actual functional that was minimized was the same as in the first case (5), minimizing term also the same as in previous case, which limits the maximum mass of one batch, is presented in (6). Another restriction terms were the ranges of optimized parameters (see Tab. 1.).

The results of simulations are presented in Fig.4. As can be seen, this optimization again found a good value for batching so that the critical temperature was not exceeded. Nevertheless, the opposite consequence was reached than required shortening duration of batch cycle and increasing the performance of reactor. The optimization found a relatively high value of the input cooling medium temperature. That essentially means

that chemical reaction was started faster and required temperature was reached in shorter time. This fact corresponds to the selected minimization functional (5), but on the other side cooling of reaction mixture was very slow after filling up the reactor and thus termination of reaction. As a result of these facts the third optimization included the optimization of reactor geometry and the minimized functional had to be changed to achieve reasonable balance between the demands of fast start of chemical reaction, critical temperature and fast cooling to initial temperature and hence to increase the performance of reactor.

Tab.1 Optimized reactor parameters

Parameter	Range
$\dot{m}_{FK}$ [kg.s <sup>-1</sup> ]	0 – 640
$T_{VP}$ [K]	253,15 – 323,15
$\dot{m}_V$ [kg.s <sup>-1</sup> ]	0-10

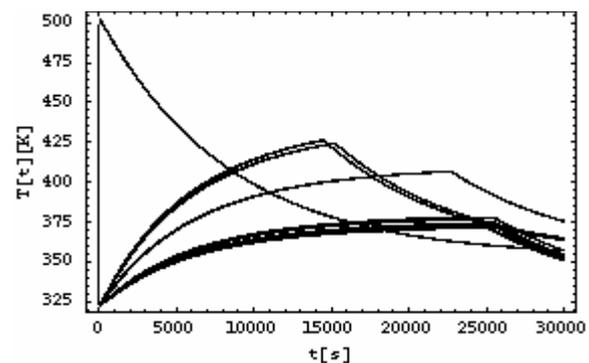
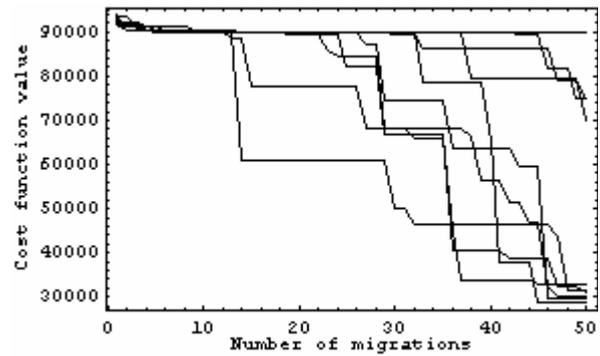


Fig.4 Results of optimization of batching value and process parameters of cooling medium

## Optimization of process parameters and the reactor geometry

This optimization proceeded with parameters that are shown in Tab.2.

Tab.2 Optimized reactor parameters

Parameter	Range
$\dot{m}_{FK}$ [kg.s <sup>-1</sup> ]	0 – 640
$T_{VP}$ [K]	273,15 – 323,15
$\dot{m}_V$ [kg]	0 – 10
$m_{VR}$ [kg]	220
$r$ [m]	0,5 – 2,5
$h$ [m]	0,5 – 2,5

As a result of last two simulations the actual functional that was minimized was changed to (7). It was divided into three time intervals and also two penalizations (8) and a special part were added. This special part ensures rapid reaction of the whole batch of chemical FK hence very low value of concentration  $a_{FK}$  of partly unreacted portions of chemical FK in reaction mixture.

The first penalization helps to find solutions in which the temperature of reaction mixture cools down fast to its initial state, and the process duration is shortened. The second corresponds to the critical temperature condition.

$$f_{cost} = \sum_{t=0}^{t_1} |w - T[t]| + \sum_{t=0}^{t_1} a_{FK}[t] + pen.1 + pen.2 \quad (7)$$

$$pen.1 = \begin{cases} 0 & \text{Max}(T[\tau]) \leq 323,15 \\ 50000 & \text{else} \end{cases} \quad (8)$$

for  $\tau \in \langle t_2, t_3 \rangle$

$$pen.2 = \begin{cases} 0 & \text{Max}(T[\tau]) \leq 373,15 \\ 50000 & \text{else} \end{cases}$$

for  $\tau \in \langle 0, t_3 \rangle$

where the time intervals were set for example as:

$$t_1 = 15000 \text{ s}; t_2 = 20000 \text{ s}; t_3 = 25000 \text{ s}$$

The minimizing term, presented in (6), is also the same as in the previous case, which limits the maximum mass of one batch. Another restriction terms were the ranges of optimized parameters (see Tab. 2.). Moreover, many parameters were interrelated due to the optimization of the reactor geometry. The relation between  $m$ ,  $m_{FK}$  and  $m_p$  was given by relationship (9).

$$m = m_p + m_{FK} \quad (9)$$

The stoichiometric ratio is given by (10).

$$m_p = 2,82236m_{FK} \quad (10)$$

The relationship between the optimized volume of reactor and the mass of added chemical FK is given by (11). Then substituting to (10) gives the mass of initial batch in reactor.

$$m_{FK} = \frac{\rho_p \rho_{FK} V}{2,82236 \rho_{FK} + \rho_p} \quad (11)$$

The results of this optimization are shown in Figs.5 and 6. Fig.6 demonstrates the behavior of a selected reactor where predictive control was applied using algorithm SOMA. From these results it is obvious that the temperature of reaction mixture moderately exceeded the critical value, but this was a simulation of uncontrolled reactor. This can be corrected by a quality control of the batch process. Another fact not to be neglected is the shortened duration of process and the improvement of reactor performance compared with the reactor set up by an expert (see Tab.3).

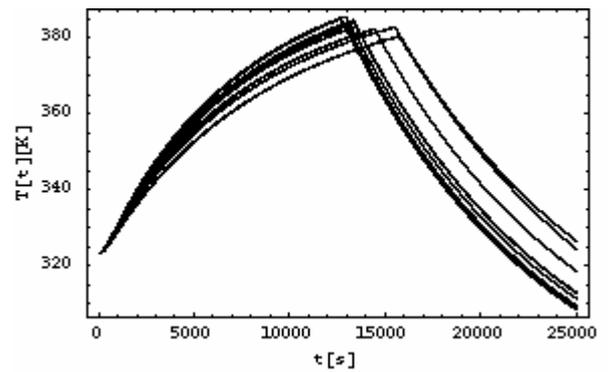
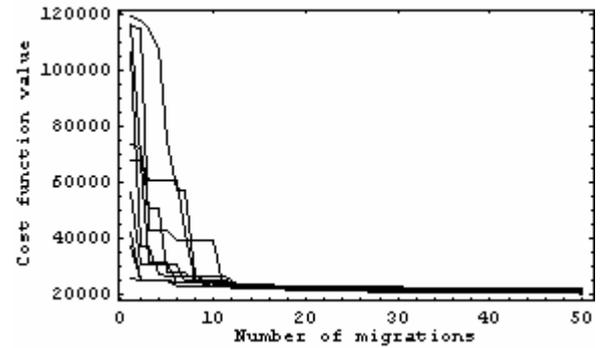


Fig.5 Results of optimization of process parameters and reactor geometry

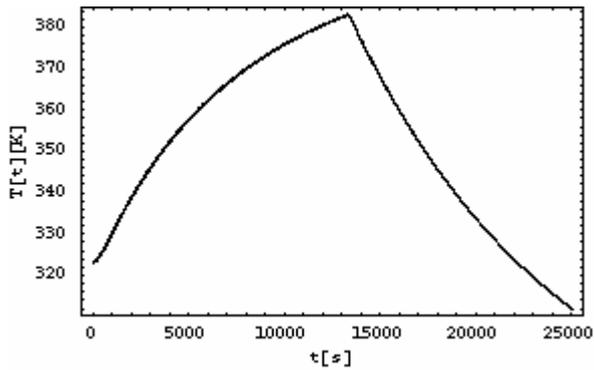


Fig.6 Reaction mixture temperature profile for the best optimized reactor

## CONCLUSION

This report describes a chemical reactor (Fig.1). After a general introduction and description of chemistry of the relevant reaction, this part discusses a nonlinear model constructed on the basis of physic-mathematical analysis (equations (1–3) including an illustrations of reactor behavior based on parameters set up by an expert (see Fig.2)).

Based on the obtained results, it may be claimed that the behavior of an uncontrolled reactor gives quite unsatisfactory results, which may be overcome through static optimization of a given reactor. The quality of results produced by the optimizations depends not only on the problem being solved but also on how a given functional is defined. Its construction may comprise not only optimization of a basic criterion but also optimization of subcriteria capable of improving optimization quality.

Basic optimizations presented here were based on a relatively simple functional. Unless the experimenter is limited by technical issues when searching for optimal parameters, there is no problem in defining more complex functional including as subcriteria e.g., stability, economic costs, time-optimal criteria, controllability, etc. or their arbitrary combinations. However, complexity of such functional indirectly implies the application of advanced software such as Mathematica or Matlab.

The advantage of parallel optimization (in the context of evolutionary algorithms – i.e. “simultaneously” seeking X possible solutions and selecting the best) lies in the fact that unless an optimal solution is found, one usually obtains suboptimal solutions, which are usually not too distant from the true optimum.

Differences between both reactors are best to be seen in Tab.3. The first part shows the parameters of reactor designed by an expert, and the second part shows the parameters obtained through static optimization. In Tab.3, the internal radius of reactor is expressed in parameter  $r$  and is related to cooling area “S” The parameter  $d$  represents the distance between the outer and inner jacket and the parameter  $h$  represents the

height of the reactor. The meaning of the other parameters is quite obvious from the table.

Tab.3 Difference between reactor designed by an expert and the optimized reactor

Parameter	Expert setting	Optimized
$\dot{m}_{FK}$ [kg.s <sup>-1</sup> ]	0 - 3	0,1021
$T_{VP}$ [K]	293,15	274,58
$\dot{m}_V$ [kg]	1	4,67
$m_{VR}$ [kg]	220	1159,7
$d$ [m]	0,03	0,096
$r$ [m]	0,78	1,017
$h$ [m]	1,11	1,382
$S$ [m <sup>2</sup> ]	7,35	12,08
$V$ [m <sup>3</sup> ]	2,12	4,49
$m_p$ [kg]	1810	3842,4
$m_{FK}$ [kg]	640	1361,4

Finally, on the basis of presented results it may be stated that the reactor parameters has been found by optimization which demonstrates performance superior to that of reactor set up by an expert.

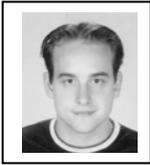
## ACKNOWLEDGEMENT

This work was supported by grant No. MSM 26500014 of the Ministry of Education of the Czech Republic and by grants of the Grant Agency of the Czech Republic GACR 102/03/0070.

## REFERENCES

- Aziz N., Hussain M.A., Mujtaba I.M. (2000). Performance of different types of controllers in tracking optimal temperature in batch reactors. *Computers and Chemical Engineering*, Vol. 27, p. 1069-1075.
- Silva C.M., Biscaia E.C. (2003). Genetic algorithm development for multi-objective optimization of batch free-radical polymerization reactors. *Computers and Chemical Engineering*, Vol. 27, p. 1329-1344
- Srinivasan B., Palanki S., Bonvin D. (2002). Dynamic optimization of batch processes I. Characterization of the nominal solution. *Computers and Chemical Engineering*, Vol. 27, p. 1-26.
- Srinivasan B., Palanki S., Bonvin D. (2002). Dynamic optimization of batch processes II. Role of Measurement in handling uncertainly. *Computers and Chemical Engineering*, Vol. 27, p. 27-44.
- Zelinka I. (2004) SOMA. In: B.V. Babu, G. Onwubolu Eds., *New optimization techniques in engineering*, Springer-Verlag, chapter 7, ISBN 3-540-20167-X

## AUTHOR BIOGRAPHIES



**ROMAN SENKERIK** was born in Czech Republic, and went to the Tomas Bata University in Zlin, where he studied technical cybernetics and obtained his degree in 2004. He is now Ph.D. student (control of chaos by evolutionary algorithms) at the same university. His e-mail address is: [senkerik@ft.utb.cz](mailto:senkerik@ft.utb.cz)



**IVAN ZELINKA** was born in Czech Republic, and went to the Technical University of Brno, where he studied technical cybernetics and obtained his degree in 1995. He obtained Ph.D. degree in technical cybernetics in 2001 at Tomas Bata University in Zlin. Now he is associated profesor (artificial intelligence, theory of information) and head of department. His e-mail address is: [zelinka@ft.utb.cz](mailto:zelinka@ft.utb.cz) and his Web-page can be found at <http://www.ft.utb.cz/people/zelinka>.

# INVESTIGATION ON SHANNON-KOTELNIK THEOREM IMPACT ON SOMA ALGORITHM PERFORMANCE

Zuzana Oplatková  
Ivan Zelinka  
Institute of Process Control and Applied Informatics  
Faculty of Technology  
Tomas Bata University in Zlin  
Mostni 5139 Zlin, Czech Republic  
E-mail: {oplatkova,zelinka}@ft.utb.cz

## KEYWORDS

Self-Organizing Migrating Algorithm (SOMA), sampling, Shannon-Kotelnik theorem

## ABSTRACT:

This paper deals with evolutionary algorithm SOMA (SelfOrganizing Migrating Algorithm) and studies the performance impact from Step size point of view based on information theory. For this preliminary study a simple unimodal (with one extreme) function was chosen. The article describes technique of analysis and results are well-arranged shown in graphs and charts.

## INTRODUCTION:

Computational performance of algorithms is mostly influenced by sampling. Too dense sampling will task Equipment with a long computational time. On the other hand, thin sampling can disturb the image of original signal. This contribution deals with an analysis of a method of sampling during run of one newer evolutionary algorithm - SelfOrganizing Migrating Algorithm (SOMA) (Zelinka 2000, 2001, 2002, 2004). The first part describes SOMA itself, then Shannon – Kotelnik sampling theorem and its application on a simple unimodal (with one extreme) will follow. Next section will focus on an analysis of sampling of SOMA on this function. As conclusion a possible improvement of SOMA will be pointed up.

## MOTIVATION:

SOMA is an algorithm which uses sampling during its run as is described in next section. Because of its properties we were interested if Step (parameter) of SOMA is set up in suitable way. In other words, if sampling of SOMA comply with known Shannon – Kotelnik theorem (Bracewell R. N., 1999). Till this time this parameter was set up heuristically on the previous experience with performed simulations (Zelinka 2000, 2001, 2002, 2004).

## SOMA

SelfOrganizing Migrating Algorithm (SOMA) is a one newer evolutionary algorithm based on a conception of cooperative – competitive strategy. The construction of new population of individuals is not based on evolution principle (two parents produce offspring) but on the behaviour of social group, e.g. a herd of animals looking for food. During one generation, in the case of SOMA this is called ‘Migration loop’, only the position of individuals in the search space is changed. From the geometrical point of view the run of SOMA is possible to imagine as movements of individuals on the surface of the cost function.

In every migration loop the best individual is chosen, i.e. individual with the minimum cost value, which is called Leader. An active individual from the population moves in the direction to Leader in the search space. At the end of the movement the position of the individual with minimum cost value is chosen. If the cost value of the new position is better than the cost value of an individual from the old population, the new one appears in new population. Otherwise the old one rests there. The movement is described by Eq. (1).

$$x_{i,j}^{MK+1} = x_{i,j,START}^{MK} + (x_{L,j}^{MK} - x_{i,j,START}^{MK}) * t * PRTVector \quad (1)$$

where

$x_{i,j}^{ML+1}$  - value of i-individual’s j-parameter, in step t in next migration loop ML + 1

$x_{i,j,START}^{ML}$  - value of i-individual’s j-parameter, Start position in actual migration loop

$x_{L,j}^{ML}$  - value of Leader’s j-parameter in migration loop ML

t - step  $\in$  <0, by Step to, PathLength>

PRTVector - is vector of ones and zeros depended on PRT. If random number from interval <0, 1> is less than PRT, then 1 is saved to PRTVector, otherwise it will be 0.

SOMA works with controlling and stopping parameters which are summarized in Table1.

Table 1: Parameters of SOMA

Parameter name	Recommended range	Note
PathLength	<1.1, 3>	Controlling parameter
Step	<.11, PathLength>	Controlling parameter
PRT	<0, 1>	Controlling parameter
Dim	Given by problem	Number of arguments in cost function
PopSize	<10, up to user>	Controlling parameter
Migrations	<10, up to user>	Stopping parameter
MinDiv	<arbitrary negative, up to user >	Stopping parameter

**PathLength**  $\in$  <1.1, 3>. This parameter defines how far an individual stops behind the Leader (PathLength=1: stop at the leader's position, PathLength=2: stop behind the leader's position on the opposite side but at the same distance as the starting point). If it is smaller than 1, then the Leader's position is not overshoot, which carries the risk of premature convergence. In that case SOMA may get trapped in a local optimum rather than finding the global optimum. The recommended value is 3.

**Step**  $\in$  <.11, PathLength>. The step size defines the granularity with which the search space is sampled. In case of simple objective functions (convex, one or a few local extremes, etc.), it is possible to use a large Step size in order to speed up the search process. If prior information about the objective function is not known, then the recommended value should be used. For greater diversity of the population, it is better if the distance between the start position of an individual and the Leader is not a multiple of the Step parameter. That means that a Step size of 0.11 is better than a Step size of 0.1, because the active individual will not reach exactly the position of the Leader. The recommended value set up to 0.11 was taken from (Zelinka 2000, 2001, 2002, 2004). But this article will show that the recommended value does not have to be always 0.11. The aim of this article is to show that according to Shannon – Kotelnik theorem the value can be changed adaptively during the run.

**PRT**  $\in$  <0, 1>. PRT stands for perturbation. This parameter determines whether an individual will travel directly towards the Leader, or not. It is one of the most sensitive control parameters. The optimal value is near 0.1. When the value for PRT is increased, the convergence speed of SOMA increases as well. In the case of low dimensional functions and a great number of individuals, it is possible to set PRT to 0.7-1.0. If PRT Equals 1 then the stochastic component of SOMA disappears and it performs only deterministic behaviour suitable for local search.

**Dim** - the dimensionality (number of optimized arguments of cost function) is given by the optimization problem. Its exact value is determined by the cost function and usually cannot be changed unless the user can reformulate the optimization problem.

**PopSize**  $\in$  <10, up to the user>. This is the number of individuals in the population. It may be chosen to be 0.2 to 0.5 times of the dimensionality (Dim) of the given problem. For example, if the optimization function has 100 arguments, then the population should contain approximately 30-50 individuals. In the case of simple functions, a small number of individuals may be sufficient; otherwise larger values for PopSize should be chosen. It is recommended to use at least 10 individuals (two are minimum), because if the population size is smaller than that, SOMA will strongly degrade its performance to the level of simple and classical optimization methods.

**Migrations**  $\in$  <10, up to user>. This parameter represents the maximum number of iterations. It is basically the same as generations for GA or DE. Here, it is called Migrations to refer to the nature of SOMA - individual creatures move over the landscape and search for an optimum solution. 'Migrations' is a stopping criterion, i.e. it tells the optimizing process when to stop.

**MinDiv**  $\in$  <arbitrary negative, up to the user >. The MinDiv defines the largest allowed difference between the best and the worst individual from the actual population. If the difference is too small, then the optimizing process will stop. It is recommended to use a small value. It is safe to use small values for the MinDiv, e.g. MinDiv = 1. In the worst case, the search will stop when the maximum number of migrations is reached. Negative values are also possible for the MinDiv. In this case, the stop condition for MinDiv will not be satisfied and thus SOMA will pass through all migrations.

## SHANNON – KOTELNIK THEOREM

SOMA has as a sampling parameter Step. To carry out an analysis, how Step influences the run of SOMA, was necessary to make a study of a cost function according to sampling Shannon - Kotelnik theorem (Abramson N. 1963, Proakis J. G., 1989, Hamming R.W., 1980).

Shannon – Kotelnik theorem says: If in the process of sampling the information must not lose, a frequency of sampling  $\omega_s$  and maximal frequency  $\omega_m$  included in the signal spectrum have to comply with a condition in the Eq. (2).

$$\omega_s \geq 2 \omega_m \quad (2)$$

It results from the definition that the aim is to find a maximal frequency in the Fourier analysis of the cost function.

## EXPERIMENTS

### Fourier transform

As a cost function it was chosen a simple unimodal function (1<sup>st</sup> DeJong) for 2 arguments Eq. (3) for this purpose.

$$\sum_{i=1}^{Dim} x_i^2 \quad (3)$$

where

$i$  –  $i$ -argument of the function

Dim – dimension (number of optimized arguments of the cost function. In this case Dim = 2.

Decomposition of a function on separate signals solves Fourier transform. It works with periodical signals but some interval of a function can be imagined as one period. Therefore it can be used Fourier transform (Bracewell R. N., 1999) also for such case. Basic Fourier transform is given by Eq. (4) (Bracewell R. N., 1999, Farlow 1993).

$$f(t) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(t) e^{j\omega t} dt \quad (4)$$

It was used to make a Fourier transform of the chosen function. The quality of solution of Fourier transform depended on number of expansion members. Graphs with the origin and created function by means of Fourier transform for number of expansion members from 1 till 20 were made. Then it was chosen heuristically which one could correspond with acceptable error.

In the Fig. 1 the blue space shows the difference between original function in Eq. (3) and the reconstructed function based on a few terms of the Fourier transform - for number of expansion members which was equal 1.

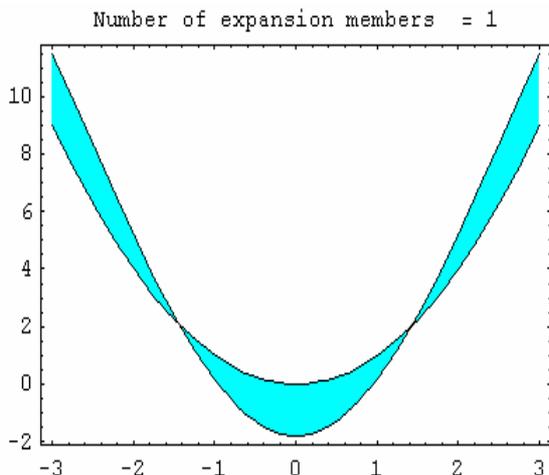


Fig. 1: Difference between original function and Fourier transform (Number of expansion members =1)

Fourier transform is an infinite series of expansion. Therefore it was necessary to find the first which could

correspond with the origin function. In this case number of expansion members Fourier transform was equal to 7 (Fig. 2).

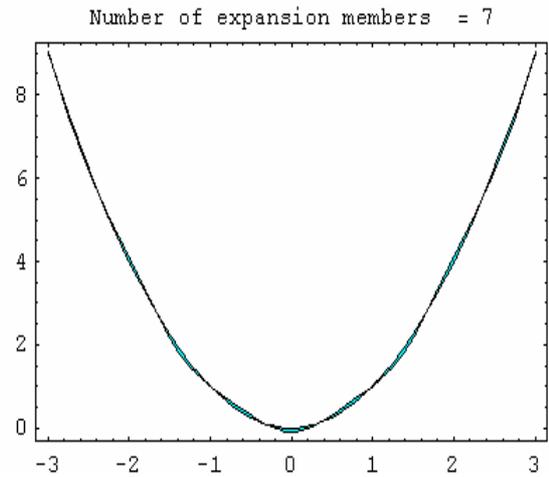


Fig. 2: Difference between original function and Fourier transform (Number of expansion members =7)

Fourier transform with coefficients for this case was found as an expression in the Eq. (5).

$$8.33333 - 10.1321 \cos[0.628319 t] + 2.53303 \cos[1.25664 t] - 1.12579 \cos[1.88496 t] + 0.633257 \cos[2.51327 t] - 0.405285 \cos[3.14159 t] + 0.281448 \cos[3.76991 t] - 0.206778 \cos[4.39823 t] \quad (5)$$

Each member of expansion can be drawn as a single curve whose sum will give the origin function. Next two figures show curves of single components of Fourier transform. The Fig. 4 is detailed view of Fig. 3 to see the line with the biggest value of frequency. On both pictures this is drawn by dashed line. Detailed view can be seen in Fig. 5.

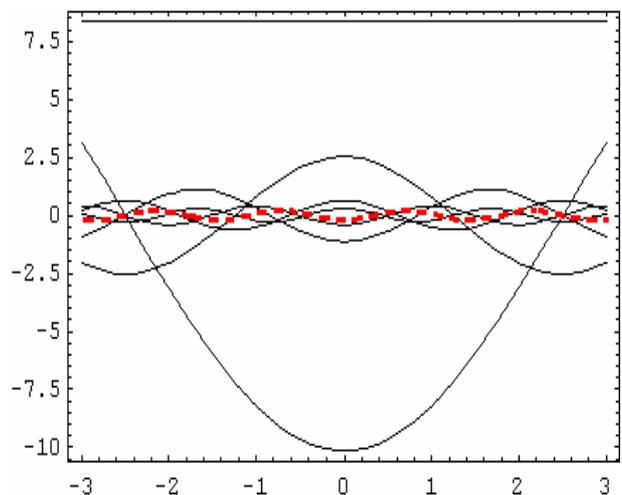


Fig. 3: Single components of Fourier transform for expansion members equal 7

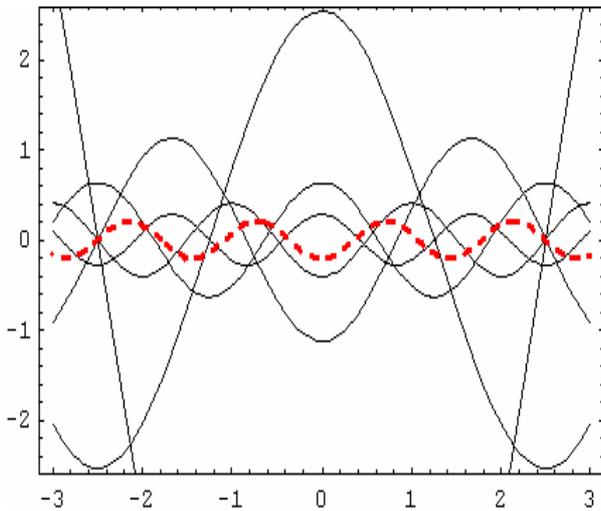


Fig. 4: Detailed view of Fig.3

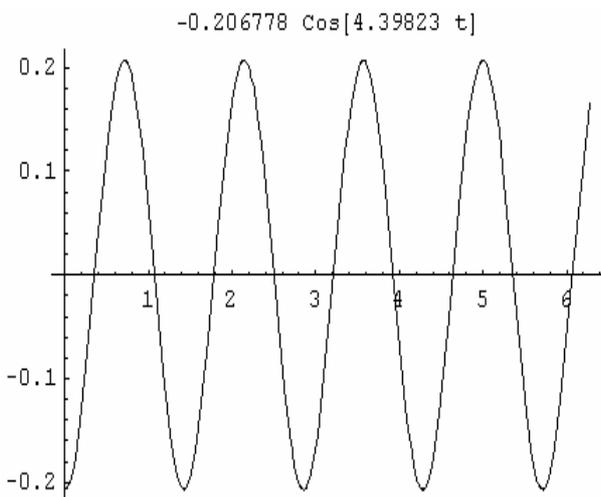


Fig. 5: Chart of the critical member of Fourier transform

This dashed drawn member interprets the critical one for the analysis of function from Shannon – Kotelnik theorem point of view. Its analytic transcription can be seen in Eq. (5) which determines the sampling frequency according to Shannon – Kotelnik theorem (Eq. (6)).

$$F_c = -0,206778 \text{ Cos} (4.39823 t) \quad (6)$$

### Computation of critical frequency

The frequency can be calculated from this expression according to Eq. (7) as reciprocal value of period (Eq. (8)). In this case the result frequency is 0.7.

$$\text{frequency} = 1 / \text{period} \quad (7)$$

$$\text{period} = (2 \pi) / 4.3982 \quad (8)$$

Then according to Eq. (2) the sampling frequency of SOMA should be twice more than the maximal frequency in the spectrum of function, i. e. The critical frequency, which a sampling of SOMA by means of Step should achieve, is equal to 1.4.

### Analysis of parameter Step

Parameter Step as was described above can be explained as sampling parameter of SOMA. It does not work in absolute values but in relative ones. Absolute value of the step is counted from length between Leader and active individual multiplied by value of parameter Step. It can be viewed as part of Eq. (1) – principle of SOMA run.

To compare values of critical frequency sampling of SOMA it was necessary to do reciprocal values of absolute step which can be regarded as a sampling period.

On the basis of carried simulations following charts were prepared. The simulation consists in comparing reciprocal values of absolute step with critical frequency. After that the critical Steps were determined. In other words, there were determined maximal values which can be set up in parameter Step satisfied the conditions of Shannon – Kotelnik theorem and thus fact that algorithm will use information obtained by stepping in an efficient way.

Next figures show the critical steps for each individual in the population during migrations. They are depicted without Leader because critical Steps of all other individuals are outspread to the position of Leader. Fig.6 shows first population which was generated randomly. The other population are generated during evolution process. It can be seen that in the first migration the critical Steps are small because individuals are spread on the surface of the function randomly and it is necessary to sample the function very precisely to get to exploit information about cost function. The individual which is very near to Leader is in the biggest peak. There is not important to make so small steps as in the case of others. The value of Critical Step is in the interval <0,3>. Where the value 3 means that the individual is on the same position as Leader and it is expected that they both could be in the extreme.

Fig. 7 shows the same case as the Fig. 6 but in the decreasing direction of values Steps. There can be better seen that not many individuals are near to extreme and that a lot need to sample the way to Leader in very small steps.

Fig. 8 and Fig. 9 are similarly generated as previous two but in selected migration during the run of SOMA. On these pictures can be seen that most of individuals achieved the same position as Leader. There is depicted agglomeration of individuals of typical run of SOMA in the global extreme as is described in (Zelinka 2002).

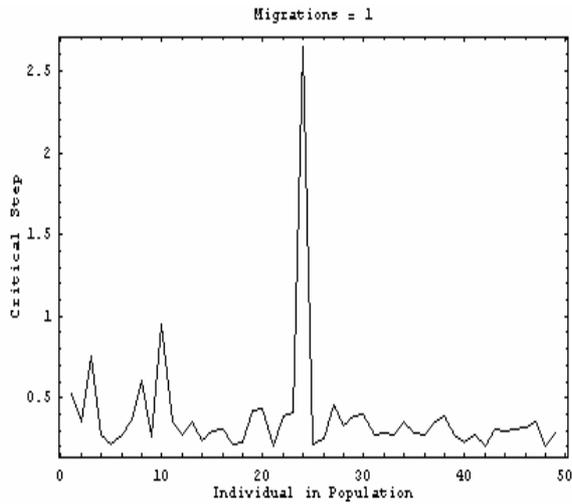


Fig. 6: Chart of the critical values of parameter Step in first migration

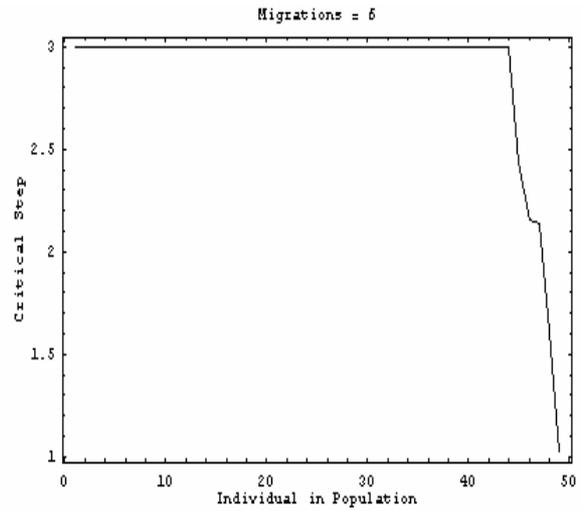


Fig. 8: Chart of the critical values of parameter Step in his decreasing direction during the run of SOMA

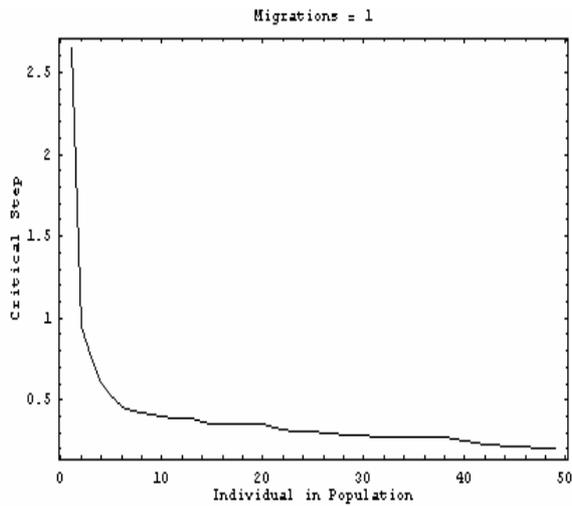


Fig. 7: Chart of the critical values of parameter Step in his decreasing direction in first migration

Fig. 9 shows the situation when all individuals are in the global extreme. The line is at the position of value 3 because such value of Step says that individual is in the same position as Leader. The aim of SOMA is to have all individual at same position in global extreme of cost function.

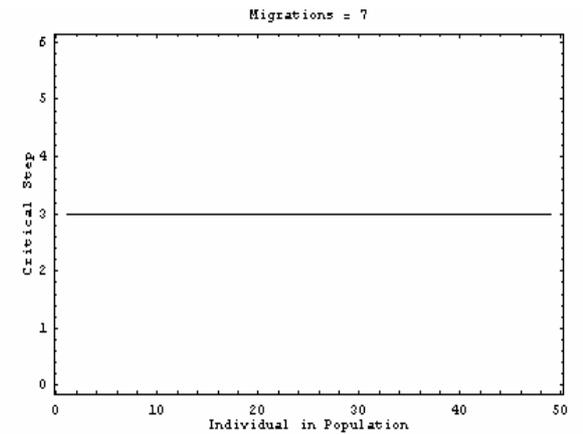


Fig. 9: Chart of the critical values of parameter Step in the last migration

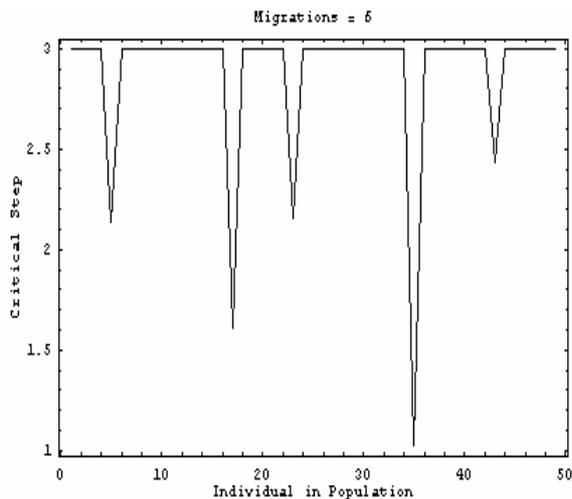


Fig. 8: Chart of the critical values of parameter Step during the run of SOMA

## CONCLUSION

This contribution deals with analysis of sampling method in the evolutionary algorithm SOMA. According to Shannon – Kotelnik theorem critical values of parameter Step were determined. They say which maximal value of Step has to be set up to keep conditions. All was well-arranged displayed in figures. This study was preliminary in this field. Our future work will be concerned to make studies on more difficult cost functions (multimodal).

On the basis of the results from these studies is expected to be tried new version of SOMA which will use the method of this analysis to tune Step adaptively during the run of SOMA. It may improve to obtain high-quality

solution in shorter time. Then it is planned to make studies how much faster the new version of SOMA will be, compared to the current one.

## ACKNOWLEDGEMENT

This work was supported by grant No. MSM 26500014 of the Ministry of Education of the Czech Republic and by grants of the Grant Agency of the Czech Republic GACR 102/03/0070.

## REFERENCES

- Abramson N., 1963, Information Theory and Coding, McGraw – Hill, Book Company, New York, 1963
- Farlow S. J., 1993, Partial Differential Equations for Scientists and Engineers, New York, Wiley, 1993, ISBN 0-486-67620-X
- Bracewell R. N., 1999, The Fourier Transform & Its Applications, McGraw-Hill Science/Engineering/Math, 1999, ISBN 0073039381
- Hamming R.W., 1980, Coding and Information Theory, Prentice – Hall, Englewood Clifts, New York, 1980
- Proakis J. G., 1989, Digital Communication, McGraw – Hill Book Company, 1989
- Zelinka I., Lampinen J. 2000, SOMA – Self-Organizing Migrating Algorithm, Mendel 2000, 6<sup>th</sup> International Conference on Soft Computing, Brno, 2000, Czech Republic, ISBN 80-214-1609-2
- Zelinka I., 2001, Prediction and analysis of behavior of dynamical systems by means of artificial intelligence and synergetic, Ph.D. thesis, UTB, Zlin, 2001
- Zelinka I., 2002, Artificial Intelligence in the Problems of Global Optimization (Czech Edition), BEN – technická literatura, Praha 2002, ISBN 80-7300-069-5.
- Zelinka I., 2004, SOMA – Self Organizing Migrating Algorithm, Chapter 7, 33p., in B.V. Babu, G. Onwubolu (eds.), New Optimization Techniques in Engineering, Springer – Verlag, ISBN 3-540-20167X

## AUTHOR BIOGRAPHIES



**ZUZANA OPLATKOVA** was born in Czech Republic, and went to the Tomas Bata University in Zlin, where she studied technical cybernetics and obtained her degree in 2003. She is now Ph.D. student and lecturer (artificial intelligence) at the same university. Her e-mail address is:

[oplatkova@ft.utb.cz](mailto:oplatkova@ft.utb.cz)



**IVAN ZELINKA** was born in Czech Republic, and went to the Technical University of Brno, where he studied technical cybernetics and obtained his degree in 1995. He obtained Ph.D. degree in technical cybernetics in 2001 at Tomas Bata University in Zlin. Now he is associated professor (artificial intelligence, theory of information) and head of department. His e-mail address is: [zelinka@ft.utb.cz](mailto:zelinka@ft.utb.cz) and his Web-page can be found at <http://www.ft.utb.cz/people/zelinka>.

# INVESTIGATING THE USE OF BAYESIAN NETWORKS TO PROVIDE DECISION SUPPORT TO MILITARY INTELLIGENCE ANALYSTS

Ken R. McNaught and Venkat V.S.S. Sastry  
Cranfield University  
Engineering Systems Department  
RMCS Shrivenham, Swindon SN6 8LA, UK.  
K.R.McNaught@cranfield.ac.uk

Bernard Ng  
Singapore Armed Forces  
MINDEF Building  
Gombak Drive, Singapore 669645.

**KEYWORDS:** intelligence, military, probabilistic, Bayesian network, decision support, information fusion

## ABSTRACT

In this paper we consider a typical military scenario where the intention of an enemy force is unknown, but there are a number of plausible hypotheses. As time passes, information in the form of various sightings and reports become available. We employ a Bayesian network, a type of probabilistic graphical model, to process these reports and update the probabilities of the various hypotheses in the light of the latest information. We also demonstrate the beneficial effect of incorporating 'negative' or false evidence on the plausibility of the various hypotheses.

## INTRODUCTION

Despite the ever-increasing sophistication of combat simulations, the representation of various aspects of C4I and ISTAR (Intelligence, Surveillance, Target Acquisition and Reconnaissance) remains limited. In this paper, we examine the potential application of Bayesian networks to act as processors of intelligence reports, and provide a simplified, illustrative example. Such processors could be considered to be expert systems or intelligent agents, providing decision support to intelligence analysts within a Headquarters cell. Equally, they could be embedded within a combat simulation, representing one of the roles normally conducted by a military Headquarters' intelligence cell.

The network envisaged does not directly process low-level sensor and signal data, but operates on higher-level information provided by an intelligence analyst. It is intended to assist such analysts in making sense of the 'bigger picture', e.g. 'What is the enemy's most likely course of action given all of the indicators received and the background to the situation?' According to the JDL model, well-known within the data fusion community (see, for example, Llinas et al, 2004), such questions are associated with Level 3 in the hierarchy of data and information fusion. Answering such questions can

involve some very complicated reasoning, largely of a diagnostic nature. Unfortunately, many psychological studies (e.g. see Kahneman, Slovic and Tversky, 1982) have demonstrated that, through no fault of their own, humans often perform poorly at such tasks. This is one reason why there is such interest in computerised systems to assist with reasoning in complicated, uncertain domains such as medicine. Such systems are not intended to replace expert assessments, but to provide a second opinion. If that second opinion concurs with the expert's, then it will provide additional confidence in the assessment. On the other hand, if there is a strong disagreement, it may encourage the expert to re-examine the situation and check that nothing crucial has been overlooked.

Bayesian networks provide a powerful method of reasoning in domains where uncertainty is prevalent. As each new piece of evidence is received, the network can propagate its effects to whichever other nodes in the network are affected by it. This results in an updated set of beliefs regarding the key unknown variables of interest - usually one or more main hypotheses which are not directly observable until it is too late. Some other fields of application include medical diagnosis (e.g. Nikovski 2000), intelligent troubleshooting systems (e.g. Breese and Heckerman 1999) and data mining (e.g. Heckermann 1997).

## BAYESIAN NETWORKS

A Bayesian network (BN) consists of a directed acyclic graph (DAG) and a set of conditional probability distributions for each node in the network. The graph comprises a set of nodes, with each node representing a proposition or variable within the domain of interest, and a set of directed arcs representing direct probabilistic dependencies between the variables. The absence of an arc between two variables is interpreted as a statement of conditional independence, i.e. the two variables are independent given some subset of the other variables in the network. For each variable without parents, we need to provide a prior probability distribution. For each variable with parents, we need to specify a conditional

probability distribution given each possible combination of parent states.

The conditional probability distributions which accompany a particular ordering of the variables in the graph, provide a compact way to specify the joint probability distribution over the entire set of variables,  $U = \{A_1, A_2, \dots, A_n\}$ :

$P(U) = \prod_i P(A_i | pa(A_i))$ , where  $pa(A_i)$  refers to the parents of variable  $A_i$  in the graph.

There are many potential orderings of variables in a network, and the ordering chosen for a BN should represent the assumed dependencies and independencies as efficiently as possible. This usually means that the direction of an arc should follow the direction of causality when the relationship between two variables is causal. So, it is the activities (or intent indicators) undertaken by the Red side which cause reports to be generated, the reports do not cause the activities to take place. Not all relationships in a BN have to be causal - weaker probabilistic dependencies will often be present. Exactly how such relationships should be represented and which way the arcs should be directed usually becomes clearer once the modeller has thought through their dependency implications. An invaluable guide in this respect is the d-separation criterion. See Pearl (1988) or Jensen (2002) for more details of this and for an introduction to Bayesian networks, more generally.

## AN ILLUSTRATIVE SCENARIO

In this paper, we consider a simplified, general scenario in which the Blue force HQ is trying to ascertain the Red force's course of action.

Four possible courses of action (COAs) are considered in the scenario - main attack (M), advance (A), defend (D) and withdraw (W). It is assumed here that the Red force will only pursue a single course of action at any given time, although this assumption could be relaxed if it were thought necessary. This does not mean, however, that we do not permit the Red force to operate a deception plan.

### Indicators of Enemy Intent

The intelligence staff tasked with inferring the Red force's most likely course of action, will have a number of cues, or key pieces of information, which they are interested in observing. These cues will be indicative of the course of action being taken by the Red force. Some of these indicators of enemy intent will be associated with a single course of action, while others will be

associated with more than one. Even when an indicator is associated with more than one COA, however, it might still provide greater support for one belief than another. The 20 indicators of enemy intent (IEI) considered here include the following:

- Increase in recce activities
- Increase in counter-recce activities
- Forward movement of supplies
- Establishment of airfields
- Forward movement of missiles
- Radio silence
- Use of smoke
- Erection of obstacles
- Preparation of dummy positions
- Increased anti-tank assets with forward units
- Evacuation of some services
- Destruction of various facilities

## A BAYESIAN NETWORK MODEL OF THE SCENARIO

A basic Bayesian network for this scenario is shown in Figure 1. It consists of a hypothesis node, 'Enemy Intent', a layer of intermediate nodes representing a range of enemy intent indicators, and a layer of nodes representing battlefield intelligence reports relating directly to these indicators. A more complete network might also include various environmental variables.

The final number attached to each report node identifies which of the 20 enemy indicators it relates to. For example, node 'S3MA1' is a report from a Blue sub-unit that Red has increased its air and ground reconnaissance (recce) activities. This relates to the first enemy intent indicator in the middle layer of nodes. Note that setting the state of a report node to 'True' is not the same as setting the state of the corresponding enemy intent indicator to 'True'. Since incorrect and incomplete reports are to be expected, it is important to distinguish between the actual state of a variable and its perceived state. The conditional probability distribution of the report variable given the actual indicator variable will determine how much our belief in the indicator variable changes given a particular report.

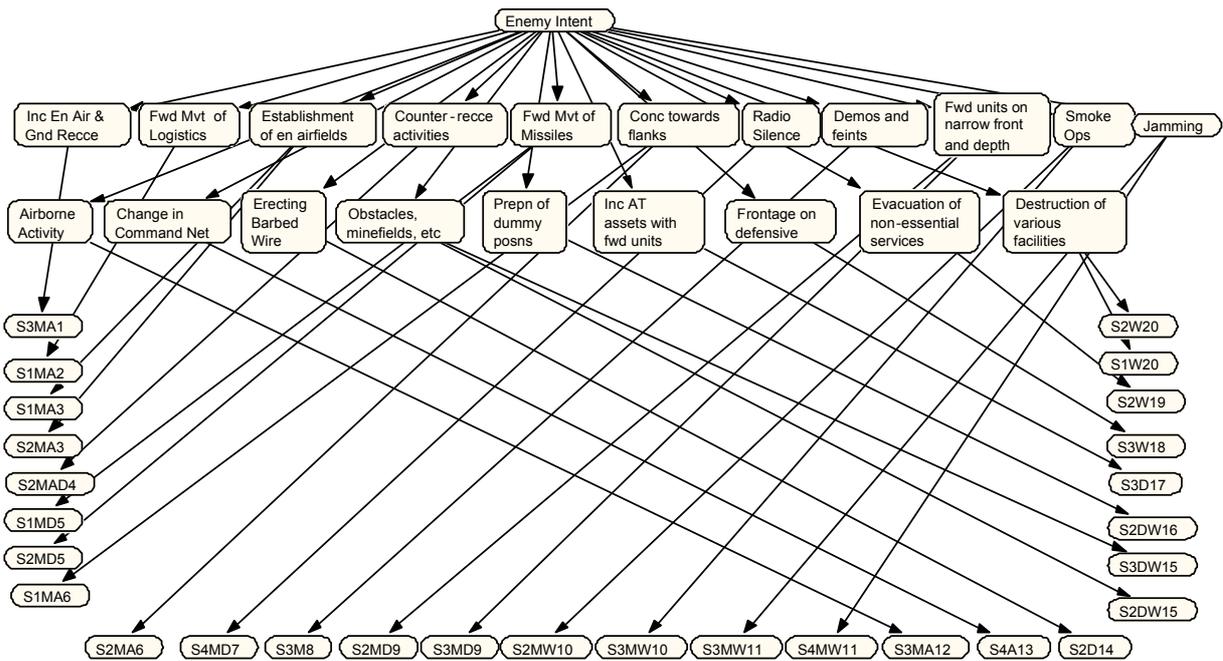


Figure 1: A Bayesian Network of the Relationships Between Enemy Intent Indicators and Intelligence Reports

While space constraints preclude a listing of the illustrative probability distributions contained within this model, a uniform prior distribution was assumed for 'Enemy Intent'. Obviously, the prior chosen reflects initial conditions and the knowledge of the Blue side. Then, as battlefield intelligence reports become

available, so the corresponding nodes will be instantiated. This leads to the probability distributions associated with the enemy intent indicator nodes and the hypothesis node being updated accordingly, as demonstrated in Figure 2.

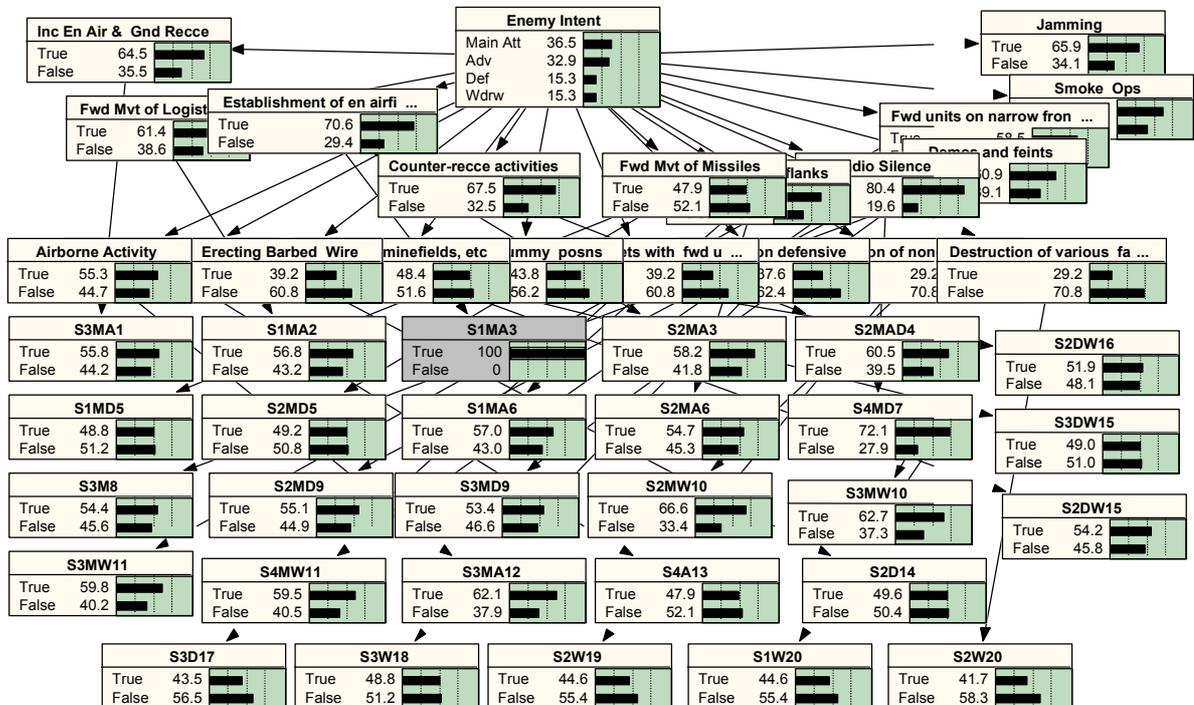


Figure 2: BN Showing Marginal Distribution of All Variables After Node 'S1MA3' is Set to 'True'

As can be seen in Figure 2, for example, if there is an air reconnaissance report of establishment of enemy airfields, represented in the network by setting node 'S1MA2' to 'True', then, denoting that evidence by 'E':  $P(M | E) = 0.365$ ;  $P(A | E) = 0.329$ ;  $P(D | E) = 0.153$  and  $P(W | E) = 0.153$ .

The marginal distribution of every variable which is d-connected to 'S1MA2' is also updated at the same time.

### Results of the First Experiment

We consider two vignettes to illustrate the approach. In the first one, the Red force is preparing to advance, while in the second one, the Red force is actually preparing to withdraw but attempts to conceal its intentions and deceive the Blue force into believing that it is preparing an attack.

The timeline for the first vignette is as shown in Table 1. Here, time is measured discretely in a number of steps. This denotes a simple chronological ordering of the events, which is sufficiently accurate for our purposes. It is not implied that the time steps are all of equal size. Also, there is a delay between an event's occurrence and the subsequent detection and reporting of that event to Blue HQ. As space constraints make it impractical to show the updated network after each time step, the graph in Figure 3 shows how the probability distribution of 'Enemy Intent' changes with time. Although this graph is shown as connected to make it easier to identify the

various states, the probability updates only occur at the discrete time steps.

Table 1: Timeline for Vignette 1.

Time Step	Actions Taken by the Red Side and Indicators Detected by the Blue Side
1	Both sides deploy air and ground recce.
2	Red deploys airborne forces to establish aux airfield; Blue sub-unit reports sighting of Red recce (S3MA1).
3	Red establishes aux airfield for CAS; Blue sub-unit reports sighting of Red airborne forces (S3MA12).
4	Red establishes counter-recce to cover advance route; Blue air and ground recce report sightings of Red aux airfield (S1MA3 and S2MA3).
5	Red formation strengthens flanks; Red forward movement of supplies; Blue ground recce reports Red counter-recce activities.
6	Red command net changes for advance; Blue air and ground recce report sightings of Red's strengthened flanks (S1MA6 and S2MA6); Blue air recce reports sightings of Red's forward movement of supplies (S1MA2).
7	Red begins advance; Blue Signals report Red change of command net (S4A13).

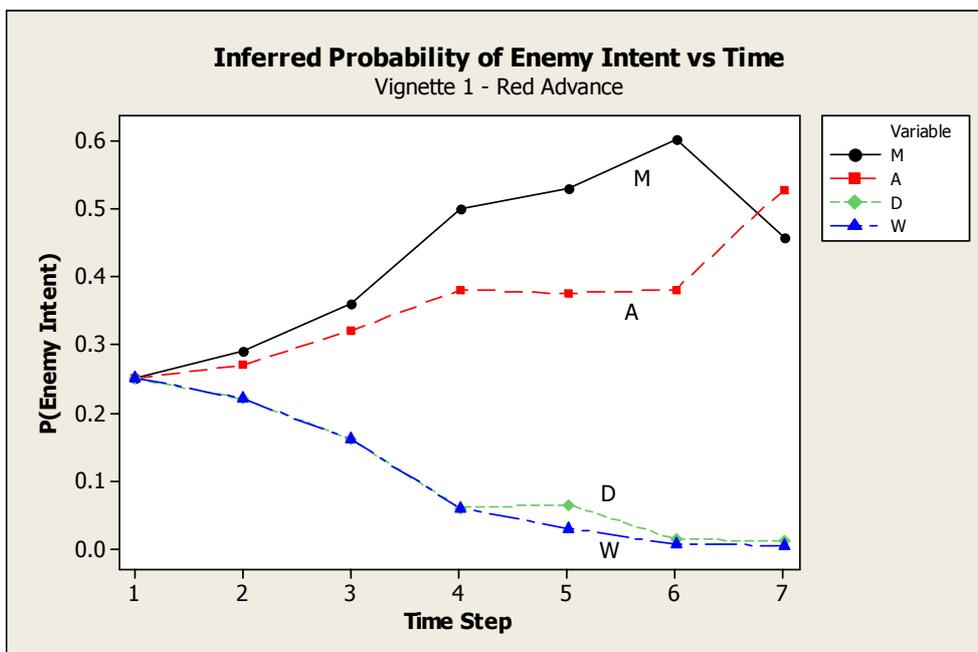


Figure 3: Probability Distribution of 'Enemy Intent' vs Time for Vignette 1

It is clear from the graph in Figure 3 that the correct enemy intent is eventually inferred, the final distribution being:

$$P(M | \text{All evidence}) = 0.46; P(A | \text{All evidence}) = 0.53;$$

$$P(D | \text{All evidence}) = 0.01; P(W | \text{All evidence}) = 0.005.$$

However, for a long time 'Main Attack' was considered the most likely intention, with 'Advance' only overtaking it towards the very end. Such an outcome could only be considered a partial success for the network.

The timeline for the second vignette is as shown in Table 2. A graph showing how the probability distribution of 'Enemy Intent' changes with time

Table 2: Timeline for Vignette 2.

Time Step	Actions Taken by the Red Side and Indicators Detected by the Blue Side
1	Blue establishes air and ground recce.
2	Red deploys air and ground recce as deception; Red increases counter-recce activities as deception; Red establishes dummy airfields as deception.
3	Red establishes demolition on bridges; Blue sub-unit reports sighting of Red recce (S3MA1); Blue ground recce reports Red counter-recce activities (S2MAD4); Blue air recce reports sighting of Red aux airfields (S1MA3).
4	Red conducts feint attacks; Blue ground recce report sighting of Red aux airfield (S2MA3) and demolition on bridges (S2DW15); Blue sub-unit reports local attacks (S3M8).
5	Red evacuates non-essential services; Blue sub-unit reports sighting of demolition on bridges (S3DW15).
6	Red employs smoke and jamming and a defensive frontage; Blue ground recce reports sighting of Red evacuation of non-essential services (S2W19) and Red's use of smoke (S2MW10); Blue sub-unit reports Red's use of smoke (S3MW10) and jamming (S3MW11); Blue Signals report Red's jamming (S4MW11); Blue sub-unit reports Red's defensive frontage (S3W18).
7	Red begins systematic destruction of bridges and commences withdrawal; Blue air and ground recce report sightings of Red destruction of bridges (S1W20 and S2W20).

for the second vignette is shown in Figure 4. Again, the correct inference is eventually made - this time the final probability distribution of 'Enemy Intent' is:

$$P(M | \text{All evidence}) = 0.39; P(A | \text{All evidence}) = 0.06;$$

$$P(D | \text{All evidence}) = 0.04; P(W | \text{All evidence}) = 0.51.$$

However, in common with the first vignette, the true enemy intent only became apparent towards the end. For much of the time, 'Main Attack' seemed the likelier option.

### Results of a Second Experiment Incorporating Negative Evidence

The intentions of the Red force in these two vignettes were clearly difficult for the Blue HQ to identify. While the correct intentions were eventually identified, these came fairly late. It could be argued, however, that not all of the available relevant information was fed into the Bayesian network. In particular, events associated with one Red intent or another which were not observed to occur were assumed unknown. What would the effect be if after a suitable period of time, such events were reported as definitely not having occurred? This is investigated in a second experiment. The same underlying events are generated as in the first experiment, and the same positive intelligence reports are received at the same times. The difference is that in addition to the positive intelligence reports, there are now a number of 'negative' intelligence reports indicating that certain things have not been reported.

In deciding when to instantiate a report node with negative evidence, we have looked at the latest time we would expect a positive report to be received across the four possible states of Enemy Intent. If it has not been received by that time, we have instantiated a negative report for that indicator in the next time-step.

The revised results for the two vignettes, incorporating the effects of negative evidence, are shown in Figures 5 and 6.

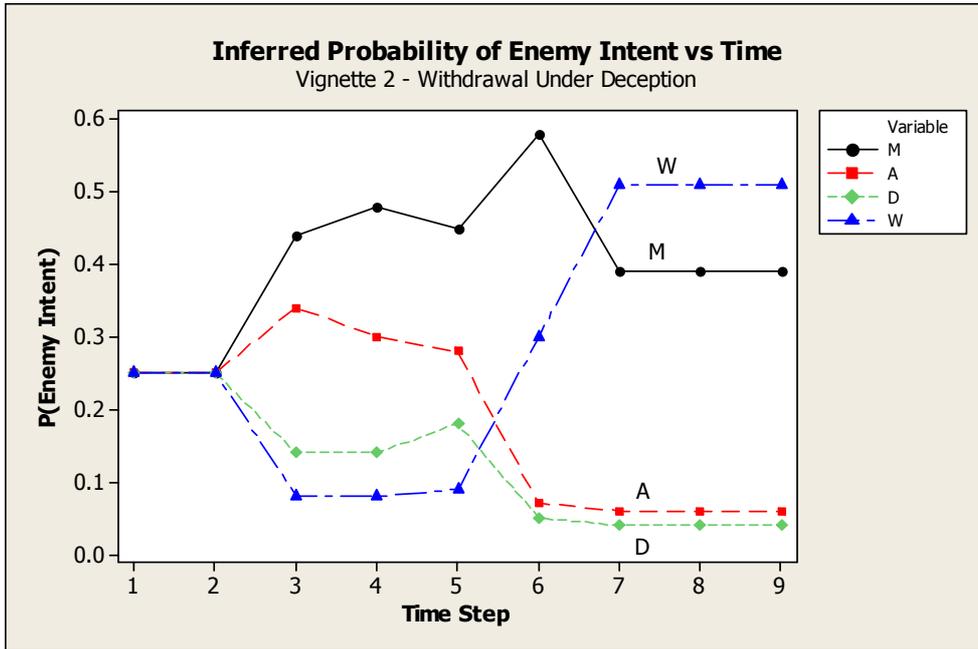


Figure 4: Probability Distribution of Enemy Intent vs Time for Vignette 2

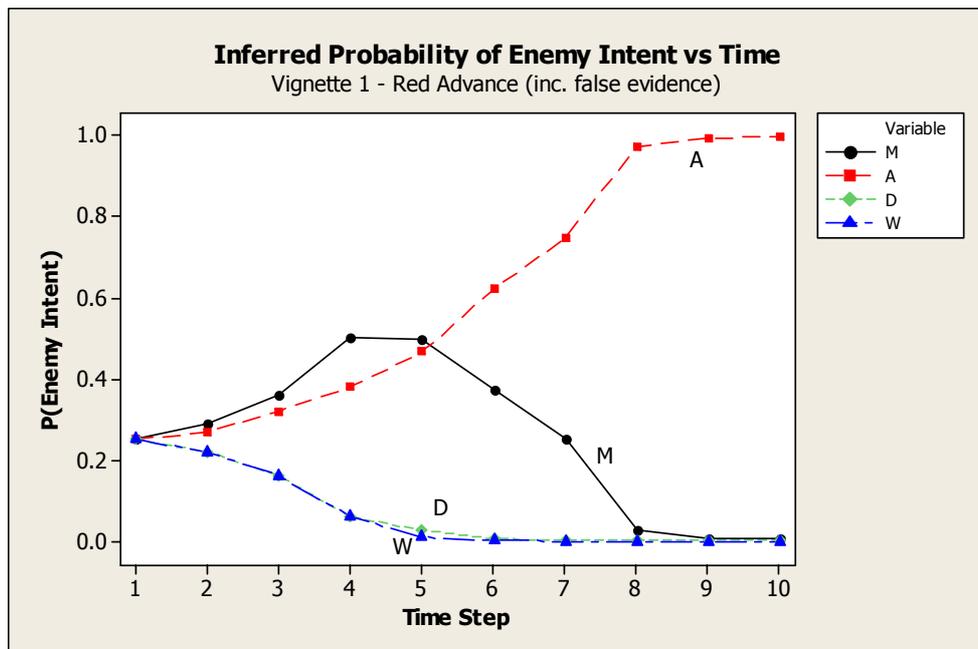


Figure 5: Probability Distribution of Enemy Intent vs Time for Vignette 1 Including Negative Evidence

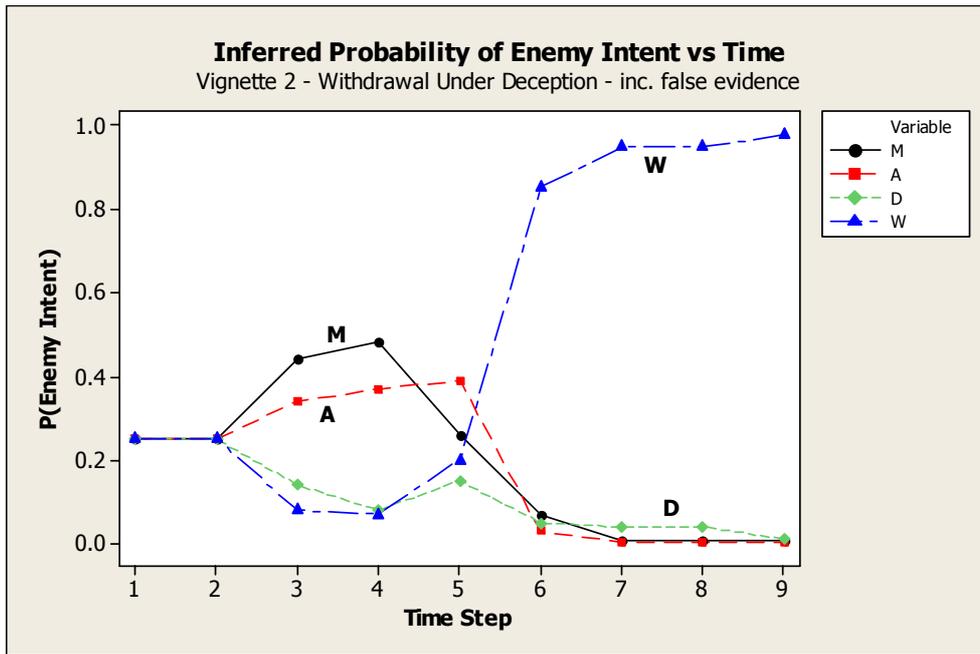


Figure 6: Probability Distribution of Enemy Intent vs Time for Vignette 2 Including Negative Evidence

Clearly, in both cases, the network performs much better when the negative evidence is also taken into account. Firstly, the final distribution of 'Enemy Intent' is more decisive in each case. In Figure 5, the final distribution of enemy intent is now given by:  $P(M | \text{All evidence}) = 0.005$ ;  $P(A | \text{All evidence}) = 0.994$ ;  $P(D | \text{All evidence}) = 0.001$ ;  $P(W | \text{All evidence}) = 0$ .

Similarly, in Figure 6, the final distribution of enemy intent is now given by:  $P(M | \text{All evidence}) = 0.01$ ;  $P(A | \text{All evidence}) = 0.005$ ;  $P(D | \text{All evidence}) = 0.01$ ;  $P(W | \text{All evidence}) = 0.975$ .

Secondly, the correct option is identified earlier by the network in both cases. While it is difficult to quantify the benefit obtained by identifying the true enemy course of action sooner, this could be addressed in a simulation study.

## CONCLUSIONS

Using only positive evidence, the network sometimes has difficulty in discriminating between some of the alternatives. Although in the examples considered here, it eventually 'got it right', this was often very late. The performance was much improved when false evidence, indicating that certain indicators of enemy intent had not been observed, was also employed in the network. False findings at an observation node were only instantiated after the latest time they would normally have been

expected to have been observed had the indicator been present.

Further work will consider how timing information can be better exploited to avoid a sudden rush of false findings towards the end of a scenario causing large, discontinuous jumps in the probability distributions. The use of distributional information on detection times should permit smoother, continual updates in the

distributions over time, with discontinuous jumps occurring only when definite findings are observed. We will also attempt to quantify the benefits which such a decision support system can bring in terms of improved responsiveness. Simulation currently appears to be the most likely method of achieving this goal.

## REFERENCES

- Breese, J. S. and Heckerman, D. 1999. "Decision-Theoretic Troubleshooting." *IEEE Transactions on Systems, Man & Cybernetics, Part A (Systems & Humans)* 26, No. 6, 838-842.
- Heckerman, D. 1997. "Bayesian Networks for Data Mining." *Data Mining and Knowledge Discovery* 1, No. 1, 79-120.
- Jensen, F.V. 2002. *Bayesian Networks and Decision Graphs*. Springer-Verlag, New York.

Kahneman, D.; Slovic, P.; and Tversky, A. (Eds.) 1982. *Judgment Under Uncertainty: Heuristics and Biases*. Cambridge University Press, Cambridge, Mass.

Llinas, J.; Bowman, C.; Rogova, G.; Steinberg, A.; Waltz, E.; and White, F. 2004. "Revisiting the JDL Data Fusion Model II." In *Proceedings of the 7th International Conference on Information Fusion*, Stockholm, 1218-1230.

Nikovski, D. 2000. "Constructing Bayesian Networks for Medical Diagnosis from Incomplete and Partially Correct Statistics." *IEEE Transactions on Knowledge and Data Engineering* 12, No. 4, 509-516.

Pearl, J. 1988. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, San Mateo, California.

## AUTHOR BIOGRAPHIES

**Ken R. McNaught** is a lecturer in Operational Research (O.R.) at Cranfield University's College of Defence Technology in Shrivenham, UK. He has an MSc in O.R. from Strathclyde University and a PhD in O.R. from Cranfield University. His current research interests include simulation, combat modelling and decision support, particularly making use of probabilistic graphical approaches such as Bayesian networks and influence diagrams.

**Bernard Ng** holds the rank of Major in the Singapore Armed Forces, in which he has served for 12 years. He holds a BSc in Computer Science from the National University of Singapore and an MSc in Defence Technology from Cranfield University. He is currently operating as a staff officer SO1 in the area of C3I.

**Venkat V. S. S. Sastry** obtained his PhD (1980) in Applied Mathematics from IISc Bangalore, India. Venkat worked on analytical models of propagation of sound over ground, in underwater sediments (1981-86) as a post doctoral research fellow. He joined Cranfield University at the Royal Military College of Science, Shrivenham in 1989 as a lecturer. He is currently the Director of Applied Mathematics and Scientific Computing. His current interests include scientific visualization, applications of Intelligent Systems and Virtual Environments for Training, gesture interpretation in virtual environments for training applications and application of agents for decision support.

# PARALLEL COMPUTATION PLATFORM FOR SOMA

Miroslav Červenka, Ivan Zelinka  
Institute of Process Control and Applied Informatics  
Faculty of Technology  
Tomas Bata University in Zlín  
Mostní 5139 Zlín, Czech Republic  
E-mail: {cervenka,zelinka}@ft.utb.cz

## KEYWORDS

Parallel computation, cluster, evolutionary algorithm, Self-Organizing Migrating Algorithm, SOMA

## ABSTRACT

This paper describes an universal portable platform for parallel computations and a parallelized version of relatively new evolutionary optimization algorithm – the Self-Organizing Migration Algorithm (SOMA). Results of efficiency tests of the platform and also test results of optimization with parallel SOMA are included.

## INTRODUCTION

Evolutionary algorithms form a class of stochastic optimization algorithms in which principles of organic evolution are regarded as rules for optimization. They are often applied to parameter optimization problems (Bäck, Schwefel, 1993), when specialized techniques are not suitable or standard methods give unsatisfactory results.

Principle of evolutionary algorithms is based on scanning through a space of possible solutions using a population of individuals. Because this could be a very time-demanding process, parallelization might be one of possible solutions. By splitting this task on more computers, the search process can be speeded-up or even improved.

The Self-Organizing Migration Algorithm (SOMA) (Zelinka 2002, 2004) is a kind of relatively new optimization algorithm, whose design enables its parallel implementation very easily. The aim of this work is to develop a universal platform for parallel applications. This framework runs on ordinary office computers used by students in the IT labs and utilizes their idle CPU time. In addition, another objective is to use SOMA to develop a general-purpose optimization toolkit.

## COMPUTATION PLATFORM

To create a computation cluster, a certain number of computers is needed. In our case 50 standard office computers in configuration Athlon XP 2200+, 512 MiB RAM, interconnected by 100Mbps LAN across the campus network were used.

To manage all this computation power a universal distributed environment for parallel applications was developed. This platform was designed to fulfil following requirements: First, there was an idea to utilize the idle CPU time in the IT-labs. These computers are owned by different departments running various operating systems on their machines, therefore Java was chosen to enable simple deployment. Second, it had to be as universal as possible to allow the user to simply create his own application using parallel capabilities without being annoyed with the background problems. Several sample applications and detailed HOW-TOs were written to demonstrate the cluster usage.

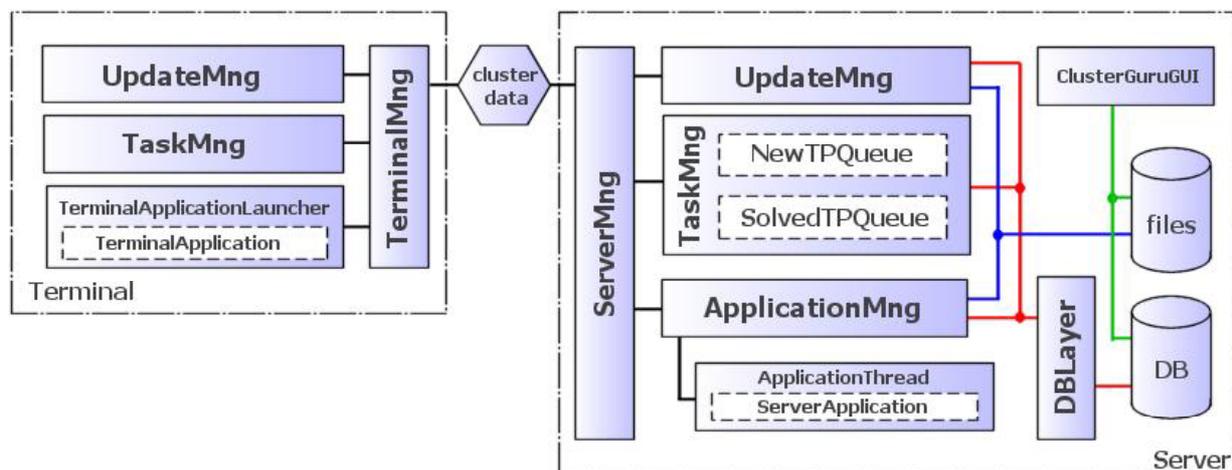


Fig. 1: Cluster structure

## CLUSTER STRUCTURE

The cluster platform is splitted into two parts – server and terminal – and the structure can be seen in Figure 1.

The server is composed of modules that provide services for the whole cluster. First of them is Update Manager, which ensures that all terminals and also user's classes are up to date on all terminals all the time. The second one is Task Manager, which is responsible for correct task distribution. It handles all task packages created by the user's server application and also all incoming task requests or packages with partial task solutions. It contains two task queues. The flow of task packages between the database and Task Manager is realized in groups of packages and thereby the response time was rapidly reduced. When a terminal fails (for example, it is turned off by a student), its task is assigned to another terminal after a certain interval, which means that it also provides basic fail-over capability. The third module is Server Application Manager, which is responsible for starting and shutting down the user's server application and provides many useful utilities for it, like task distribution or direct connection to terminals. The server is able to handle multiple instances of different user's server applications.

The terminal is a standalone application which starts just after computer boots. On the Windows platform it runs as an NT system service, so the student using the computer is not disturbed. The terminal was intended as a smart application which can ask server for an update or task to solve. Furthermore, it is able to find other terminals in the network and create a virtual network to enable direct communication with other terminals. As soon as a package with task description is delivered on the terminal, the relevant class able to process this task is found and thereby the user's terminal application is started. While a task is processed, the terminal's task manager requests another package with task description. This feature can eliminate time delays caused by communication and might be very useful when transferring huge amount of data. The terminal cannot process concurrent tasks.

## PARALLEL APPLICATION

The user's parallel application is also separated into two parts. First of them is the server-side application, which is primarily supposed to manage run of the whole process. It creates packages with description of partial tasks or data needed for the actual computation. Moreover, it decides whether the application run is

over or other calculations have to be done. It can directly communicate with one or more terminals and usually does not perform any computations.

The second part is the terminal-side application where the computation operations are performed. When there is not a task to process, the terminal is hibernated and after a longer period of time it performs a request for a task again.

## CLUSTER MANAGEMENT

All operations required to manage a user's distributed project are available through a graphical user interface (GUI) presented on the web. This solution was chosen to enable an easy access to the cluster server for anyone using just a web browser and simplify all operations as much as possible. In this interface the parallel application project can be created, modified, launched or terminated. The progress of the application and the time needed to finish it can also be seen here. The results are easily accessible without a need of having a shell access. Another advantage of this solution is that the cluster administration can be accessed from various mobile devices, like cell phones or handhelds.

## CLUSTER EFFICIENCY

Quality of distributed environment can be considered according to its effectivity. The loss of performance can be caused by server software, where the server modules are not able to handle incoming requests immediately. In some cases the problem could be caused by the user's server application, when it is not able to create task parts fast enough. In addition, the throughput and time delay of the network might be a limitation.

To find the real efficiency of this parallel platform an artificial problem was created. It emulates computation by n-second time delay on terminals. When the number of terminals and also the total computation duration is known, it is very easy to calculate the time needed for execution on certain number of (at least similar) computers. For example – if we have 10 computers and will use 600 task parts with delay of 20 seconds, we can easily calculate the time needed on the number of computers that are available.

Table 1 and Figure 2 display results of the "TimeCluster" test. For each cluster configuration a virtual problem like in the example described above was created. The average delay was about 6 seconds, which might be due to the task check interval inside the

TimeCluster- artificial test results	Delay [s]	20			
	Packs [-]	600			
Number of terminals [-]	10	20	30	40	50
Time spent on 'computations' in cluster [s]	1207	606	406	308	245
Ideal time in cluster [s]	1200	600	400	300	240
Delay [s]	7	6	6	8	5
Efficiency [%]	99	99	99	97	98

Tab. 1: TimeCluster test results

platform and could be influenced by settings in the configuration file. This partially decreases the cluster efficiency that can be seen in all columns of Table 1. Even though, the cluster efficiency is high enough to consider the platform suitable for parallel computations.

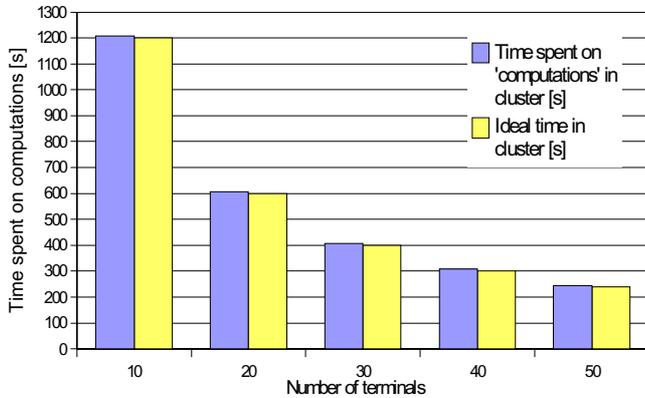


Fig. 2: TimeCluster – artificial test results

## PARALLEL SOMA

SOMA immitates competitive-cooperative behaviour of intelligent creatures. A group of wolves or other predators may be a good example. If they are looking for food, they usually cooperate and compete so that if one member of the group is more successful than the previous best one ( e.g. has found more food ) then all members change their trajectories towards the new most successful member. Exactly this approach is used in the SOMA's One-To-One strategy.

Positions on the hyper-plane are represented by members of the SOMA population – individuals. Each of them stands for an actual solution for the given problem. The best position (the best solution in the particular migration loop) is called leader.

Since evaluation of the optimized cost function for each individual's step can be a time demanding operation, the numerous population of individuals was divided into groups and each group became a separate population on one terminal. Single terminals send a message informing the othes about their local best position – the position of local leader. The information about leader can be either sent from terminals to the server and later back to all terminals, or better, terminals can communicate directly among themselves, which is faster and also more efficient. All features can be set in the relevant configuration files.

While in classic clusters the data is accessible on all nodes at the same time thanks to the shared memory

feature, in this case a little different approach is used. The information about the position of the leader spreads over other terminals in steps, as displayed in Figure 3. After one migration loop the information about the best position is passed only to the neighbouring terminals (to 4 neighbours in this case). If they consider the position to be better as their local leader's position, they will use it as a new local leader and after the next migration loop the information is passed on. The number of neighbours, which influences the information spreading velocity, can be set in the configuration file. This parameter might also have impact on the robustness of the parallelized evolutionary algorithm.

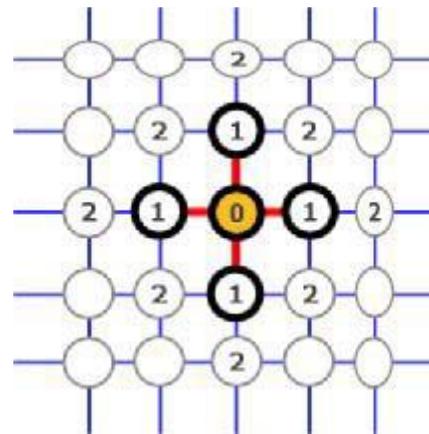


Fig. 3: Spreading of information in the terminal's virtual network

## TESTING

Parallelized SOMA was tested on teaching a neural network. Described network configuration and the training set was chosen only for cluster efficiency tests. It has no other meaning. SOMA configuration was: step = 0.11, pathLength = 3, prt = 0.3, minDiv = 0.001, migrations = 4000. Number of modified neuron-wages = 315, population size = dim/2 = 157 individuals. The network had 4 layers with 3 neurons in the input layer, two hidden layers, 15 neurons each, and 3 neurons in the output layer. Hyperbolic cosine was used as transfer function in all neurons. The teaching set had 21 input and 21 output vectors and its structure was as follows:

input: -1, -1, -1; output: -1, -1, -1  
input: -0.9, -0.9, -0.9; output: -0.9, -0.9, -0.9  
input: -0.8, -0.8, -0.8; output: -0.8, -0.8, -0.8  
....  
input: 1, 1, 1; output: 1, 1, 1

<b>SOMA – neural network teaching</b>		Single computer time [min]: 230				
Number of terminals [-]		10	20	30	40	50
Time spent on 'computations' in cluster [min]		23.8	11.8	7.9	5.9	4.7
Ideal time in cluster [min]		23	11.5	7.67	5.75	4.6
Delay [min]		0.8	0.3	0.23	0.15	0.1
Efficiency [%]		97	97	97	97	98

Tab. 2: Neural network taught by SOMA - test results

INPUTS			OUTPUTS		
Input 1	Input 2	Input 3	Output 1	Output 2	Output 3
-1.00	-1.00	-1.00	-0.91	-0.92	-0.91
-0.90	-0.90	-0.90	-0.86	-0.87	-0.86
-0.80	-0.80	-0.80	-0.80	-0.80	-0.80
-0.70	-0.70	-0.70	-0.72	-0.72	-0.72
-0.60	-0.60	-0.60	-0.62	-0.61	-0.62
-0.50	-0.50	-0.50	-0.51	-0.50	-0.51
-0.40	-0.40	-0.40	-0.40	-0.39	-0.40
-0.30	-0.30	-0.30	-0.30	-0.30	-0.30
-0.20	-0.20	-0.20	-0.20	-0.21	-0.20
-0.10	-0.10	-0.10	-0.10	-0.11	-0.10
0.00	0.00	0.00	0.00	0.00	0.00
0.10	0.10	0.10	0.10	0.11	0.10
0.20	0.20	0.20	0.20	0.21	0.20
0.30	0.30	0.30	0.30	0.30	0.30
0.40	0.40	0.40	0.40	0.39	0.40
0.50	0.50	0.50	0.51	0.50	0.51
0.60	0.60	0.60	0.62	0.61	0.62
0.70	0.70	0.70	0.72	0.72	0.72
0.80	0.80	0.80	0.80	0.80	0.80
0.90	0.90	0.90	0.86	0.87	0.86
1.00	1.00	1.00	0.91	0.92	0.92

Tab. 3: Output of teaching process

The results performed by the parallel version can be seen in Table 2, progress of global error history in Figure 4. The quality of teaching process (in Table 3) can be considered as very high. Only values around -1 and 1 are not accurate, which is caused by the behaviour of the neuron's transfer function.

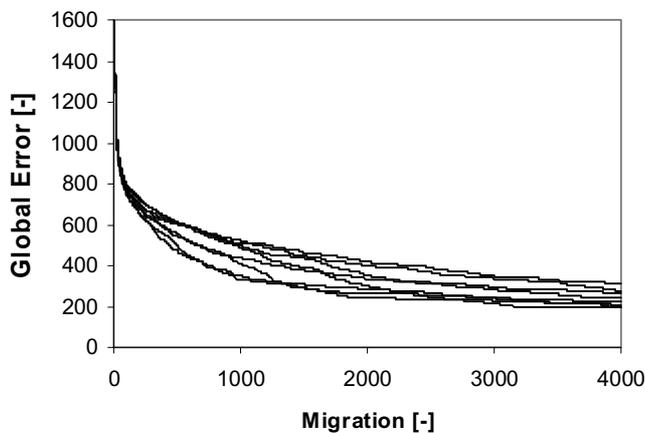


Fig. 4: An example of global error history progress

As can be seen in Figure 5, nearly the same platform efficiency as in the first test was reached. There are several parts of code which can be improved from the efficiency point of view. The fully artificial TimeCluster test shows probably the highest efficiency that can be reached at this stage of development. Remember, that as computation nodes the computers in study classrooms were used and performed simulations were interrupted by computer reboot caused students from time to time. Real applications using described parallel framework might show results similar to the neural network teaching test.

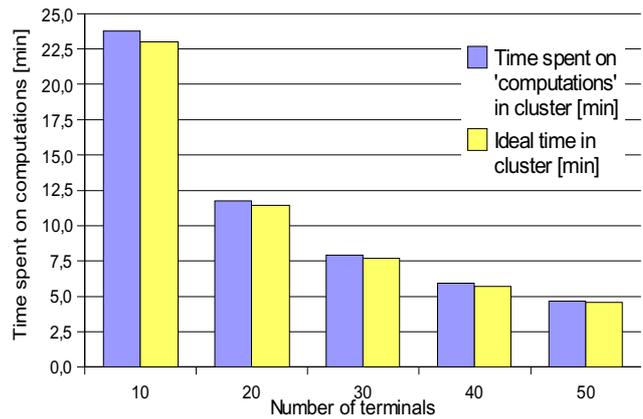


Fig. 5: Neural network taught by SOMA – ideal and real times

## CONCLUSION

A platform for parallel computations was described. Results of efficiency tests demonstrated, that the framework is ready to use for distributed computing. Furthermore, the parallel version of SOMA was introduced and an example of its utilisation was shown. The future work will concentrate on practical applications of parallel SOMA and its improvement. Furthermore, a parallel version of Differential Evolution for the mentioned platform will be thoroughly tested.

## ACKNOWLEDGEMENT

This work was supported by grant No. MSM 26500014 of the Ministry of Education of the Czech Republic and by grants of the Grant Agency of the Czech Republic GACR 102/03/0070.

## REFERENCES

- Bäck T., Schwefel H. P., 1993, An overview of evolutionary algorithms for parameter optimization. *Evolutionary Computation Journal*, No.1, 1-23.
- Zelinka I., 2002, *Artificial Intelligence in the Problems of Global Optimization* (Czech Edition), BEN – technická literatura, Praha 2002, ISBN 80-7300-069-5.
- Zelinka I., 2004, SOMA – Self Organizing Migrating Algorithm“, Chapter 7, 33 p. in: B.V. Babu, G. Onwubolu (eds), *New Optimization Techniques in Engineering*, Springer-Verlag, ISBN 3-540-20167X

- [4] Bajaj R., Agrawal D. P., 2004, Improving Scheduling of Tasks in a Heterogenous Enviroment, IEEE Transactions on Parallel and Distributed Systems, vol. 15, No. 2, February 2004
- [5] Bose N.K., Liang P. 1996, Neural Network Fundamentals with Graphs, Algorithms, and Applications, McGraw-Hill Series in Electrical and Computer Engineering, ISBN 0-07-006618-3, 1996
- [6] Ka-Po Chow, Yu-Kwong Kwok, 2002, On Load Balancing For Distributed Multiagent Computing, IEEE Transactions on Parallel and Distributed Systems, vol. 13, No. 8, August 2002
- [7] Masters T, 1993, Practical Neural Networks Recipes in C++, Academic Press, ISBN 0-12-479040-2, 1993
- [8] Price K., 1999, An Introduction to Differential Evolution, in New Ideas in Optimization, D. Corne, M. Dorigo and F. Glover, Eds., s. 79–108, McGraw-Hill, London, UK, 1999. ISBN 007-709506-5
- [9] Topcuoglu H., Hariri S., Min-You Wu 2002, Performance-Effective and Low-Complexity Task Scheduling For Heterogenous Computing, IEEE Transactions on Parallel and Distributed Systems, vol. 13, No. 3, March 2002
- [10] Yang Y., Wang J., 2003, Nonblocking k-Fold Multicast Networks, IEEE Transactions on Parallel and Distributed Systems, vol. 14, No. 2, February 2003

## AUTHOR BIOGRAPHIES

**MIROSLAV ČERVENKA** was born in Czech Republic, and went to the Tomas Bata University in Zlín, where he studied technical cybernetics and obtained his master degree in 2003. Now, he is a Ph.D. student at the same university. His e-mail address is [cervenka@ft.utb.cz](mailto:cervenka@ft.utb.cz)

**IVAN ZELINKA** was born in Czech Republic, and went to the Technical University of Brno, where he studied technical cybernetics and obtained his degree in 1995. He obtained Ph.D. degree in technical cybernetics in 2001 at Tomas Bata University in Zlín. Now he is associated professor (artificial intelligence, theory of information) and head of department. His e-mail address is [zelinka@ft.utb.cz](mailto:zelinka@ft.utb.cz) and his web-page can be found at [www.ft.utb.cz/people/zelinka/](http://www.ft.utb.cz/people/zelinka/)

# A FAST NEURAL ALGORITHM FOR PATTEN DETECTION USING CROSS CORRELATION IN THE FREQUENCY DOMAIN

Hazem M. El-Bakry

*University of Aizu, Aizu Wakamatsu, Japan 965-8580  
hazemelbakry1@yahoo.com*

**ABSTRACT**—Recently, fast neural networks for object/face detection were presented in [1-3]. The speed up factor of these networks relies on performing cross correlation in the frequency domain between the input image and the weights of the hidden layer. But, these equations given in [1-3] for conventional and fast neural networks are not valid for many reasons presented here. In this paper, correct equations for cross correlation in the spatial and frequency domains are presented. Furthermore, correct formulas for the number of computation steps required by conventional and fast neural networks given in [1-3] are introduced. A new formula for the speed up ratio is established. Also, corrections for the equations of fast multi scale object/face detection are given. Moreover, commutative cross correlation is achieved. Simulation results show that sub-image detection based on cross correlation in the frequency domain is faster than classical neural networks.

**KEYWORDS**—Conventional Neural Networks, Fast Neural Networks, Cross Correlation in the Frequency Domain.

## I. INTRODUCTION

Pattern detection is a fundamental step before pattern recognition. Its reliability and performance have a major influence in a whole pattern recognition system. Nowadays, neural networks have shown very good results for detecting a certain pattern in a given image [7,20]. But the problem with neural networks is that the computational complexity is very high because the networks have to process many small local windows in the images [4,19]. The authors in [1-3] have proposed a multilayer perceptron (MLP) algorithm for fast object/face detection. The same authors claimed incorrect equation for cross correlation between the input image and the weights of the neural networks. They introduced formulas for the number of computation steps needed by conventional and fast neural networks. Then, they established an equation for the speed up ratio. Unfortunately, these formulas contain many errors which lead to invalid speed up ratio. Other authors developed their work based on these incorrect equations [5-18],[20-30]. So, the fact that these equations are not valid must be cleared to all

researchers. It is not only very important but also urgent to notify other researchers not to waste their time and effort doing research based on wrong equations. The main objective of this paper is to correct the formulas of cross correlation as well as the equations which describe the computation steps required by conventional and fast neural networks presented in [1-3]. Some of these wrong equations were corrected in our previous publications [5-18], [20-30]. Here, all of these errors are corrected. In section II, fast neural networks for object/face detection are described. Comments on conventional neural networks, fast neural networks, and the speed up ratio of object/face detection are presented in section III.

## II. FAST OBJECT/FACE DETECTION USING MLP AND FFT

In [1-3], a fast algorithm for object/face detection based on two dimensional cross correlations that take place between the tested image and the sliding window (20x20 pixels) was described. Such window is represented by the neural network weights situated between the input unit and the hidden layer. The convolution theorem in mathematical analysis says that a convolution of  $f$  with  $h$  is identical to the result of the following steps: let  $F$  and  $H$  be the results of the Fourier transformation of  $f$  and  $h$  in the frequency domain. Multiply  $F$  and  $H$  in the frequency domain point by point and then transform this product into spatial domain via the inverse Fourier transform. As a result, these cross correlations can be represented by a product in the frequency domain. Thus, by using cross correlation in the frequency domain a speed up in an order of magnitude can be achieved during the detection process [1-3].

In the detection phase, a sub image  $I$  of size  $m \times n$  (sliding window) is extracted from the tested image, which has a size  $P \times T$ , and fed to the neural network. Let  $W_i$  be the vector of weights between the input sub image and the hidden layer. This vector has a size of  $m \times n$  and can be represented as  $m \times n$  matrix. The output of hidden neurons  $h(i)$  can be calculated as follows:

$$h_i = g \left( \sum_{j=1}^m \sum_{k=1}^n W_i(j,k) I(j,k) + b_i \right) \quad (1)$$

where  $g$  is the activation function and  $b(i)$  is the bias of each hidden neuron ( $i$ ). Eq.1 represents the output of each hidden neuron for a particular sub-image  $I$ . It can be computed for the whole image  $\Psi$  as follows:

$$h_i(u,v) = g \left( \sum_{j=-m/2}^{m/2} \sum_{k=-n/2}^{n/2} W_i(j,k) \Psi(u+j, v+k) + b_i \right) \quad (2)$$

Eq.2 represents a cross correlation operation. Given any two functions  $f$  and  $g$ , their cross correlation can be obtained by [2]:

$$f(x,y) \otimes g(x,y) = \left( \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f(m,n) g(x+m, y+n) \right) \quad (3)$$

Therefore, Eq.2 may be written as follows [1,2]:

$$h_i = g(W_i \otimes \Psi + b_i) \quad (4)$$

where  $h_i$  is the output of the hidden neuron ( $i$ ) and  $h_i(u,v)$  is the activity of the hidden unit ( $i$ ) when the sliding window is located at position  $(u,v)$  in the input image  $\Psi$  and  $(u,v) \in [P-m+1, T-n+1]$ .

Now, the above cross correlation can be expressed in terms of the Fourier Transform:

$$\Psi \otimes W_i = F^{-1} \left( F(\Psi) \bullet F^* \left( W_i \right) \right) \quad (5)$$

(\*) means the conjugate of the FFT for the weight matrix. Hence, by evaluating this cross correlation, a speed up ratio can be obtained comparable to conventional neural networks. Also, the final output of the neural network can be evaluated as follows:

$$O(u,v) = g \left( \sum_{i=1}^q w_o(i) h_i(u,v) + b_o \right) \quad (6)$$

where  $q$  is the number of neurons in the hidden layer.  $O(u,v)$  is the output of the neural network when the sliding window located at the position  $(u,v)$  in the input image  $\Psi$ .

The authors in [1-3] analyzed their proposed fast neural network as follows: For a tested image of  $N \times N$  pixels, the 2D-FFT requires  $O(N^2(\log_2 N)^2)$  computation steps. For the weight matrix  $W_i$ , the 2D-FFT can be computed off line since these are constant parameters of the network independent of the tested image. The 2D-FFT of the tested image must be computed. As a result,  $q$  backward and one forward transforms have to be computed. Therefore, for a tested image, the total number of the 2D-FFT to compute is  $(q+1)N^2(\log_2 N)^2$ . In addition, the input image and the weights should be multiplied in the frequency domain. Therefore, computation steps of  $(qN^2)$  should be added. This yields

a total of  $O((q+1)N^2(\log_2 N)^2 + qN^2)$  computation steps for the fast neural network.

Using sliding window of size  $n \times n$ , for the same image of  $N \times N$  pixels,  $qN^2n^2$  computation steps are required when using traditional neural networks for the face detection process. The theoretical speed up factor  $\eta$  can be evaluated as follows [1]:

$$\eta = \frac{qn^2}{(q+1)\log^2 N} \quad (7)$$

### III. COMMENTS ON FAST NEURAL NET PRESENTED FOR OBJECT/ FACE DETECTION

The speed up factor introduced in [1] and given by Eq.7 is not correct for the following reasons:

1- The number of computation steps required for the 2D-FFT is  $O(N^2 \log_2 N^2)$  and not  $O(N^2 \log^2 N)$  as presented in [1,2]. Also, this is not a typing error as the curve in Fig.2 in [1] realizes Eq.7, and the curves in Fig.15 in [2] realizes Eq.31 and Eq.32 in [2].

2- Also, the speed up ratio presented in [1] not only contains an error but also is not precise. This is because for fast neural networks, the term  $(6qN^2)$  corresponds to complex dot product in the frequency domain must be added. Such term has a great effect on the speed up ratio. Adding only  $qN^2$  as stated in [2] is not correct since a one complex multiplication requires six real computation steps.

3- For conventional neural networks, the number of operations is  $(q(2n^2-1)(N-n+1)^2)$  and not  $(qN^2n^2)$ . The term  $n^2$  is required for multiplication of  $n^2$  elements (in the input window) by  $n^2$  weights which results in another new  $n^2$  elements. Adding these  $n^2$  elements, requires another  $(n^2-1)$  steps. So, the total computation steps needed for each window is  $(2n^2-1)$ . The search operation for a face in the input image uses a window with  $n \times n$  weights. This operation is done at each pixel in the input image. Therefore, such process is repeated  $(N-n+1)^2$  times and not  $N^2$  as stated in [1,3].

4- Before applying cross correlation, the 2D-FFT of the weight matrix must be computed. Because of the dot product, which is done in the frequency domain, the size of weight matrix should be increased to be the same as the size of the input image. Computing the 2D-FFT of the weight matrix off line as stated in [1-3] is not practical. In this case, all of the input images must have the same size. As a result, the input image will have only a one fixed size. This means that, the testing time for an image of size  $50 \times 50$  pixels will be the same as that image of size  $1000 \times 1000$  pixels and of course, this is unreliable. So, another number of complex computation steps to perform 2D-FFT for  $(N \times N)$  matrix should be added to the complex number of computation

steps ( $\sigma$ ) required by the fast neural networks as follows:

$$\sigma = ((2q+1)(N^2 \log_2 N^2) + 6qN^2) \quad (8)$$

This will increase the computation steps required for the fast neural networks especially when  $q$  is more than one neuron.

5- It is not valid to compare number of complex computation steps by another of real computation steps directly. The number of computation steps given by pervious authors [1-3] for conventional neural networks is for real operations while that is required by the fast neural networks is for complex operations. To obtain the speed up ratio, the authors in [1-3] have divided the two formulas directly without converting the number of computation steps required by the fast neural networks into a real version. It is known that the two dimensions Fast Fourier Transform requires  $(N^2/2)\log_2 N^2$  complex multiplications and  $N^2 \log_2 N^2$  complex additions. Every complex multiplication is realized by six real floating point operations and every complex addition is implemented by two real floating point operations. Therefore, the total number of computation steps required to obtain the 2D-FFT of an  $N \times N$  image is:

$$\rho = 6((N^2/2)\log_2 N^2) + 2(N^2 \log_2 N^2) \quad (9)$$

which may be simplified to:

$$\rho = 5(N^2 \log_2 N^2) \quad (10)$$

6- For the weight matrix to have the same size as the input image, a number of zeros  $= (N^2 - n^2)$  must be added to the weight matrix. This requires a total real number of computation steps  $= q(N^2 - n^2)$  for all neurons. Moreover, after computing the 2D-FFT for the weight matrix, the conjugate of this matrix must be obtained. So, a real number of computation steps  $= qN^2$  should be added in order to obtain the conjugate of the weight matrix for all neurons. Also, a number of real computation steps equal to  $N$  is required to create butterflies complex numbers  $(e^{-jk(2\pi n/N)})$ , where  $0 < K < L$ . These  $(N/2)$  complex numbers are multiplied by the elements of the input image or by previous complex numbers during the computation of 2D-FFT. To create a complex number requires two real floating point operations. Thus, the total number of computation steps required by the fast neural networks is:

$$\sigma = ((2q+1)(5N^2 \log_2 N^2) + 6qN^2 + q(N^2 - n^2) + qN^2 + N) \quad (11)$$

which can be reformulated as:

$$\sigma = ((2q+1)(5N^2 \log_2 N^2) + q(8N^2 - n^2) + N) \quad (12)$$

Therefore, the correct speed up ratio is as follows:

$$\eta = \frac{q(2n^2 - 1)(N^2 - n^2 + 1)}{(2q + 1)(5N^2 \log_2 N^2) + q(8N^2 - n^2) + N} \quad (13)$$

The correct theoretical speed up ratio (Eq.13) with different sizes of the input image and different in size

weight matrices is listed in Table 1. Practical speed up ratio for manipulating images of different sizes and different in size weight matrices is listed in Table 2 using 700 MHz processor and Matlab ver 5.3.

For general fast cross correlation the speed up ratio becomes in the following form:

$$\eta = \frac{q(2n^2 - 1)(N^2 - n^2 + 1)}{(2q + 1)(5(N + \tau)^2 \log_2 (N + \tau)^2) + q(8(N + \tau)^2 - n^2) + (N + \tau)} \quad (14)$$

where  $\tau$  is a small number depends on the size of the weight matrix. General cross correlation means that the process starts from the first element in the input matrix. The theoretical speed up ratio for general fast cross correlation (Eq.14) is shown in Table 3. Compared with MATLAB cross correlation function (xcorr2), experimental results show that the our proposed algorithm is faster than this function as shown in Table 4.

7- Furthermore, there are critical errors in Eq.3 and Eq.4 (which is Eq.4 in [1] and also Eq.13 in [2]). Eq.3 is not correct because the definition of cross correlation is:

$$f(x,y) \otimes g(x,y) = \left( \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f(x+m, y+n) g(m,n) \right) \quad (15)$$

and then Eq.4 must be written as follows:

$$h_i = g(\Psi \otimes W_i + b_i) \quad (16)$$

Therefore, the cross correlation in the frequency domain given by Eq.5 does not represent Eq.4. This is because the fact that the operation of cross correlation is not commutative ( $W \otimes \Psi \neq \Psi \otimes W$ ). As a result, Eq.4 does not give the same correct results as conventional neural networks. This error leads the researchers in [22-30] who consider the references [1-3] to think about how to modify the operation of cross correlation so that Eq.4 can give the same correct results as conventional neural networks. Therefore, errors in these equations must be cleared to all the researchers. In [21-30], the authors proved that a symmetry condition must be found in input matrices (images and the weights of neural networks) so that fast neural networks can give the same results as conventional neural networks. In case of symmetry  $W \otimes \Psi = \Psi \otimes W$ , the cross correlation becomes commutative and this is a valuable achievement. In this case, the cross correlation is performed without any constrains on the arrangement of matrices. As presented in [22-30], this symmetry condition is useful for reducing the number of patterns that neural networks will learn. This is because the image is converted into symmetric shape by rotating it down and then the up image and its rotated down version are tested together as one (symmetric) image. If a pattern is detected in the rotated down image, then, this means that this pattern is found at the relative position in the up image. So, if

conventional neural networks are trained for up and rotated down examples of the pattern, fast neural networks will be trained only to up examples. As the number of trained examples is reduced, the number of neurons in the hidden layer will be reduced and the neural network will be faster in the test phase compared with conventional neural networks.

8- Moreover, the authors in [1-3] stated that the activity of each neuron in the hidden layer (Eq.4) can be expressed in terms of convolution between a bank of filter (weights) and the input image. This is not correct because the activity of the hidden neuron is a cross correlation between the input image and the weight matrix. It is known that the result of cross correlation between any two functions is different from their convolution. As we proved in [21-30] the two results will be the same, only when the two matrices are symmetric or at least the weight matrix is symmetric.

9- Images are tested for the presence of a face (object) at different scales by building a pyramid of the input image which generates a set of images at different resolutions. The face detector is then applied at each resolution and this process takes much more time as the number of processing steps will be increased. In [1-3], the authors stated that the Fourier transforms of the new scales do not need to be computed. This is due to a property of the Fourier transform. If  $z(x,y)$  is the original and  $a(x,y)$  is the sub-sampled by a factor of 2 in each direction image then:

$$a(x, y) = z(2x, 2y) \quad (17)$$

$$Z(u, v) = FT(z(x, y)) \quad (18)$$

$$FT(a(x, y)) = A(u, v) = \frac{1}{4} Z\left(\frac{u}{2}, \frac{v}{2}\right) \quad (19)$$

This implies that we do not need to recompute the Fourier transform of the sub-sampled images, as it can be directly obtained from the original Fourier transform. But experimental results have shown that Eq.17 is valid only for images in the following form:

$$\Psi = \begin{bmatrix} A & A & B & B & C & C & \dots & \dots & \dots \\ A & A & B & B & C & C & \dots & \dots & \dots \\ \cdot & & & & & & & & \\ \cdot & & & & & & & & \\ \cdot & & & & & & & & \\ \cdot & & & & & & & & \\ S & S & X & X & Y & Y & \dots & \dots & \dots \\ S & S & X & X & Y & Y & \dots & \dots & \dots \end{bmatrix} \quad (20)$$

In [1], the author claimed that the processing needs  $O((q+2)N^2 \log^2 N)$  additional number of computation steps. Thus the speed up ratio will be [1]:

$$\eta = \frac{qn^2}{(q+2)\log^2 N} \quad (21)$$

Of course this is not correct, because the inverse of the Fourier transform is required to be computed at each neuron in the hidden layer (for the resulted matrix from the dot product between the Fourier matrix in two dimensions of the input image and the Fourier matrix in two dimensions of the weights, the inverse of the Fourier transform must be computed). So, the term  $(q+2)$  in Eq.21 should be  $(2q+1)$  because the inverse 2D-FFT in two dimensions must be done at each neuron in the hidden layer. In this case, the number of computation steps required to perform 2D-FFT for the fast neural networks will be:

$$\varphi = (2q+1)(5N^2 \log_2 N^2) + (2q)5(N/2)^2 \log_2 (N/2)^2 \quad (22)$$

In addition, a number of computation steps equal to  $6q(N/2)^2 + q((N/2)^2 - n^2) + q(N/2)^2$  must be added to the number of computation steps required by the fast neural networks.

#### IV. CONCLUSION

It has been shown that the equations given in [1-3] for conventional and fast neural networks contain errors. The reasons for these errors have been proved. Correct equations for cross correlation in the spatial and frequency domains have been presented. Furthermore, correct equations for the number of computation steps required by conventional, and fast neural networks have been introduced. A new correct formula for the speed up ratio has been established. Also, correct equations for fast multi scale object/face detection have been given. Moreover, commutative cross correlation has been achieved by converting the non-symmetric input matrices into symmetric forms. Theoretical and practical results after these corrections have shown that generally fast neural networks requires fewer computation steps than conventional one. Finally, the proposed model has no shortcomings. This is because the results of conventional and fast neural networks are the same. Therefore, the performance (detection rate) is the same.

#### REFERENCES

- [1] S. Ben-Yacoub, "Fast Object Detection using MLP and FFT," IDIAP-RR 11, IDIAP, 1997.
- [2] B. Fasel, "Fast Multi-Scale Face Detection, " IDIAP-Com 98-04, 1998.
- [3] S. Ben-Yacoub, B. Fasel, and J. Luetin , "Fast Face Detection using MLP and FFT, " in Proc. Second International Conference on Audio and Video-based Biometric Person Authentication (AVBPA'99), 1999.
- [4] Y. Zhu, S. Schwartz, and M. Orchard, "Fast Face Detection Using Subspace Discriminate Wavelet Features," Proc. of IEEE Computer Society International Conference on Computer Vision and Pattern Recognition (CVPR'00), South Carolina, June 13 - 15, 2000, vol.1, pp. 1636-1643.

- [5] H. M. El-bakry, M. A. Abo-elsoud, and M. S. Kamel, "Fast Modular Neural Networks for Human Face Detection," Proc. of IEEE-INNS-ENNS International Joint Conference on Neural Networks, Como, Italy, Vol. III, pp. 320-324, 24-27 July, 2000.
- [6] H. M. El-bakry, "Fast Iris Detection using Cooperative Modular Neural Nets," Proc. of the 6<sup>th</sup> International Conference on Soft Computing, 1-4 Oct., 2000, Japan.
- [7] H. M. El-Bakry, "Automatic Human Face Recognition Using Modular Neural Networks," Machine Graphics & Vision Journal (MG&V), vol. 10, no. 1, 2001, pp. 47-73.
- [8] H. M. El-bakry, "Fast Iris Detection Using Cooperative Modular Neural Networks," Proc. of the 5<sup>th</sup> International Conference on Artificial Neural Nets and Genetic Algorithms, pp. 201-204, 22-25 April, 2001, Sydney, Czech Republic.
- [9] H. M. El-bakry, "Fast Iris Detection Using Neural Nets," Proc. of the 14<sup>th</sup> Canadian Conference on Electrical and Computer Engineering, pp.1409-1415, 13-16 May, 2001, Canada.
- [10] H. M. El-bakry, "Human Iris Detection Using Fast Cooperative Modular Neural Nets," Proc. of INNS-IEEE International Joint Conference on Neural Networks, pp. 577-582, 14-19 July, 2001, Washington, DC, USA.
- [11] H. M. El-bakry, "Human Iris Detection for Information Security Using Fast Neural Nets," Proc. of the 5<sup>th</sup> World Multi-Conference on Systemics, Cybernetics and Informatics, 22-25 July, 2001, Orlando, Florida, USA.
- [12] H. M. El-bakry, "Human Iris Detection for Personal Identification Using Fast Modular Neural Nets," Proc. of the 2001 International Conference on Mathematics and Engineering Techniques in Medicine and Biological Sciences, pp. 112-118, 25-28 July, 2001, Monte Carlo Resort, Las Vegas, Nevada, USA.
- [13] H. M. El-bakry, "Human Face Detection Using Fast Neural Networks and Image Decomposition," Proc. the fifth International Conference on Knowledge-Based Intelligent Information & Engineering Systems 6-8 September 2001, Osaka-kyoiku University, Kashiwara City, Japan, pp. 1330-1334.
- [14] H. M. El-Bakry, "Fast Iris Detection for Personal Verification Using Modular Neural Networks," Proc. of the International Conference on Computational Intelligence, 1-3 Oct., 2001, Dortmund, Germany, pp.-269-283.
- [15] H. M. El-bakry, "Fast Cooperative Modular Neural Nets for Human Face Detection," Proc. of IEEE International Conference on Image Processing, 7-10 Oct., 2001, Thessaloniki, Greece.
- [16] Hazem El-Bakry, "Fast Face Detection Using Neural Networks and Image Decomposition," Proc. of the 6<sup>th</sup> International Computer Science Conference, Active Media Technology, Dec. 18-20, 2001, Hong Kong – China, pp. 205-215, 2001.
- [17] H. M. El-Bakry, "Face detection using fast neural networks and image decomposition," Neurocomputing Journal, vol. 48, 2002, pp. 1039-1046.
- [18] Hazem El-Bakry, "Face Detection Using Fast Neural Networks and Image Decomposition," Proc. of INNS-IEEE International Joint Conference on Neural Networks, 14-19 May, 2002, Honolulu, Hawaii, USA.
- [19] S. Srisuk and W. Kurutach, "A New Robust Face Detection in Color Images," Proc. of IEEE Computer Society International Conference on Automatic Face and Gesture Recognition, Washington D.C., USA, May 20-21, 2002, pp. 306-311.
- [20] H. M. El-Bakry, "Human Iris Detection Using Fast Cooperative Modular Neural Networks and Image Decomposition," Machine Graphics & Vision Journal (MG&V), vol. 11, no. 4, 2002, pp. 498-512.
- [21] H. M. El-Bakry, "Comments on Using MLP and FFT for Fast Object/Face Detection," Proc. of IEEE IJCNN'03, Portland, Oregon, pp. 1284-1288, July, 20-24, 2003.
- [22] H. M. El-Bakry, and H. Stoyan, "Fast Neural Networks for Object/Face Detection," Proc. of the 30<sup>th</sup> Anniversary SOFSEM Conference on Current Trends in Theory and Practice of Computer Science, 24-30 January, 2004, Hotel VZ MERIN, Czech Republic.
- [23] H. M. El-Bakry, and H. Stoyan, "Fast Neural Networks for Sub-Matrix (Object/Face) Detection," Proc. of IEEE International Symposium on Circuits and Systems, Vancouver, Canada, 23-26 May, 2004.
- [24] H. M. El-Bakry, "Fast Sub-Image Detection Using Neural Networks and Cross Correlation in Frequency Domain," Proc. of IS 2004: 14<sup>th</sup> Annual Canadian Conference on Intelligent Systems, Ottawa, Ontario, 6-8 June, 2004.
- [25] H. M. El-Bakry, and H. Stoyan, "Fast Neural Networks for Code Detection in a Stream of Sequential Data," Proc. of CIC 2004 International Conference on Communications in Computing, Las Vegas, Nevada, USA, 21-24 June, 2004.
- [26] H. M. El-Bakry, "Fast Neural Networks for Object/Face Detection," Proc. of 5<sup>th</sup> International Symposium on Soft Computing for Industry with Applications of Financial Engineering, June 28 - July 4, 2004, Sevilla, Andalucia, Spain.
- [27] H. M. El-Bakry, and H. Stoyan, "A Fast Searching Algorithm for Sub-Image (Object/Face) Detection Using Neural Networks," Proc. of the 8<sup>th</sup> World Multi-Conference on Systemics, Cybernetics and Informatics, 18-21 July, 2004, Orlando, USA.
- [28] H. M. El-Bakry, and H. Stoyan, "Fast Neural Networks for Code Detection in Sequential Data Using Neural Networks for Communication Applications," Proc. of the First International Conference on Cybernetics and Information Technologies, Systems and Applications: CITSA 2004, 21-25 July, 2004, Orlando, Florida, USA, Vol. IV, pp. 150-153.
- [29] H. M. El-Bakry, and Q. Zhao, "A New Symmetric Form for Fast Sub-Matrix (Object/Face) Detection Using Neural Networks and FFT," accepted and under publication in the International Journal of Signal Processing.
- [30] H. M. El-Bakry, and Q. Zhao, "Fast Pattern Detection Using Normalized Neural Networks and Cross Correlation in the Frequency Domain," accepted and under publication in the EURASIP Journal on Applied Signal Processing.

Table 1: The theoretical speed up ratio for images with different sizes.

Image size	Speed up ratio (n=20)	Speed up ratio (n=25)	Speed up ratio (n=30)
100x100	3.67	5.04	6.34
200x200	4.01	5.92	8.05
300x300	4.00	6.03	8.37
400x400	3.95	6.01	8.42
500x500	3.89	5.95	8.39
600x600	3.83	5.88	8.33
700x700	3.78	5.82	8.26
800x800	3.73	5.76	8.19
900x900	3.69	5.70	8.12
1000x1000	3.65	5.65	8.05

Table 2: Practical Speed up ratio for images with different sizes Using MATLAB ver 5.3.

Image size	Speed up ratio (n=20)	Speed up ratio (n=25)	Speed up ratio (n=30)
100x100	7.88	10.75	14.69
200x200	6.21	9.19	13.17
300x300	5.54	8.43	12.21
400x400	4.78	7.45	11.41
500x500	4.68	7.13	10.79
600x600	4.46	6.97	10.28
700x700	4.34	6.83	9.81
800x800	4.27	6.68	9.60
900x900	4.31	6.79	9.72
1000x1000	4.19	6.59	9.46

Table 3: The theoretical speed up ratio for the general fast cross correlation algorithm.

Image size	Speed up ratio (n=20)	Speed up ratio (n=25)	Speed up ratio (n=30)
100x100	5.39	8.36	11.95
200x200	4.81	7.49	10.75
300x300	4.51	7.03	10.16
400x400	4.32	6.73	9.68
500x500	4.18	6.52	9.37
600x600	4.07	6.35	9.13
700x700	3.99	6.21	8.94
800x800	3.91	6.10	8.77
900x900	3.84	6.00	8.63
1000x1000	3.78	5.91	8.51

Table 4: Simulation results of the speed up ratio for the general fast cross correlation compared with the MATLAB cross correlation function (xcorr2).

Image size	Speed up ratio (n=20)	Speed up ratio (n=25)	Speed up ratio (n=30)
100x100	10.14	13.05	16.49
200x200	9.17	11.92	14.33
300x300	8.25	10.83	13.41
400x400	7.91	9.62	12.65
500x500	6.77	9.24	11.77
600x600	6.46	8.89	11.19
700x700	5.99	8.47	10.96
800x800	5.48	8.74	10.32
900x900	5.31	8.43	10.66
1000x1000	5.91	8.66	10.51

# CONSTRUCTING FUZZY CLASSIFICATION SYSTEMS FROM WEIGHTED TRAINING PATTERNS

Tomoharu Nakashima, Yasuyuki Yokota  
Hisao Ishibuchi

College of Engineering  
Osaka Prefecture University  
Gakuen-cho 1-1, Sakai, Osaka 599-8531, Japan  
{nakashi, yokota, hisaoi}@ie.osakafu-u.ac.jp

Andrzej Bargiela

Department of Computing  
The Nottingham Trent University  
Burton Street, Nottingham NG1 4BU, U.K.  
andre@doc.ntu.ac.uk

## KEYWORDS

Fuzzy rule-based systems, pattern classification, data mining, classification cost

## ABSTRACT

In this paper, we propose a fuzzy rule-generation method for pattern classification problems. We consider a situation where each training pattern has a weight. The weight is considered as a cost of misclassification/rejection of classification. Our fuzzy classification system consists of a set of fuzzy if-then rules. The antecedent part of fuzzy if-then rules linguistically specifies a subarea of a pattern space. Thus, fuzzy if-then rules are linguistically interpretable. The main aim of this paper is to construct fuzzy classification systems that reflect weighted training patterns. That is, we make the cost of misclassification and rejection of classification for unseen patterns as small as possible. In computer simulations, we also propose two methods for assigning appropriate weights from the distribution of given training patterns. The performance of the proposed fuzzy rule-generation method is examined for several real-world pattern classification systems that have been used in literature.

## INTRODUCTION

Fuzzy rule-based systems have been applied mainly to control problems (Lee 1990, Leondes 1999, Sugeno 1985). One advantage of a fuzzy rule-based system is its interpretability. Recently fuzzy rule-based systems have also been applied to pattern classification problems. There are many approaches to the automatic generation of fuzzy if-then rules from numerical data for pattern classification problems. Genetic algorithms have also been used for generating fuzzy if-then rules for pattern classification (Ishibuchi and Nakashima 1999a, Ishibuchi Nakashima 1999b, Ishibuchi et al. 1995, Yuan and Zhan 1996).

In this paper, we propose a fuzzy rule-generation method for pattern classification problems where a weight is assigned to each given training pattern. We assume that weights of training patterns are given before the actual fuzzy classification systems are constructed. The proposed method incorporates the weights in

generating fuzzy if-then rules as well as the compatibility of training patterns.

We also propose weight assignment techniques for our computer simulations as in many benchmark problems weights of training patterns are not available.

A series of computer simulations examine the advantage of our proposed method over the conventional method. We also show then one of the weight assignment techniques is useful for obtaining fuzzy classification systems with high generalization ability.

## FUZZY RULE-BASED CLASSIFICATION

### Pattern Classification Problems

Various methods have been proposed for fuzzy classification (Nozaki et al. 1996, Klir and Yuan 1995, Ishibuchi et al. 1992, Grabisch 1996, Grabisch and Dispot 1992, Grabixch and Nicolas 1994). Let us assume that our pattern classification problem is an  $n$ -dimensional problem with  $C$  classes and  $m$  given training patterns  $\mathbf{x}_p = (x_{p1}, x_{p2}, \dots, x_{pn})$ ,  $p = 1, 2, \dots, m$ . Without loss of generality, each attribute of the given training patterns is normalized into a unit interval  $[0, 1]$ . That is, the pattern space is an  $n$ -dimensional unit hypercube  $[0, 1]^n$  in our pattern classification problem.

In this study we use fuzzy if-then rules of the following type as a base of our fuzzy rule-based classification systems:

$$\begin{aligned} \text{Rule } R_j : & \text{ If } x_1 \text{ is } A_{j1} \text{ and } \dots \text{ and } x_n \text{ is } A_{jn} \\ & \text{ then Class } C_j \text{ with } CF_j, \quad j = 1, 2, \dots, N, \end{aligned} \quad (1)$$

where  $R_j$  is the label of the  $j$ -th fuzzy if-then rule,  $A_{j1}, \dots, A_{jn}$  are antecedent fuzzy sets on the unit interval  $[0, 1]$ ,  $C_j$  is a consequent class (i.e. one of the  $C$  given classes),  $CF_j$  is the grade of certainty of the fuzzy if-then rule  $R_j$ , and  $N$  is the total number of fuzzy if-then rules. As antecedent fuzzy sets, we use triangular fuzzy sets as in Fig. 1 where we show various partitions of the unit interval into a number of fuzzy sets.

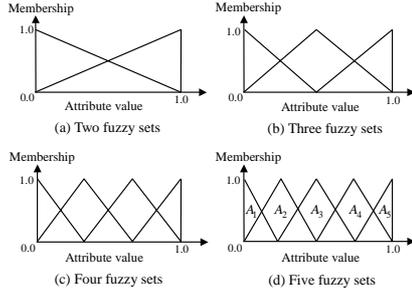


Fig. 1 An example of antecedent fuzzy

### Generating Fuzzy If-Then Rules

In our fuzzy rule-based classification systems, we specify the consequent class and the grade of certainty of each fuzzy if-then rule from the given training patterns (Ishibuchi et al. 1992). In (Ishibuchi and Nakashima 2001) it is shown that the use of the grade of certainty in fuzzy if-then rules allows us to generate comprehensible fuzzy rule-based classification systems with high classification performance.

The consequent class  $C_j$  and the grade of certainty  $CF_j$  of fuzzy if-then rule are determined in the following manner:

#### [Generation Procedure of Fuzzy If-Then Rule]

Step 1: Calculate  $\beta_{\text{Class } h}(R_j)$  for Class  $h$  as

$$\beta_{\text{Class } h}(R_j) = \sum_{x_p \in \text{Class } h} \mu_{j1}(x_{p1}) \cdot \dots \cdot \mu_{jn}(x_{pn}),$$

$$h = 1, 2, \dots, C, \quad (2)$$

where  $\mu_{ji}(\cdot)$  is the membership function of the fuzzy set  $A_{ji}$ .

Step 2: Find Class  $\hat{h}$  that has the maximum value of  $\beta_{\text{Class } h}(R_j)$ :

$$\beta_{\text{Class } \hat{h}}(R_j) = \max\{\beta_{\text{Class } 1}(R_j), \dots, \beta_{\text{Class } C}(R_j)\}. \quad (3)$$

If two or more classes take the maximum value, the consequent class  $C_j$  of the rule  $R_j$  can not be determined uniquely. In this case, specify  $C_j$  as  $C_j = \phi$ . If a single class takes the maximum value, let  $C_j$  be Class  $\hat{h}$ . The grade of certainty  $CF_j$  is determined as

$$CF_j = \frac{\beta_{\text{Class } \hat{h}}(R_j) - \bar{\beta}}{\sum \beta_{\text{Class } h}(R_j)}, \quad (4)$$

where

$$\bar{\beta} = \frac{\sum_{h \neq \hat{h}} \beta_{\text{Class } h}(R_j)}{c-1}. \quad (5)$$

The number of fuzzy if-then rules in a fuzzy rule-based classification system depends on how each attribute is partitioned into fuzzy subsets. For example, when we divide each attribute into three fuzzy subsets in a ten-dimensional pattern classification problem, the total number of fuzzy if-then rules is  $3^{10} = 59049$ . This is what is called *the curse of dimensionality*. The grade of certainty  $CF_j$  can be adjusted by a learning algorithm (Nozaki et al. 1996).

### Fuzzy Reasoning

By the rule generation procedure in the last subsection, we can generate  $N$  fuzzy if-then rules in (1). After both the consequent class  $C_j$  and the grade of certainty  $CF_j$  are determined for all  $N$  rules, a new pattern  $\mathbf{x}$  is classified by the following procedure:

#### [Fuzzy reasoning procedure for classification]

Step 1: Calculate  $\alpha_{\text{Class } h}(\mathbf{x})$  for Class  $h$ ,  $j = 1, 2, \dots, C$ , as

$$\alpha_{\text{Class } h}(\mathbf{x}) = \max\{\mu_j(\mathbf{x}) \cdot CF_j \mid C_j = \text{Class } h\},$$

$$h = 1, 2, \dots, C, \quad (6)$$

where

$$\mu_j(\mathbf{x}) = \mu_{j1}(x_1) \cdot \dots \cdot \mu_{jn}(x_n). \quad (7)$$

Step 2: Find Class  $h'$  that has the maximum value of  $\alpha_{\text{Class } h}(\mathbf{x})$ :

$$\alpha_{\text{Class } h'}(\mathbf{x}) = \max\{\alpha_{\text{Class } 1}(\mathbf{x}), \dots, \alpha_{\text{Class } C}(\mathbf{x})\}. \quad (8)$$

If two or more classes take the maximum value, then the classification of  $\mathbf{x}$  is rejected (i.e.  $\mathbf{x}$  is left as an unclassifiable pattern), otherwise assign  $\mathbf{x}$  to Class  $h'$ .

## PROPOSED METHOD

In this section we propose a fuzzy rule-generation method for pattern classification problems where a weight is given for each training pattern. First we explain the role of weights and then we present how fuzzy if-then rules are generated from weighted training patterns.

### Concept of Weight

Let us consider an example of classification problems in which the idea of weighted training patterns is necessary. In the medical diagnosis of cancer two kinds of misclassification have to be taken into account. One is the case where a person is diagnosed as having cancer while he/she does not. The other case is that a person with cancer is classified as not having cancer. Although

the misclassification should be as small as possible in both cases, the latter misclassification should be treated more seriously than the former case. In this paper we use the concept of weights to tackle this problem.

### Cost Function

The weight of training patterns can be viewed as the importance of the patterns. We place more emphasis on those patterns with large weights than on those with small weights. The weight of misclassified/rejected patterns is considered as a cost of misclassification or rejection. Thus, our objective in this paper is to construct a fuzzy classification system  $S$  that minimizes the following cost function :

$$Cost(S) = \sum_{p=1}^m w_p \cdot z_p(S), \quad (9)$$

where  $Cost(S)$  is the cost of misclassification/rejection made by a fuzzy classification system  $S$ ,  $m$  is the number of training patterns,  $w_p$  is the weight of the training pattern  $\mathbf{x}_p$ , and  $z_p(S)$  is a binary variable that is determined according to the classification result of the training pattern  $\mathbf{x}_p$  by  $S$ :  $z_p(S) = 0$  if  $\mathbf{x}_p$  is correctly classified by  $S$ , and  $z_p(S) = 1$  otherwise.

### Generating Fuzzy If-Then Rules from Weighted Training Patterns

Let us assume that we have  $m$  training patterns  $\mathbf{x}_p = (x_{p1}, x_{p2}, \dots, x_{pn})$ ,  $p = 1, 2, \dots, m$ , and we also assume that a weight is given a priori for all training patterns. We modify the fuzzy rule generation procedure presented in the last section to accommodate the weighted training patterns :

#### [Generation Procedure of Fuzzy If-Then Rule]

Step 1: Calculate  $\beta_{Class\ h}(R_j)$  for Class  $h$  ( $h = 1, \dots, C$ ) as

$$\beta_{Class\ h}(R_j) = \sum_{\mathbf{x}_p \in Class\ h} \mu_{j1}(x_{p1}) \cdot \dots \cdot \mu_{jn}(x_{pn}) \cdot w_p, \quad (10)$$

$$h = 1, 2, \dots, C.$$

Step 2: Find Class  $\hat{h}$  that has the maximum value of  $\beta_{Class\ h}(R_j)$  :

$$\beta_{Class\ \hat{h}}(R_j) = \max\{\beta_{Class\ 1}(R_j), \dots, \beta_{Class\ C}(R_j)\}. \quad (11)$$

This procedure is the same as the one that is used in the last section except for (10). In order to focus on those training patterns with large weights, we include the weight in the calculation of the compatibility of training patterns with the antecedent part of fuzzy if-then rules when we determine the consequent class of the fuzzy if-then rule.

## WEIGHT ASSIGNMENT

In the next section we examine the performance of the proposed method by computer simulation. We use real-world pattern classification problems that are commonly used in literature. All the classification problems are available from the UCI machine learning repository. Since weights for training patterns are not included in these pattern classification sets, we propose two methods for weighting training patterns in order to make a synthetic situation where a weight is given to each training pattern. One is a class-based weighting method and the other is an overlap-based weighting method.

### Class-Based Weighting Method

The aim of the class-based weighting method is to make a bias toward the classification of patterns from a particular class. For example, if the bias is toward the classification of Class 1 patterns, classification systems are expected to correctly classify Class 1 patterns even if the number of misclassification/rejection is large for other classes. Medical diagnoses is example appropriate for this weighting method.

In this weighting method, a weight for the pattern  $\mathbf{x}_p$  is determined by the following equation :

$$w_p = \begin{cases} 1.0 & \text{if class of } \mathbf{x}_p \text{ is to be emphasized,} \\ 0.5 & \text{otherwise.} \end{cases} \quad (12)$$

### Overlap-Based Weighting Method

The aim of the overlap-based weighting method is to focus on overlapped areas between multiple classes. In order to determine the weights of given training patterns, we count the number of patterns from the same class in their neighborhood. Let us denote the neighborhood size as  $N_{size}$ . We examine  $N_{size}$  nearest patterns from each of given training patterns for determining the value of the weight. In the overlap-based weighting method we use the following equation to determine the weight of the  $p$ -th given pattern  $w_p$  :

$$w_p = \frac{N_p^{same}}{N_p^{nearest}} \quad (13)$$

where  $N_p^{nearest}$  is the number of nearest training patterns to  $\mathbf{x}_p$  and  $N_p^{same}$  is the number of given patterns from the same class as the  $p$ -th given pattern. The weight  $w_p$  of the  $p$ -th given pattern can be viewed as a measure of overlaps. That is, if the value of  $w_p$  is large, there are many patterns from the same class as  $p$ -th training pattern. On the other hand, the  $p$ -th given pattern is possibly an outlier if the value of  $w_p$  is low.

## COMPUTER SIMULATIONS

We examined the performance of the proposed method for eight real-world pattern classification problems that are available from the UCI machine learning repository. We show the details of the nine classification problems in Table 1.

Table 1 Classification problems

Data set	Attributes	Classes	Patterns
Balance scale	4	3	625
Breast cancer	9	2	683
CMC	9	3	1473
Glass	9	7	214
Haberman	3	2	306
Hayes roth	4	3	132
Iris	4	3	150
Wine	13	3	178

In the following subsections we show the comparison of performance between conventional fuzzy classification systems and the proposed fuzzy systems. As discussed the difference between the conventional and the proposed method is that fuzzy if-then rules are generated by using (2) in the case of the conventional fuzzy classification systems and by (10) in the case of the proposed one. In both cases, we partition each attribute of classification problems into three fuzzy sets (see Fig.1(b)). That is, the total number of fuzzy if-then rules generated for an  $n$ -dimensional pattern classification problem is  $3^n$ .

### Random Weighting

In a preliminary experiments we examined the performance of the proposed method under the situation that weights of training patterns are randomly determined. A weight was determined by a uniform random number in the interval of  $[0, 1]$ . That is, we randomly assign the importance of classification of training patterns in this subsection.

From these randomly weighted training patterns we generate fuzzy if-then rules by using (10) to construct a fuzzy classification system. The classification ability of the system was examined for all given training patterns. We iterated the procedure 100 times. Note that we did not change the attribute values of training patterns but the value of their weights. That is, we examined the classification ability of fuzzy classification systems with 100 different sets of weights for training patterns. It should also be noted that the classification ability of the fuzzy classification that are generated by the conventional method (i.e., using (2)) is constant for the 100 iterations as the fuzzy rule-generation process does not consider the weights of training patterns.

We show the classification results of both the proposed and the conventional fuzzy classification systems in Table 2. Table 3 shows the cost of misclassification/rejection. From Table 2 and Table 3, we can see that the cost of misclassification/rejection is reduced by the proposed method while the number of correctly classified patterns by the proposed method is smaller than that by the conventional method. The reason of the reduction of the number of correctly classified patterns is that the fuzzy classification system constructed by the proposed method focuses on important training patterns with large weights.

Table 2 Classification results (Random weights)

Data set	Proposed	Conventional
Balance scale	91.5%	91.2%
Breast cancer	98.2%	98.2%
CMC	54.2%	55.4%
Glass	71.0%	72.0%
Haberman	73.8%	74.2%
Hayes roth	81.1%	86.4%
Iris	93.2%	94.0%
Wine	99.0%	98.9%

Table 3 Costs (Random weights)

Data set	Proposed	Conventional
Balance scale	24.5	27.3
Breast cancer	5.1	6.0
CMC	318.2	327.8
Glass	29.6	30.3
Haberman	39.8	39.5
Hayes roth	10.5	9.1
Iris	5.1	4.6
Wine	0.8	1.0

### Class-Based Weighting

In this subsection we examine the case where the weights of training patterns are determined by using the class-based weighting method. That is, in this weighting method it is assumed that there is a priority of classification. We use the Haberman and Breast cancer data sets because they are related to medical diagnosis and are hence suitable for this weighting method. Both are two-classification problems. First, we focus on Class 1 training patterns. That is, we set  $w_p = 1.0$  for Class 1 training patterns and  $w_p = 0.5$  for Class 2 patterns. Table 4 and Table 5 show the classification results and the cost of misclassification /rejection, respectively.

We show in Table 6 and Table 7 the performance of fuzzy classification systems in the case where Class 2 training patterns are more focused on than Class 1 training patterns. From Tables 4-7, we can see that the cost of misclassification/rejection is reduced by the proposed method.

Table 4 Classification results (Class 1 focused)

Data set	Proposed	Conventional
Breast cancer	98.4%	98.2%
Haberman	73.9%	74.2%

Table 5 Costs (Class 1 focused)

Data set	Proposed	Conventional
Breast cancer	7	9
Haberman	40	40.5

Table 6 Classification results (Class 2 focused)

Data set	Proposed	Conventional
Breast cancer	98.4%	98.2%
Haberman	73.9%	74.2%

Table 7 Costs (Class 2 focused)

Data set	Proposed	Conventional
Breast cancer	7.5	9
Haberman	62.5	78.9

### Overlap-Based Weighting

We examined the performance of our fuzzy classification systems for the case where overlap-based weighting method is used for training patterns. For each pattern classification problem, we specify the number of nearest training patterns as  $N_p^{\text{nearest}} = 50$ . That is, the proportion of the number of patterns from the same class to its 50 nearest patterns is used as the weight of the training pattern.

We examined the performance of the proposed method and the conventional one for all eight pattern classification problems from Table 1. Classification results and costs of misclassification/reject are given in Table 8 and Table 9. From these tables, we can see that the performance of the proposed method is not better than that of the conventional one in terms of both classification results and the cost of misclassification/rejection. This is because many important training patterns with large weights are near the boundary area and it is difficult to correctly classify all important training patterns.

Table 8 Classification results (Overlap weighting)

Data set	Proposed	Conventional
Balance scale	90.9%	91.2%
Breast cancer	97.8%	98.2%
CMC	58.7%	60.8%
Glass	66.4%	72.0%
Haberman	74.2%	74.2%
Hayes roth	85.6%	86.4%
Iris	93.3%	94.0%
Wine	98.3%	98.9%

Table 9 Costs (Overlap weighting)

Data set	Proposed	Conventional
Balance scale	9.0	8.7
Breast cancer	2.3	2.4
CMC	189.6	184.2
Glass	17.8	15.9
Haberman	23.3	23.0
Hayes roth	7.1	6.8
Iris	4.4	3.8
Wine	1.7	1.3

Next, we examined the performance for test patterns. We used only 20% of the weighted training patterns to construct the fuzzy classification system. The other 80% were used as test patterns. That is, these test patterns are used to check the performance of the system for correctly classifying unseen patterns and for minimizing the cost function for unseen patterns. We conducted this kind of computer simulations 100 times, i.e. 100 different 20%-80% partitions of weighted patterns were used. We show the result of these computer simulations in Table 10 and Table 11. From these tables we can see that the performance of the proposed fuzzy rule-generation method for test patterns was improved.

Table 10 Classification results for test patterns

Data set	Proposed	Conventional
Balance scale	85.3%	84.3%
Breast cancer	93.8%	93.5%
CMC	45.3%	42.8%
Glass	57.4%	56.5%
Haberman	73.7%	73.6%
Hayes roth	43.0%	41.6%
Iris	91.1%	92.1%
Wine	89.1%	87.5%

Table 11 Costs for test patterns

Data set	Proposed	Conventional
Balance scale	23.9	27.5
Breast cancer	19.7	21.5
CMC	219.5	240.7
Glass	20.3	21.7
Haberman	18.1	18.6
Hayes roth	23.3	24.2
Iris	5.2	4.6
Wine	9.7	12.0

### CONCLUSIONS

In this paper we proposed a fuzzy rule-generation method from weighted training patterns. Weights of patterns can be viewed as the grade of importance in the classification. The weights were treated as the costs of training misclassification/rejection. We formulate the problem of constructing classification problems from weighted training patterns as minimization of a cost function. Fuzzy if-then rules are generated by first

specifying the antecedent fuzzy sets and then determining the consequent class and grade of certainty. The determination of the consequent class and the grade of certainty of a fuzzy if-then rule is done by using compatible training patterns with the antecedent part of the fuzzy if-then rule. The proposed method considers the weight of training patterns in this process while the conventional one does not.

We also proposed two methods for weighting training patterns. One is a class-based weighting method where the weights of training patterns are determined by their class. This method can be used when different misclassifications should be distinguished or when patterns from some particular class should be focused on. The other method for weighting patterns is a overlap-based weighting method. This method can be used to focus on vague patterns that are placed near boundary areas and to obtain high generalization ability.

In computer simulations, we examined the performance of several fuzzy classification problems. We compared the performance of the proposed method and that of the conventional method. It was shown that the cost of misclassification/rejection was reduced by the proposed method.

Future work will incorporate a learning method of the grade of certainty. Since the determination of the grade of certainty is heuristically determined, it has to be optimized. We can consider two kinds of optimization for the learning of the grade of certainty. One is the optimization of the classification ability. That is, we can modify the grade of certainty so that the classification rate is maximized. In this case, the optimization function is formulated based on the number of correctly classified training patterns. This method is related to the learning method for the conventional fuzzy rule-generation method (see Nozaki et al. 1996). The other is the optimization of the cost of misclassification/rejection. That is, the modification of the grade of certainty is performed to minimize the cost. The decision of which optimization is used is dependent on the users' choice in real-application.

Another future work is how to optimize the grade of certainty. There are two main streams in the learning of fuzzy rules. One is a local learning method where optimization is performed by considering just one single fuzzy rule to be optimized. In this case, the optimal value of the grade of certainty is easily calculated from the formulation of optimization problem. However, this optimal value does not guarantee the globally optimum performance of fuzzy classification system as a whole. The other is a global learning method where the modification of the grade of certainty is performed from the classification results. This method is likely to produce the optimum fuzzy classification method, however, it is difficult to formulate the optimization problem mathematically and to calculate the optimal value of the grades of certainty.

## REFERENCES

- M. Grabisch. 1996, "The representation of importance and interaction of features by fuzzy measures", *Pattern Recognition Letters*, Vol. 17:567-575.
- M. Grabisch, F. Dispot. 1992, "A comparison of some methods of fuzzy classification on real data", *Proc. of 2nd Intl. Conf. on Fuzzy Logic and Neural Networks*, 659-662.
- M. Grabisch, and J.-M. Nicolas. 1994, "Classification by fuzzy integral: performance and tests", *Fuzzy Sets and Systems*, Vol. 65, No. 2/3:255-271.
- H. Ishibuchi, T. Nakashima. 1999a, "Performance evaluation of fuzzy classifier systems for multi-dimensional pattern classification problems", *IEEE Trans. on Syst., Man, Cybernetics*, Part B Vol. 29:601-618.
- H. Ishibuchi, T. Nakashima. 1999b, "Improving the performance of fuzzy classifier systems for pattern classification problems with continuous attributes", *IEEE Trans. on Industrial Electronics*, Vol. 46, No. 6:1057- 1068.
- H. Ishibuchi, T. Nakashima. 2001, "T. Effect of rule weights in fuzzy rule-based classification systems", *IEEE Trans. on Fuzzy Systems*, Vol. 9, No. 4:506-515.
- H. Ishibuchi, K. Nozaki, H. Tanaka. 1992, "Distributed representation of fuzzy rules and its application to pattern classification", *Fuzzy Sets and Systems*, Vol. 52, No. 1:21-32.
- H. Ishibuchi, K. Nozaki, N. Yamamoto, H. Tanaka. 1995, "Selecting fuzzy if-then rules for classification problems using genetic algorithms", *IEEE Trans. on Fuzzy Systems*, Vol. 3, No. 3:260-270.
- G.J. Klir, B. Yuan. 1995, *Fuzzy Sets and Fuzzy Logic*, Prentice-Hall.
- C.C. Lee. 1990, "Fuzzy Logic in Control Systems: Fuzzy Logic Controller Part I and Part II", *IEEE Trans. Syst., Man, Cybernetics*, Vol. 20:404-435.
- C.T. Leondes. 1999, "Fuzzy theory Systems", *Techniques and Applications. Academic Press*, San Diego, Vol.1-4, 1999.
- T. Nakashima, H. Ishibuchi, A. Bargiela. 2004, "Constructing Fuzzy Classification Systems from Weighted Training Patterns," *Proc. of 2004 IEEE International Conference on Systems, Man and Cybernetics*, pp. 2386-2391.
- K. Nozaki, H. Ishibuchi, H. Tanaka. 1996, "Adaptive fuzzy rule-based classification systems", *IEEE Trans. on Fuzzy Systems*, Vol. 4, No. 3:238-250.
- M. Sugeno. 1985, "An Introductory Survey of Fuzzy Control", *Information Science*, Vol. 30, No. 1/2:59-83.
- Y. Yuan, H. Zhang. 1996, "A genetic algorithms for generating fuzzy classification rules", *Fuzzy Sets and Systems*, Vol. 84, No. 1:1-19.

# NTUNE – AN EDUCATIONAL NEURAL NETWORK SIMULATOR

Michał Czardybon<sup>1</sup>, Lars Nolle<sup>2</sup> and Gerald Schaefer<sup>2</sup>

<sup>1</sup>Silesian University of Technology  
Faculty of Automatic Control, Electronics and Computer Science  
Wydział Automatyki, Elektroniki i Informatyki  
ul. Akademicka 16, 44-100 Gliwice, Poland

<sup>2</sup>School of Computing and Informatics  
The Nottingham Trent University  
Burton Street, Nottingham, NG1 4BU, UK  
lars.nolle@ntu.ac.uk

## KEYWORDS

Artificial Neural Networks, Simulator, Education.

## ABSTRACT

This paper describes NTUNE (Nottingham Trent University Neural Environment), a multi-platform neural network simulation tool for feed-forward network architectures, which was especially designed as an educational tool for teaching purposes in higher education. NTUNE can easily be extended and trained networks can be exported to be used in other C++ applications.

## INTRODUCTION

When looking for a tool to be used for teaching undergraduates the principles of standard feed-forward neural networks (Picton, 2000), it became eminent that freely available tools such as SNNS (Zell, 1994) were designed for experts, who would apply them to real problems, rather than for students to learn the principles of neural networks. Hence, these tools tend to be very complex, which means that undergraduates would have to spend a long time to learn how to use these tools rather than to learn about neural networks. The aim of this research was to develop an easy to use simulation tool for standard artificial neural networks for teaching purposes in higher education.

### Artificial Neuronal Networks

First models of artificial neural networks were introduced in 1943 by Warren McCulloch and Walter Pitts (1943) but it took another 20 years to overcome the problems of their first approach and to find appropriate learning methods. Since then, applications of artificial neural networks have grown rapidly in a wide area of science and industry.

Artificial neural networks simulate the biological structure of neural networks in brains in a simplified way to endow computers with the very considerable abilities of biological neural networks: the ability of learning from examples, pattern classification, generalization and prediction.

Both artificial and biological neural networks consist of a large number of simple processing elements called units or nodes which are connected to other processing elements. The great performance of neural networks stems from this massive parallelism (Rummelhart and McClelland, 1986).

It was proven by Hornik, et al. (1989) that “... standard multilayer feedforward networks with as few as one hidden layer using arbitrary squashing functions are capable of approximating any Borel measurable function from one finite dimensional space to another to any desired degree of accuracy ...”

Hornik also provided evidence that such networks “... are capable of arbitrarily accurate approximation to a function and its derivatives” (Hornik, 1991). Therefore, feed-forward networks can be used to approximate a given process by presenting suitable examples to it, provided a deterministic relationship exists between the input parameter and the output parameter. Because of this, the feed-forward network architecture is the most commonly used type of artificial neural networks. Therefore, the network simulator was implemented for this network type.

### Feed-forward networks

A feed-forward network can be seen as a black box with defined and directed inputs and outputs (Figure 1).

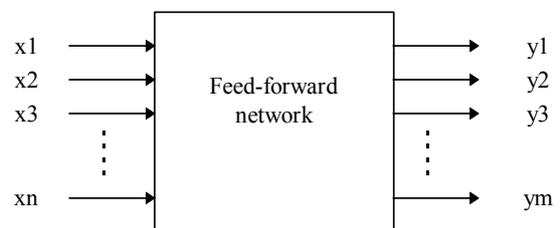


Figure 1 – Feed-forward network as black box.

That means the information is fed through the net in only one direction without any feedback loops. Figure 2 shows a single processing element of a feed-forward network.

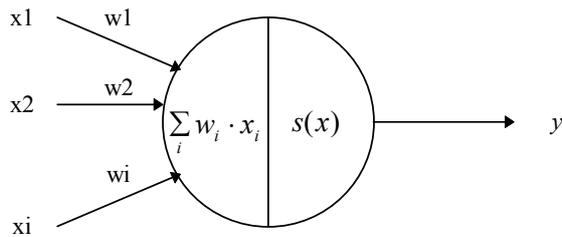


Figure 2 – Processing element.

The output of the processing element is calculated by summing up all inputs multiplied with their associated weight (Equation 1) and passing the sum through a non-linear transfer function, usually a sigmoid or tangent hyperbolic function (Equation 2).

$$x = \sum_i x_i \cdot w_i \quad (1)$$

$$y = s(x) = \frac{1}{1 + e^{-x}} \quad (2)$$

The processing elements are arranged in layers, one input layer, one or more hidden layers - so called because they are ‘invisible’ from the outside - and one output layer. Each output of each unit in one layer is connected usually with every unit in the next layer. Units within the same layer are not connected to each other (Figure 3).

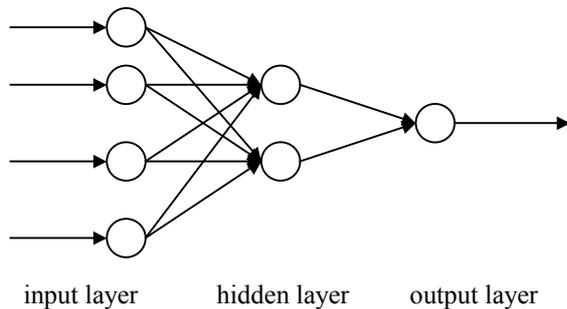


Figure 3 – Feed-forward network with one hidden layer.

The networks work in two modes: the training mode and the testing mode. In training mode, a set of training input patterns together with the demanded output pattern is used to adjust the weights in order to minimize the error of the produced network output and the demanded network output. In test mode, new input patterns can be presented to the network and the network output is used in the application.

## Requirements

The following requirements were identified before the development began. The system should have an easy to use graphical user interface. The model error and the correlation between the model output and the expected output should be plotted in real-time. The system should be easily extendible. It should be possible to save trained networks in order to use them in other applications, and it should also be possible to

export the plot data so that it could be processed with other software products. Another requirement was that the simulator should be available for different computer platforms, e.g. MS Windows and Linux. Finally, in order to allow the software to be released as an Open Source package under the GNU Public License, no commercial tools or libraries could be used.

## NTUNE

NTUNE is a software package consisting of a C++ library and a multi-platform graphical tool for multi-layer feed forward neural networks. The graphical user interface is developed using the wxWidgets library (Smart, 2003), which is available for many different operating systems under the GPL. NTUNE can be used to create and train neural networks in a graphical environment using data provided by the user. The accompanying C++ class library can then be employed to use the trained networks in other programs. Alternatively, the user can use the library to create, train and use feed-forward networks without the graphical tool from within their own programs.

The design of the library is fully object-oriented with much emphasis put on high reuse and extension capabilities. In general, this objective was met by using the highest level of abstraction that was possible using the C++ language. An important goal was to fully separate the learning algorithm from the structure of the network. This was achieved by using the Decorator design pattern (Freeman et. al, 2004) to have a basic computing class, which is dynamically extended with the ability to learn with a particular learning algorithm. The standard error back-propagation algorithm (Rummelhart et al, 1986) is provided with the library, but others can easily be added.

There are several groups of potential users of the package, for example students or researchers who want to experiment with multi-layer feed-forward neural networks, but do not have the time to program it from a scratch. Other potential users are the ones who want to apply neural networks to solve a particular problem. They can make use of the graphical tool to train a network and integrate it into their own programs using the library.

Additionally, NTUNE was designed for learning and studying the error back-propagation learning process, as there is a special visualization feature: a function learned by a network with two inputs and one output can be considered as a gray-scale image. This image can be displayed during the training. Usually it shows emerging and moving regions, which finally form the target image.

## Features

NTUNE simulates multi-layer feed-forward neural networks and provides a standard error back-propagation supervised learning algorithm. In the learning phase, the learning rate and the momentum

constant can be adjusted. Also, the error and the correlation for both training and testing sets are plotted in real-time. Once a network is created and trained, it can be exported to be used in other C++ applications. The plot data can also be exported to be further processed. A special feature of NTUNE is a visualization tool for networks with two inputs and one output

### Main window

Figure 4 shows the main window after the program is started.

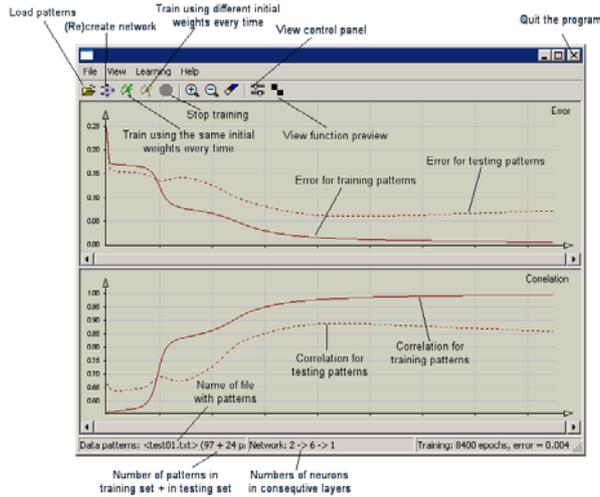


Figure 4 – NTUNE main windows.

There are four main menus, File, View, Learning, and Help. The File menu can be used to load pattern files, to create a new network, and to export the network data or the plot data.

The View menu can be used to zoom in and out, and to clear the plot windows. The update rate for the plot windows can be adjusted, and the Control Panel and the Function Preview Window can be enabled or disabled.

The Learning menu is used to start training the network, initialising it with either the same random values every time a training is started, or with different random values in different training sessions.

The training can also be stopped and a single pattern can be presented to the network.

Finally, the Help menu can be used to access the on-line documentation.

There is a menu bar and icons for easy access to the main functions of the program. The main window is then split in two plot sections. The upper one displays the network error over training cycles in real-time, whereas the bottom one displays the correlation of the network over training cycles in real-time.

### Creating and training a network

Before a network can be created, a pattern file needs to be loaded. After loading a pattern file, the user needs to specify the percentage of patterns that have to be

used as training set, while the remaining ones will form the test set (Figure 5).

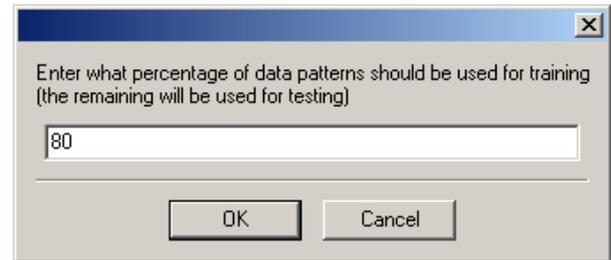


Figure 5 – Dividing the pattern data.

The pattern file defines the number of units in the input layer and in the output layer, but it is also necessary to specify the number of hidden layers and the number of units in each of the hidden layers using the dialog box shown in Figure 6:



Figure 6 – Entering the number of hidden layers and their numbers of units.

The next step is to select the Learning Rate and the Momentum for the Back-propagation algorithm (Figure 7). The order in which the patterns are used within one cycle can also be defined as fixed sequence, i.e. as they appear in the file (Batch mode), or in random order.

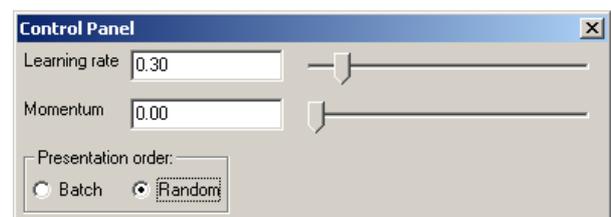


Figure 7 – Control Panel for learning algorithm.

The next figure shows a typical training run for a network (Figure 8). The errors over time for the training set and the test set are displayed on-line in the top graph, while the correlation over time for the training set and the test set are displayed in the bottom graph.

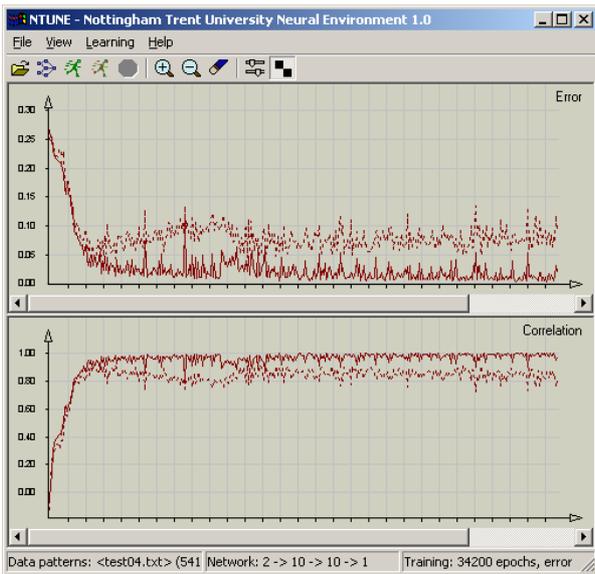


Figure 8 – Typical training run.

A novel feature of NTUNE is the Function Preview Window, which shows in real-time the development of the trained function for networks with a two-dimensional input space and one output neuron. This feature will be described in the next section.

### The Function Preview Window

In order to visualize the learning process and to demonstrate the concept of linearly separable functions (Picton, 2000) a Function Preview Window has been developed, which shows in real-time the development of the trained function for networks with a two-dimensional input space and one output neuron.

Once the pattern file has been loaded, the patterns of the training set or the test set are displayed in a two-dimensional grid, and the patterns are displayed either red or green, indicating the associated output values below 0.5 or above (Figure 9).

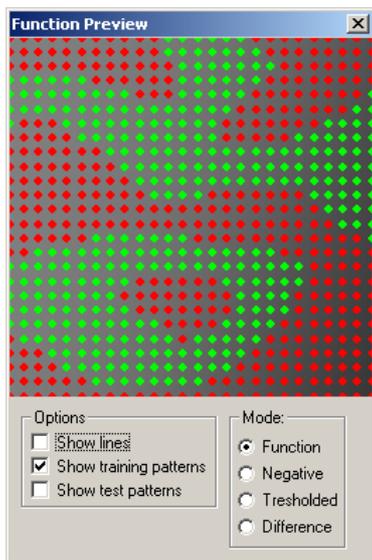


Figure 9 – Function Preview Window with patterns.

As the search progresses, the underlying function that the network has learned is displayed online (Figure 10) where a zero value is displayed as a white pixel and a one value is displayed as a black pixel. Intermediate values are displayed in shades of grey.

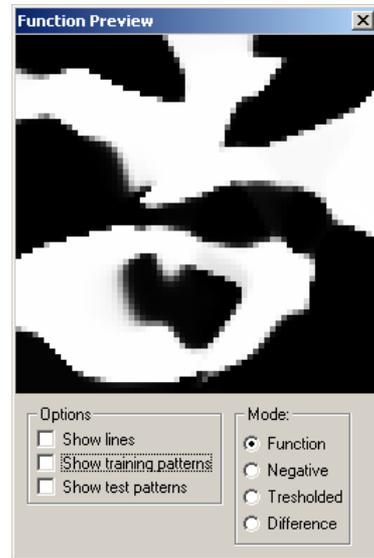


Figure 10 – Underlying function learned by the network.

It is also possible to display the lines corresponding to decision boundaries of neurons in the first layer on-line during training. Decision lines are determined by:

$$w_1 \cdot x + w_2 \cdot y + b = 0 \quad (3)$$

where  $w_1$  and  $w_2$  are the weights of a neuron and  $b$  is the bias of a neuron.

Figure 11a shows the decision lines before training takes place and Figure 11b shows the decision lines after training:

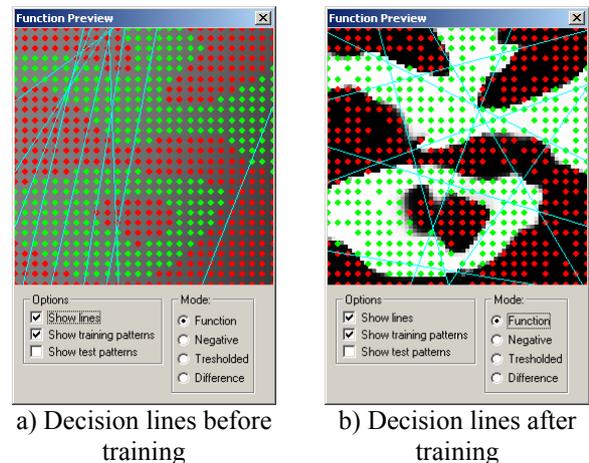


Figure 11 – Function Preview Window showing decision lines.

## Testing single patterns

Once a network is trained, it can be tested manually with new patterns using the option “test by hand” in the learning menu (Figure 12).



Figure 12 – Dialog box for testing a single pattern.

In this dialog box, a single pattern can be entered and the resulting response from the network will be displayed in a message box (Figure 13).

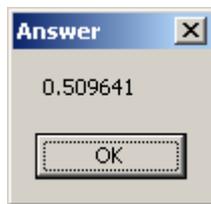


Figure 13 – Resulting answer from the network.

This feature can be used, for example, by students to validate a trained network.

## FILE FORMATS

The following section describes the file formats used with NTUNE, the format of the patterns files, the network files, and the plot data files.

### Format of patterns files

Patterns files are ASCII text files. At the beginning, there are three numbers: number of inputs for the network, number of outputs of the network, and the number of patterns in the file. Then follow the patterns. Each pattern is described as a sequence of real numbers for consecutive inputs and then the outputs. These numbers need to be from the range [0,1]. Figure 14 shows an example of a pattern file:

```
2 1
10
0.560000 1.000000 0.000000
0.800000 0.120000 0.000000
0.080000 1.000000 0.000000
0.960000 0.200000 1.000000
0.720000 0.000000 0.000000
0.200000 0.600000 1.000000
0.680000 0.400000 1.000000
0.440000 0.640000 1.000000
0.720000 0.920000 0.000000
0.120000 0.600000 1.000000
```

Figure 14 – Example of a NTUNE pattern file.

Note: Some problems may occur under Linux - the end-of-line character probably has to be the Unix-type

one and on some distributions the decimal separator character for floating-point numbers has to be comma instead of dot.

### Format of network files

Network files are also ASCII text files. At the beginning, there are three numbers: the number of inputs nodes, the number of outputs and the number of network layers. Then follow descriptions of each layer: each description starts with the number of neurons in that layer. Then follows a list of weights, real numbers corresponding to consecutive weights. Figure 15 shows an example of a network description file.

```
2 1
3
2
-0.264863300744 0.469694440597 0.386450929964
-0.109636118507 -0.804090330799 -
0.147520877600
2
-1.844448806314 0.075209802857 -1.228686078689
-1.647298861078 -1.276881329589 -
0.109270145153
1
0.076063794241 -0.707801538823 0.426283502664
```

Figure 15 – Example of a network description file.

### Format of Plot files

Finally, the data representing the network errors and the correlation for the datasets over training cycles can be exported and processed with other applications. Figure 16 shows an example of an NTUNE plot file.

```
----- Error plots -----
Plot #0:
0.252300
0.257100
0.251700
0.252200
0.250000
0.250300
Plot #1:
0.247300
0.247000
0.245400
0.245000
0.263000
0.255200
----- Correlation plots -----
Plot #0:
-0.018000
-0.021800
-0.009100
-0.004300
0.005700
0.006200
0.007600
Plot #1:
0.000000
0.000900
0.003600
0.001800
0.016700
0.017000
```

Figure 16 – Example of an NTUNE plot file.

A plot file starts with the data for the error plots, followed by the data for the correlation plots.

## CONCLUSIONS

NTUNE is an educational neural network tool, which can not only be used for teaching purposes but also for real applications. It is released under the GNU General Public Licence and hence is freely available. Installation files for Linux and Windows are available at <http://www.mczard.republika.pl/ntune.html>.

## AUTHOR BIOGRAPHY



Michal Czardybon studies Computer Science at The Silesian University of Technology and he is expected to finish his Master degree in 2005. He was visiting the Nottingham Trent University one semester as a Socrates/Erasmus student in 2004. His research interests include artificial intelligence and programming languages design.



Lars Nolle graduated from the University of Applied Science and Arts in Hanover in 1995 with a degree in Computer Science and Electronics. After receiving his PhD in Applied Computational Intelligence from The Open University, he worked as a System Engineer for EDS. He returned to The Open University as a Research Fellow in 2000. He joined The Nottingham Trent University as a Senior Lecturer in Computing in February 2002. His research interests include: applied computational intelligence, distributed systems, expert systems, optimisation and control of technical processes

Gerald Schaefer gained his PhD in Computer Vision from the University of East Anglia. He joined the School of Computing and Informatics at Nottingham Trent University as a Senior Lecturer in 2001. His research interests include colour image analysis, physics-based vision, image retrieval, medical imaging and artificial intelligence.

## REFERENCES

- Freeman, E., Freeman, E., Sierra, K., Bates, B. (2004) "Head First Design Patterns", O'Reilly
- Hornik, K., Stinchcombe, M., White, H. (1989) "Multilayer Feedforward Networks are Universal Approximators", *Neural Networks*, Vol. 2, pp 359-366
- Hornik, K. (1991) "Approximation Capabilities of Multilayer Feedforward Networks", *Neural Networks*, Vol. 4, pp 251-257
- McCulloch, W.S., Pitts, W. (1949) "A logical calculus of the ideas imminent in nervous activity", *Bulletin of Mathematical Biophysics*, Vol.5, pp115-133
- Picton, P. D. (2000) "Neural Networks", 2nd Edition, Palgrave
- Rumelhart, D.E., McClelland, J.L. (1986) "Parallel Distributed Processing", MIT Press, Cambridge, Massachusetts
- Rummelhart, D. E., Hinton, G. E., Williams, R. J. (1986) "Learning representations by back-propagating errors", *Nature*, vol. 323, pp 533-536
- Smart, J. (2003) "Presenting xwWindows: a native UI cross-platform application framework", FOSDEM 2003, Brussels
- Zell, A. (1994) "Simulation Neuronaler Netze", Addison-Wesley

# Introducing The Swingometer Crossover And Mutation Operators For Floating-Point Encoded Genetic Algorithms

Shane Lee and Hefin Rowlands

University of Wales, Newport  
School of Computing and Engineering  
Allt-yr-yn Campus  
PO Box 180  
Newport Gwent UK  
Telephone (01633) 432441  
Fax (01633) 432442  
s.lee@newport.ac.uk  
h.rowlands@newport.ac.uk

## Abstract

Genetic algorithms that utilise floating point-encoded chromosomes instead of the traditional binary or Gray code have become wide spread in recent years. This paper introduces new floating-point crossover and mutation operators for such genetic algorithms. The operators are derived from examining the implicit constraints that traditional binary crossover and mutation operators impose on the values of the parameters being affected. It is shown by an example that a genetic algorithm using these operators does converge and they should be considered for general use in floating-point genetic algorithms.

Keywords Genetic Algorithm, Floating-point encoded, Real encoded

## 1. Introduction.

Genetic algorithms are now a well-known set of search algorithms based on the ideas and theories of natural selection [Goldberg 1989; Holland 1975; Workman & Reader 2004]. Traditionally, a population of binary or Gray coded candidate solutions are created at the initialisation stage of a genetic algorithm, the population ‘evolves’ by utilising a collection of selection, recombination and mutation operators.

However, several possible weaknesses have been identified with genetic algorithms that use binary or Gray code [Wright 1991; Oyama A Et al 2000]. An alternative encoding method using real numbers, often floating point numbers has become popular in recent years. One such weakness of binary encoding is the resolution. Consider a binary bit string of defined length  $l$ . The string can represent  $2^l$  real numbers. The range of the real numbers that the string represent is bounded by the upper value  $R_U$  and lower value  $R_L$ , where  $R_U - R_L = 2^l$ . The resolution is restricted by the value of the least significant bit. Floating-point numbers are not subject to this restriction. Floating-point numbers in genetic

algorithms would also be free of the necessity of  $R_U - R_L$  being a ‘round’ or normalized base two number. If a problem to be tackled by a binary genetic algorithm is not neatly bounded by a ‘round’ base two number in each dimension then new operators may need to be introduced in order to disqualify chromosomes that have parameters outside the required range.

Wright [1991] analyses the effect crossover has on the real values of parameters represented by the bit strings in binary and Gray encoded genetic algorithms by viewing the change in the ‘perturbations’ in the bit strings. However, implementing this view of crossover is necessarily conscious of the binary/Gray bit strings that might represent real numbers. But real floating-point numbers have a potentially infinite number of decimal places and therefore defy conversion to a limited length binary bit string without making approximations that would reintroduce the resolution weakness. Wright [1991] proposes an alternative crossover operator for real numbers named ‘Linear Crossover’ that is quite unlike the conventional crossover operators used in binary/Gray coded genetic algorithms and doesn’t implement the ‘perturbations’ revealed in his analysis. Djurišić et al [1997] use a crossover operator that restrict themselves to swapping complete floating-point encoded genes, as each real encoded gene is equivalent to many binary encoded genes the mixing in recombination does not produce new values for any parameter.

## 2 The Implicit Constraints of Binary Encoded Crossover.

### 2.1 Traditional Crossover

To illustrate the implicit constraints that traditional crossover imposes, it is wise to consider an example. Figure 1 shows an instance of single point crossover on a pair of binary encoded chromosomes. In the example, the chromosomes have two dimensions; each

dimension is in the range 0 to 63. Parent chromosome 1 represents the vector (25,35) while parent 2 represents the vector (42,56). The single crossover point falls between the 9<sup>th</sup> and 10<sup>th</sup> bits, the trailing bits are swapped producing child chromosomes 1 & 2 that represent the vectors (25,32) and (42,56) respectively.

considered as this will result in swapping unmodified parameters. When the crossover occurs, bits are swapped between bit strings. Equivalent positioned bits in each parameter are swapped. If equivalent bits swapped agree (i.e. both bits have 1 or both have a 0) then neither parameter is modified. If equivalent bits swapped

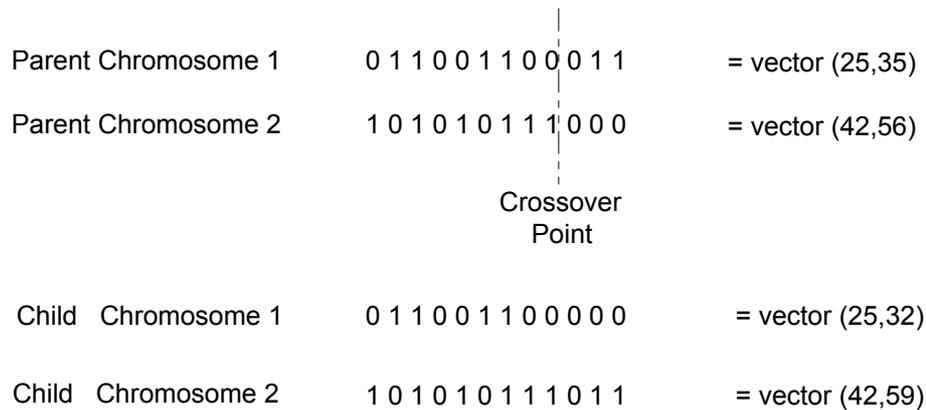


Figure 1, An illustration of traditional single point crossover

## 2.2 Equality of Modification, The First Constraint.

Before crossover occurs two parent chromosomes are chosen. In single point crossover, the crossover point will generally fall within the bit string of one dimension of the vector represented by the chromosomes. In figure 1 this occurs in the parameter of the second dimension. This will generally modify the value of each parameter to produce novel values for this parameter in the child chromosomes. The case where the point of crossover falls between parameters need not be

disagree (i.e. one bit is 1 and the other is a 0) one parameter has the value of that bit subtracted and the other has that value added. Consider the case in figure 2, here only the parameters where the crossover occurs are viewed. In this example the crossover point falls between the 5<sup>th</sup> and 6<sup>th</sup> bits. After crossover parameter  $A_i$  has been modified by  $-1$  while parameter  $B_i$  has been modified by  $+1$ . Another way of stating this is that the sum of the parent parameters before crossover must be equal to the sum of the child parameters post crossover.

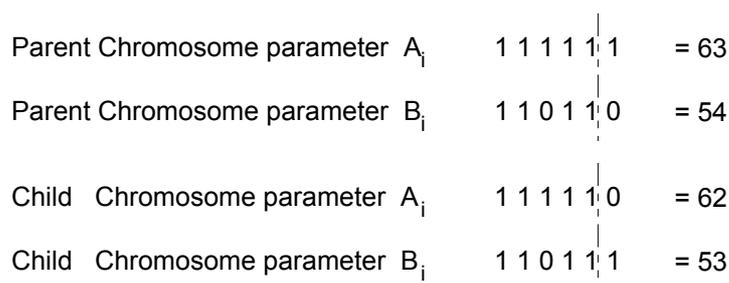


Figure 2, Illustration of the difference between parameters constraint.

To give these assertions algebraic form consider two parent chromosomes 'A' and 'B' each with 'n' parameters represented by floating-point values  $(A_1, A_2, \dots, A_i, \dots, A_n)$  and  $(B_1, B_2, \dots, B_i, \dots, B_n)$ . The crossover point falls within parameter  $i$ . The child chromosomes are  $(A_1, A_2, \dots, A_i - \Delta, \dots, B_n)$  and  $(B_1, B_2, \dots, B_i + \Delta, \dots, A_n)$ . Where ' $\Delta$ ' is the absolute value that modifies each affected parameter. Therefore the first constraint can be expressed as equation 1 where the left hand side of the equation is the affected parameters prior to crossover and the right hand side represents the new parameters post crossover.

$$(A_i + B_i) = (A_i - \Delta + B_i + \Delta) \dots\dots\dots 1$$

**2.3 Difference between parameters, The Second Constraint.**

Parameters with similar bit strings modify each other far less than those with dissimilar ones. The extreme case of identical bit strings cannot modify each other at all. In fact the magnitude of the modification is limited to the difference between them. Figure 3, the parameters differ by only one bit, no matter where the crossover point is, parameter  $A_i$  can change by no more than the value of the bit that differs from  $B_i$ .

Parent Chromosome parameter $A_i$	1 1 1 1 1   1	= 63
Parent Chromosome parameter $B_i$	1 1 0 1 1   0	= 54
Child Chromosome parameter $A_i$	1 1 1 1 1   0	= 62
Child Chromosome parameter $B_i$	1 1 0 1 1   1	= 53

Figure 2, Illustration of the difference between parameters constraint.

In general algebraic form the second constraint is

$$\Delta \leq |A_i - B_i| \dots\dots\dots 2$$

**2.4 Boundary Restrictions, The Third Constraint.**

The value of a parameter after crossover can be no more than the value of the upper boundary ' $R_U$ ' no less than their lower boundary ' $R_L$ '

allowed for that dimension. With brief inspection of figure 3, where the shown parameters have an upper boundary of 63 and lower boundary of 0, it can be seen that ' $\Delta$ ' cannot be greater than 63 or less than 0. In algebraic form these constraints for ' $\Delta$ ' using parameters  $A_i$  and  $B_i$  can be expressed as:-

$$\Delta \leq R_U - A_i \dots\dots\dots 3a$$

$$\Delta \leq A_i - R_L \dots\dots\dots 3b$$

$$\Delta \leq R_U - B_i \dots\dots\dots 3c$$

$$\Delta \leq B_i - R_L \dots\dots\dots 3d$$

**3 The Swingometer Creating the Operator.**

**3.1 Creating the Operator.**

The first step for the new floating-point crossover operator is to calculate the maximum value that each parameter can have subtracted without violating the constraints described in sections 2.3 and 2.4. These values,  $\text{Max } \Delta_A$  and  $\text{Max } \Delta_B$ , can be derived by choosing the minimum value allowed as expressed by equations 4a and 4b.

$$\text{Max } \Delta_A = \min [|A_i - R_L|, |A_i - B_i|, |R_U - B_i|] \dots\dots 4a$$

$$\text{Max } \Delta_B = \min [|B_i - R_L|, |A_i - B_i|, |R_U - A_i|] \dots\dots 4b$$

Parameter  $A_i$  can have no more than  $\Delta_A$  subtracted from it without violating the second and third constraints. Of course the argument is equally true for  $B_i$ . It follows that the maximum value allowed by the first constraint ' $\text{Max } \Delta$ ' is given by equation 5.

$$\text{Max } \Delta = \min [\text{Max } \Delta_A, \text{Max } \Delta_B] \dots\dots\dots 5$$

The value ‘ $\Delta$ ’ for the crossover is calculated by generating a random number between 0 and ‘Max  $\Delta$ ’. The final step to implement the operator is to randomly choose to subtract ‘ $\Delta$ ’ from one parameter and add it to the other. This automatically imposes the first constraint (section 2.2).

**3.2 Visualising the Operator.**

The floating-point operator can be visualised as a ‘swingometer’ (figure 4). The arc of the swingometer is delimited by the values +Max  $\Delta$  and -Max  $\Delta$ , represented in figure 4 by points ‘a’ and ‘b’. The length of the arc is 2Max  $\Delta$ . A value ‘ $\Delta$ ’ is selected randomly along the length of the arc. If the value ‘ $\Delta$ ’ is between ‘a’ and ‘ $\Delta = 0$ ’ it is added to  $A_i$  while simultaneously being subtracted  $B_i$ ; but if ‘ $\Delta$ ’ is between ‘ $\Delta = 0$ ’ and ‘b’ it is added to  $B_i$  while simultaneously being subtracted  $A_i$ .

**4. The Swingometer Mutation Operator For Real Encoded Genetic Algorithms**

The swingometer mutation operator has a very similar derivation to the equivalent swingometer crossover operator. However, the only constraint that applies for mutation is similar to the boundary restriction described in section 2.4. A parameter ‘ $A_i$ ’ selected for mutation after the operation cannot be greater than ‘ $R_U$ ’ or less than ‘ $R_L$ ’. For mutation consider ‘ $\Delta$ ’ to be a number added to ‘ $A_i$ ’ or subtracted from ‘ $A_i$ ’. The maximum value that can be added to ‘ $A_i$ ’ is given by equation 6a and the maximum value that can be subtracted from ‘ $A_i$ ’ is given by equation 6b.

$$\begin{aligned} \text{Max } +\Delta &\leq R_U - A_i \dots\dots\dots 6a \\ \text{Max } -\Delta &\leq A_i - R_L \dots\dots\dots 6b \end{aligned}$$

For the swingometer, in figure 4, the limits ‘a’ and ‘b’ are Max + $\Delta$  and Max - $\Delta$  respectively. As for crossover, a random number is generated in along the arc ‘a’ to ‘b’ and the parameter  $A_i$  is modified accordingly.

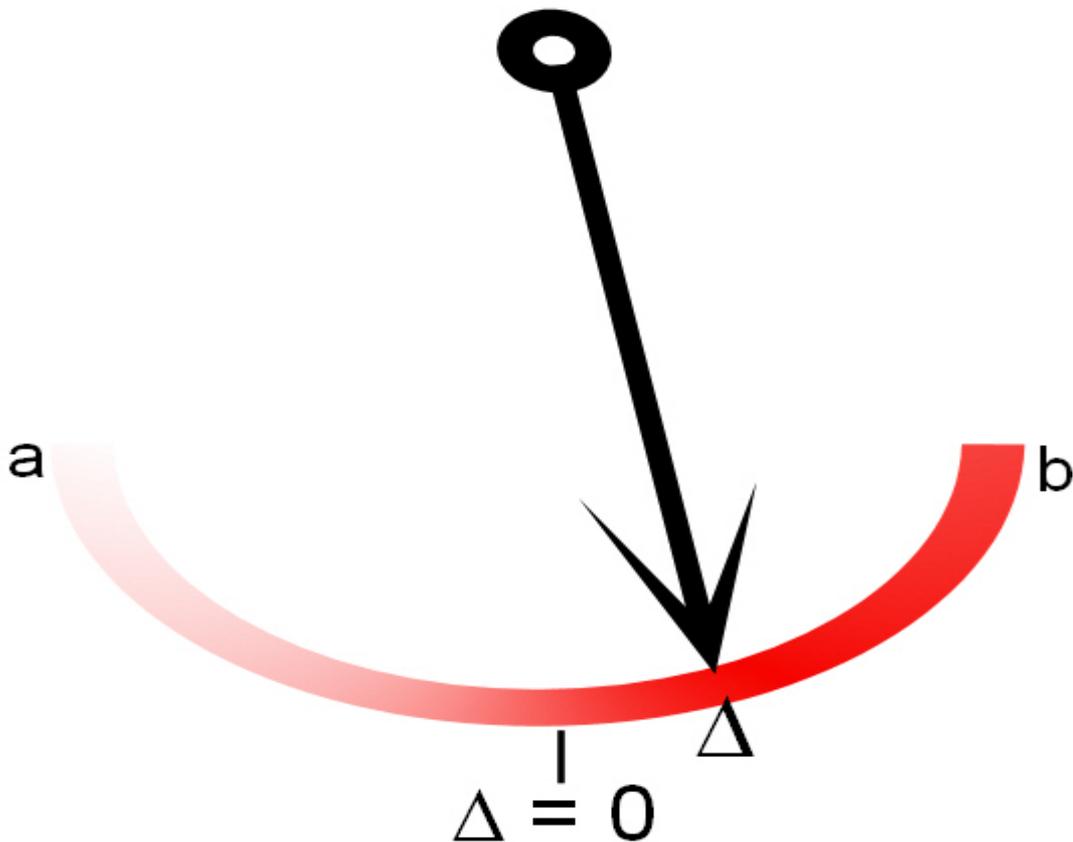


Figure 4, Visual representation of the swingometer crossover and mutation operators.  
 For crossover a = Max + $\Delta$  and b = Max - $\Delta$ .  
 For mutation a = +Max  $\Delta$  and b = -Max  $\Delta$ .

### 5. A Working Example.

This section shows the results of a set of 60 identical experiments that demonstrate the new operators in use within a genetic algorithm. The genetic algorithm used a population of 25, a crossover rate of 0.9 and a mutation rate of 0.2. The function that was optimised (equation 7) is the 5<sup>th</sup> of the Dejong suite[1975] normalised to give a global optimum fitness of 10,000. For each generation the fitness of the fittest chromosome is recorded and averaged for the 60 identical experiments.

$$F_5 = \left[ 0.002 + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^2 (x_i - a_{ij})^6} \right]^{-1} \dots 7$$

Figure 5 shows the result; the genetic algorithm does perform as expected. The fitness of the fittest in the population rises generation by generation until a plateau of convergence is reached.

### 6 Discussion And Further Work.

This paper has set out to describe two new operators for floating-point encoded genetic algorithms and has shown that they do allow the algorithm to converge. The two operators have

been extensively and successfully used in [2002]. However, no attempt has been made to measure its relative efficacy when compared to other floating-point operators; experiments to fill this gap are a priority. Even though the new operators have yet to be fully tested they are simple to implement and do work and should be considered when designing a floating-point encoded genetic algorithm.

### 7. References.

- K. Dejong , 1975, An Analysis of the Behaviour of a Class of Genetic Adaptive Systems, Doctoral Thesis, University of Michigan.
- Djurišić A Et Al, 1997, Genetic Algorithms For Continuous Optimization Problems – A Concept Of Parameter-Space Adjustment, Journal of Physics A-mathematical & General. Pp 7849-7861.
- D. E. Goldberg, 1989, Genetic Algorithms in Search, Optimization & Machine Learning, Addison & Wesley.
- J. H. Holland, 1975, Adaptation In Natural And Artificial Systems, The University of Michigan Press.
- S. Lee, 2002, Genetic Algorithms, Orthogonal Arrays And Diploid Chromosomes, Doctoral Thesis, University of Wales

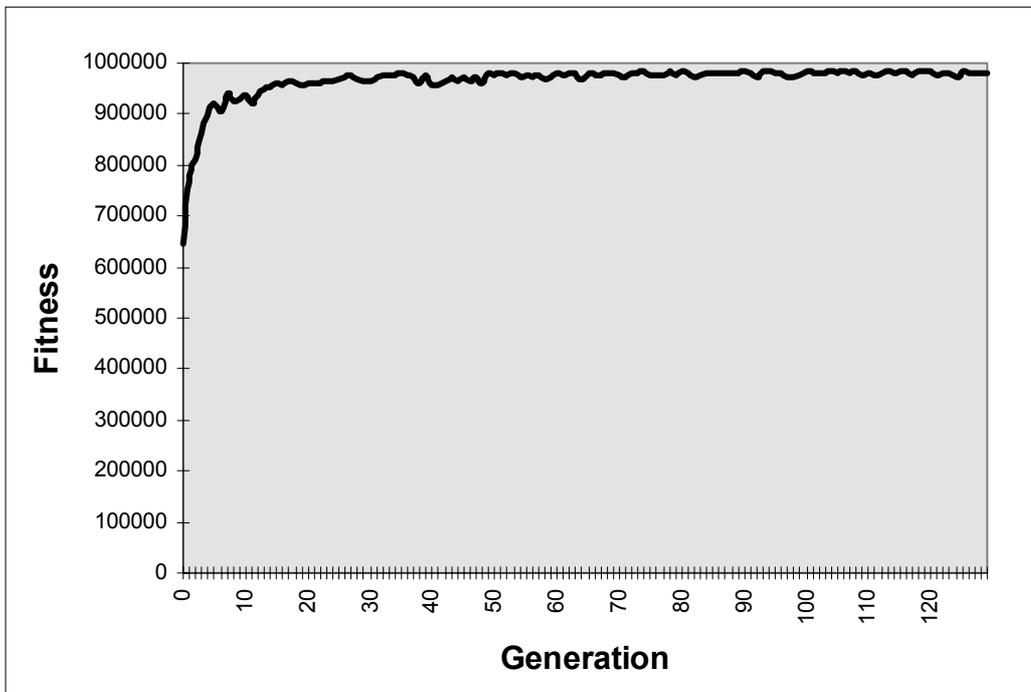


Figure 5, Result of the experiments to demonstrate that the new operators can work within a genetic algorithm.

Oyama A Et al, 2000, Real\_Coded Adaptive Range Genetic Algorithm and Its Application to Aerodynamic Design and its Application, JSME International Journal Series A, Vol 43, Part 2, pp 125-129.

L.Workmen & W. Reader, 2004, Evolutionary Psychology: An Introduction, Cambridge University Press.

A. Wright , 1991, Genetic Algorithms for Real Parameter Optimization, Foundations of Genetic Algorithms , pp205-218.

## 8. Biographies



### **Dr Shane Lee**

Senior Lecturer, School of Computing and Engineering –UW,N

His research field is computational intelligence in particular evolutionary computation and covers areas related to traditional engineering applications, computer games and art. He is the programme leader and creator of the BSc (hons) Games Development and Artificial Intelligence.

Outside academia he is a promoter and performer of stand-up comedy.



### **Professor Hefin Rowlands**

Head of Research & Enterprise, School of Computing and Engineering –UW,N

Professor Hefin Rowlands is responsible in promoting, developing and leading research and enterprise activities in the University College's School of Computing and Engineering.

His research field, in the area of Quality Systems, covers research into Quality Management and Strategies including Six-sigma and Business Excellence Models.

Research interests also cover process modelling and developing an integrated model for business systems.

He has been involved in many successful TCS and College & Business Partnership (CBP) projects which enables local businesses to benefit from the expertise of a recently qualified graduate and the knowledge base of the School of Computing and Engineering at the University College.

# FAST SIMULATION AND OPTIMIZATION WITH NEURAL NETWORKS

Friedrich Biegler-König  
Fachbereich Mathematik und Technik  
Fachhochschule Bielefeld  
33604 Bielefeld, Germany  
E-mail: friedrich.biegler-koenig@fh-bielefeld.de

Peter Deeskow  
STEAG KETEK IT GmbH  
Centroallee 261  
46047 Oberhausen, Germany  
E-mail: peter.deeskow@steag-ketek.de

## KEYWORDS

Neural Networks, Real-Time Simulation, Power Plant Optimisation, Simulated Annealing

## ABSTRACT

Neural Networks can be powerful tools for the approximation of complex nonlinear behaviour. After a learning phase neural networks are able to predict results with high accuracy. While the learning phase might be time consuming, prediction is very fast. After substituting complex simulation programmes by Neural Networks, these can be used in real-time operation, e.g. for optimisation processes. We demonstrate this approach by modelling the operation of a multi-block power plant. Afterwards operation can be optimised using any mixed integer optimisation algorithm, in our case Simulated Annealing.

## INTRODUCTION

Essentially no complex technical device is built today without previous software-simulation. In many cases even ab initio models which are directly based on the law of physics can be employed.

However, these models which are used in the design process of industrial plants are often less usable in everyday business.

In everyday business the configuration of a plant is rarely changed but the personnel has to react quickly to changes of external parameters in order to get an optimal production output. Simulation with models which were used when designing the plant are usually too slow to meet the requirements of real time production.

In many cases simplified models are constructed to allow real time operation. Linear, piecewise linear, or polynomial fits to complex functions are often employed as well as characteristic curves for technical components.

These approaches work satisfactorily if there are only very few influencing variables (one or two) and if the

behaviour of the fitted functions is similar to the used fits (linear, quadratic, ...).

In this paper we present neural networks as a universal method to replace complex nonlinear models of several variables. The type of neural network we are describing can be used for any continuous differentiable function. They produce a very high accuracy of prediction and are not bound to a special behaviour of the original model.

In the next chapter we will give a short introduction into neural networks. Then we will apply this technique to a complex technical plant, and illustrate the possibilities opened by the neural network approximation.

## NEURAL NETWORKS

Neurons in Computer Science are simple computational units. They receive inputs from outside or from other neurons, calculate a response, and send it on to other neurons or the outside. The connections between Neurons are also called synapses (with reference to the biological model). The strength of a connection is determined by its "synaptic weight".

In Computer Science a great number of possible neurons and connection types have been examined. For practical applications the so called feed-forward model is the one most widely used. Here the neurons are arranged in layers. Each neuron gets inputs from all neurons of the previous layer or (if it is the first one) from the outside and sends its output to all neurons of the following layer or (if it is the last one) the outside. Hence these networks always have an input and an output layer and a number of "hidden layers".

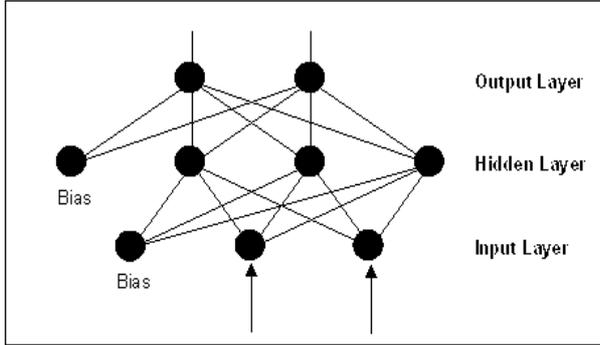
The neurons of this network type usually take the weighted sum of their inputs and apply an "activation function" to the result. Let  $i_1, \dots, i_n$  be the input values of a neuron, and  $w_1, \dots, w_n$  be the synaptic weights on the incoming connections. Then the output value of the neuron is calculated as:

$$a = \sigma(\text{net}) = \sigma\left(\sum_{k=1}^n w_k \cdot i_k\right) \quad (1)$$

with a so-called Sigmoid function

$$\sigma(x) = \frac{1}{1 + e^{-x}}. \quad (2)$$

In most cases only one hidden layer is used. Hecht-Nielsen proved the theoretical result that any analytical function can be approximated by a network with only three layers (Hecht-Nielsen 1987) (see figure 1).



Figures 1: Neural Network with 3 Layers

### Learning

A neural network learns, if its synaptic weights are determined in a way that for given input values also given output values are generated. This is called “supervised learning“.

The synaptic weights are determined using a “learning algorithm“. The set of given input- and output-values is called “learning set“.

The best known learning algorithm is called Backpropagation (Rumelhart et. al. 1988). It minimizes the so-called energy function using a gradient descent scheme:

$$E = \frac{1}{2} \sum_{k=1}^N \sum_{j=1}^m (o_j^k - t_j^k)^2, \quad (3)$$

where N is the number of examples, m the number of neurons in the output layer, and  $o_j^k$  the output of the network in neuron j for example k.  $t_j^k$  is the wanted output value.

For neural networks with only one hidden layer there are other, sometimes more efficient learning algorithms (Bärmann and Biegler-König, 1992).

### Prediction

It is the purpose of the learning process is to enable the Neural Network to predict output values for such examples which are not in the learning set.

In order to verify the quality of the learning process one keeps another set of examples, called the validation set.

This set is disjoint from the learning set, but also contains the correct output values which are compared to the predictions of the network.

### Coding

Output values of Neural Networks with Sigmoid functions are always contained in the interval (0, 1). In order to get a normalization, this is also desirable for the input values.

Before learning a learning set, its examples have to be transformed into values between 0 and 1. This coding has to be invertible to transform the net predictions into uncoded form.

In many cases a linear coding of the example sets is sufficient, but sometimes prediction quality can be enhanced by choosing a suitable more complex coding.

### Measuring Errors

To quantify the prediction capacity of a network, we have to define a learning error (for the learning set) and a prediction error (for the validation set). Usually the Mean Squared Error is used for this purpose:

$$MSE = \frac{1}{N} \sum_{k=1}^N \sum_{j=1}^m (o_j^k - t_j^k)^2 \quad (4)$$

Another error measurement is provided by the empirical correlation coefficient:

$$EKK = \frac{\sum_{k,j} (o_j^k - o_j) \cdot (t_j^k - t_j)}{\sqrt{\sum_{k,j} (o_j^k - o_j)^2 \cdot (t_j^k - t_j)^2}}, \quad (5)$$

where  $o_j$  denotes the mean value of the outputs in neuron j and  $t_j$  the mean value of the corresponding exact values. If all examples in an example set are predicted correctly, the MSE will vanish and the EKK will be 1.

### FAST SIMULATION OF POWER PLANTS

An example of the need of fast simulation can be found in the optimisation of power plants. We are considering a power plant with four blocks which are identical in construction (a simple case). Each block can generate power in the range of 108 MW to 360 MW. A block cannot produce less than 108 MW. If a block is switched off, there are considerable costs to start it up again.

A detailed model of a power plant block has been built using Epsilon, a programme developed by the company Sofbid (<http://www.sofbid.com/epsilon/>, see Brinkmann and Pawellek 2003 and 2004, and Brinkmann 2003) in Zwingenberg, Germany. Epsilon is a simulator specialized in power generating facilities. Epsilon

models can be very complex (see Figure 2 for a section of a model of a power station block). After the model has been constructed, Ebsilon needs about 5 to 20 seconds on a PC to simulate a given situation for one block. This is by far not fast enough e.g. for a complex real-time optimisation which usually needs many thousand evaluations.

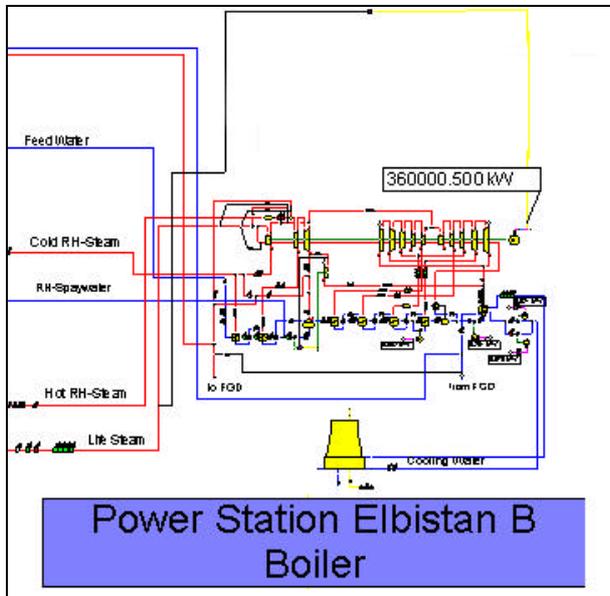


Figure 2: Section of Ebsilon Power Station Model

A specific situation of the power plant block is determined by only a few input parameter. The most important parameters is the amount of energy which the block is to produce per hour.

The other parameters describe the external conditions and settings by the operational staff:

- Temperature of cooling water (between 0 and 30 °C).
- Air ratio in combustion chamber (between 1.1 and 1.4).
- Live steam temperature (between 510 and 540 °C)
- Hot reheat temperature (between 510 and 540 °C)
- Flue gas recirculation (between 0 and 50 kg/s).
- 8 more parameters specifying the degree of heat surface fouling.

The main result Ebsilon is supplying is the amount of coal needed. This in turn determines the main part of the total costs.

In a first experiment 15000 random input data sets were produced, each set containing 14 randomly determined values for the input parameter. A data set describes a specific situation of a power plant block. Employing Ebsilon, for each data set the required amount of coal is calculated. This may take a long time.

We now have a set of 15000 data sets with 14 input and one output value. We divided this set into a learning set (10000 data sets) and a validation set (5000 data sets).

After coding the examples (standard linear coding) and deciding on a number of hidden layer neurons (60), we can try to teach the learning set to a neural network. The learning algorithm converges quickly and yields a MSE of 0.00009 and an EKK of 0.99901 after a few hundred iterations.

After decoding the examples, our network is able to predict the required amount of coal for a given situation with an average error of 0.6%. This error has the same order of magnitude as the simulation error of an Ebsilon calculation. The following is a typical example from the validation set.

#### Input Values:

Power requirement:	317801 kW
Temperature of cooling water:	5.583 °C
Air ratio:	1.347
Live steam temperature:	522.517 °C
Hot reheat temperature:	513.577 °C
Flue gas recirculation:	38.055 kg/s
Heat surface fouling parameters:	1.390, 1.09, 1.332, 1.217, 1.108, 0.801, 0.909, 1.269

#### Output Values:

Amount of coal needed (Ebsilon calculation):	633833 kg/h
Amount of coal needed (Network prediction):	636844 kg/h
Absolute error:	3011
Relative error:	0.475%

The main advantage of our neural network is the response time. It is more than 5000 times faster than Ebsilon (response time: less than 0.001 seconds).

### A SIMPLE CASE OF POWER PLANT OPTIMISATION

We will now consider a first and comparatively simple case of optimisation:

Usually a request for a certain amount of power is communicated to the power plant. In order to minimize the costs, we wish to find the least expensive configuration of blocks meeting this requirement.

This is a mixed integer optimisation problem. It contains an integer part and a continuous part. The integer part determines which blocks are switched off, and which are running. The continuous part determines the distribution of the requirement among the running power station blocks.

This type of problem has been solved by fitting a linear curve to the coal-power function and employing techniques from linear programming (Schultz 2003). This is no longer possible if we take all 14 input values into account.

We used a simulated annealing method (Gerdes et al. 2004; Nolle et al. 2001) to compute the global optimum

of our problem. This heuristic optimisation method turned out to be very reliable, but always needed between 1000 and 3000 evaluations to find the global optimum (clearly not feasible directly with Epsilon in a real-time environment).

We also used a standard optimisation algorithm, an SQP scheme ("Successive Quadratic Programming", see e.g. Bazaraa 1993). These algorithms have to be applied to each possible block-configuration, since they only optimise the continuous part of the problem. Furthermore, there are several minima for each block-configuration. In order to find the global optimum, we have to apply the SQP-scheme with many sets of starting values.

In the end, the SQP-scheme needs about 3000 evaluations to find the same optimum as the simulated annealing process. In a more complex situation, simulated annealing would clearly be superior.

## SUMMARY AND FUTURE PROSPECTS

Approximating the behaviour of complex plants with neural networks turned out to be simple and easy-to-use method. These neural networks can replace elaborate simulation programmes in cases where several thousand evaluations are needed and real time results are required. This is, for instance, true in power plant optimisation for which a simple case has been described here.

In future we plan to apply optimisation algorithms to time series of power plant operations with a 24 hour advance planning. In these models, among others, start up costs and standstill periods have to be incorporated.

## REFERENCES

- Hecht-Nielsen, R. 1987. "Kornolgorov's Mapping Neural Network Existence Theorem". *IEEE 1987*. San Diego, Vol. II.
- Rumelhart, D. et al. 1988. *Parallel distributed processing*. MIT Press.
- Bärnann, F. and Biegler-König, F. 1992. "On a Class of Efficient Learning Algorithms for Neural Networks". *Neural Networks* 5, 1992.
- Brinkmann, K. and Pawellek, R. 2003. "Epsilon - Examples for the easier design and better operation of power plants". Published at the conference "Energy Forum 2003" at Sv. Konstantin Varna.
- Brinkmann, K. and Pawellek, R. 2004. "Optimierte Prozessführung von Kraftwerksblöcken mit Online-Werkzeugen: Betriebserfahrungen". VDI Tagung "Wissensbasiertes Betriebsmanagement senkt Kosten", Frimmersdorf.
- Brinkmann, K. 2003. "Epsilon-EPOS - Beispiele für eine einfachere Auslegung und einen besseren Betrieb von Kraftwerken", 3. VDI Symposium Sao Paulo.
- Schultz. 2003. "Integer Programming Applied to Power Systems' Generation Operating Planning", Course Material.
- Gerdes, I.; Klawonn, F.; Kruse, R. 2004. *Evolutionäre Algorithmen*. Vieweg Verlag.
- Nolle, L.; Goodyear, A.; Hopgood, A.; Picton, P.; Braithwaite, N. 2001. "On Step Width Adaptation in Simulated Annealing for Continuous Parameter Optimisation", *Fuzzy Days 2001*, 589-598.
- Bazaraa, M. S. 1993, *Nonlinear Programming*, Wiley.

# **Simulation of Complex Systems**



# MODELING OF BROADCASTED TSUNAMI ALERTS A PROPOSAL

Pauli Lallo  
Biomedical Engineering Centre  
Tallin Technical University  
Ehitajate tee 5 EE 19086 Tallinn Estonia  
E-mail: pauli.lallo@kolumbus.fi

## KEYWORDS

Multi-tone waveform, Alert broadcasting, Interface, Secured transmission

## ABSTRACT

This paper presents a description and proposal for tsunami alert system using a software modem in detection and public broadcasting for delivering critical information. The standard modem technology for data transmission and detection is based on hardware structures. Already during the World War II radio waves for used for triggering mines using coded tones. Present software modem technology gives low cost alternatives for hardware filtering and decoding of tones. This report briefly describes modelling and some simulation results of transmissions of multi-tone data alerts over broadcast systems. The security aspects are of special interest and simulation results show the level of security against interference, jamming and channel impairments. The importance of alerting systems has been dramatically demonstrated in tsunami. One concern is the costs, which can be kept on a low level using software-based solutions and existing present broadcasting infrastructures on the areas.

## 1 INTRODUCTION

Within minutes following an alarm signalling the strong earthquake on Dec. 26, 2004 the NOAA Pasific Tsunami Warning Center in Hawaii issued an information bulletin to nations in the Pacific at 8:14 p.m. EST Saturday, indicating that a magnitude 8.0 earthquake (later upgraded to magnitude 9.0) had occurred off the west coast of Northern Sumatra, Indonesia (Noaa 2004). "Tsunami" is the Japanese term meaning wave in the harbor. The wave arrived about two hour later in the "harbors". The alarm system, if there was any working on the areas concerned, had enough time for saving lives, however...

BANGKOK: Thailand will press ahead with its own tsunami warning system, Prime Minister Thaksin Shinawatra said on Monday, Jan. 31, 2005 in Asia Pasific News. "We don't care whether other countries cooperate or not. We will go ahead and we are ready to invest, even if we are the only country involved," he told reporters. The plan aims to avoid a repeat of the December 26 tragedy, when more than 280,000 people died after an earthquake off Indonesia sent tsunamis crashing into 11 nations.

Communication networks and broadcasting have been used for delivering critical information to people. Data transmission using standard ITU-T (CCITT 1989) modems have been made for use on wired telephone lines, thus they are not a good solution for mobile

wireless alert systems. There are on-going activities, radio amateurs and military forces, to develop new waveforms for wireless use. Voice grade or band-limited data transmission systems have been developed for mobile phones but the trend is to wide-band applications.

We have studied adaptive data transmission with an adaptive data modem (Lallo 1999) and are now proposing a new approach for the secure alert transmission systems. The following features of the modem are adaptively selectable: carrier frequencies (for example a three tone), symbol rate, the number of bits in one symbol i.e. the modulation method, and bit rate. The selectivity of the modulation method makes this approach adaptive to the channel in different cases.

The evolution in software modem technology based on the microelectronics expansion makes new things, such as waveform processing and soft waveform detection, reality in present and future modems (software modems). Hardware detection of the complex waveforms in practise is even impossible. In an alert system alone it is not necessary to use very complex waveforms. However, if the security of the data transmission is needed, the complex algorithmic approach in the detection and generation of waveforms is the right answer. We have simulated these waveforms for use in band-limited channels. The system may be called a band-limited OFDM approach.

## 2 DETECTION of MULTI-TONE WAVEFORMS

In figure 1 we have an example of the multi-tone hardware structure used during the II WW for military operations in Vyborg by the former Soviet Union forces in 1941. In the figure we see the receiver structure made of electron tubes, hardware filters and relays. The relays triggered huge amounts of explosives at strategic places (including a telecommunications exchange building).



Figure 1: Multi-Tone Receiver  
Signals Museum, Riihimäki, Finland

We have now designed an adaptive prototype modem for use in piece time operations and in this case as a proposal for an early warning system, for example a broadcasted tsunami alert. The prototype of figure 2 presents a radio interface circuit needed with the algorithmic generation and detection of waveforms with a PC. The solution on the board includes only a few electronic components: interfaces to the PC on the left, radio interface on the right and a controller circuit in the middle.

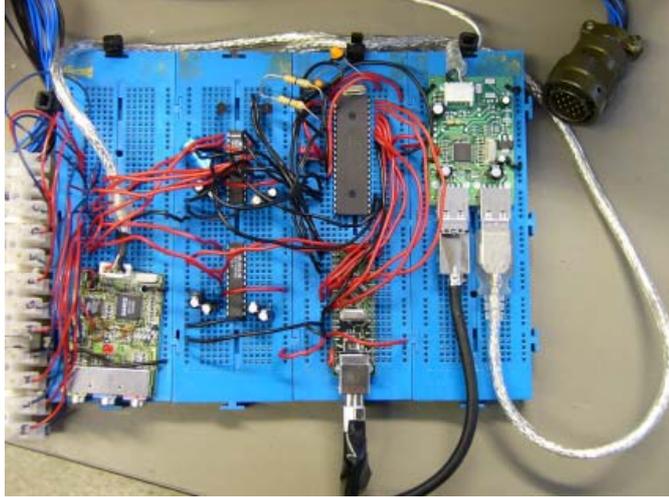


Figure 2: Interface Prototype

The prototype has just been designed and tested by Mr. K. Heinäaro as a part of his MSc Thesis.

The adaptive modem technology uses microcomputers (dsp) and has presently such a high calculation speed that we can use DFT, formula (1) as

$$S_x[m\Delta(f)] = \sum_{n=1}^N x[n\Delta(t)]e^{-2j\pi m\Delta(f)n\Delta(t)} \quad (1)$$

in the detection of multi-tone waveforms, for example a three-tone, generated as formula (2) presents

$$S(m_1, m_2, m_3) = \sum_{n=1}^N x[n\Delta(t)]e^{-2j\pi m_1\Delta(f)n\Delta(t)} + \sum_{n=1}^N x[n\Delta(t)]e^{-2j\pi m_2\Delta(f)n\Delta(t)} + \sum_{n=1}^N x[n\Delta(t)]e^{-2j\pi m_3\Delta(f)n\Delta(t)} \quad (2)$$

Where  $m_1$ ,  $m_2$ , and  $m_3$  represent three different carriers,  $N$  number of samples used in the symbol,  $\Delta(t)$  one sample time, and  $\Delta(f)$  the frequency selectivity. The selectivity of a  $N$ -point DFT is  $f_s/N$ , where  $f_s$  is the sampling frequency.

Table 1: DFT versus FFT

Samples N	Multiplications DFT	Multiplications FFT
13	169	-
16	256	32
26	676	-
32	1024	80
64	4096	192

Detection is based on the DFT Formula (1). It gives us the symbol detection from  $N$  samples, any of  $m_i$ ,  $i=1,2,3$  frequencies with resolution  $\Delta(f)$ , and symbol time  $N\Delta(t)$ . In the formula an example setting of the number of samples is  $N=26$ . We can set both  $\Delta(f)$ , and  $N\Delta(t)$  by adaptive modem software. FFT uses  $N$ -values only as powers of two, Table 1. Thus it is not fully adaptive as a DFT-solution.

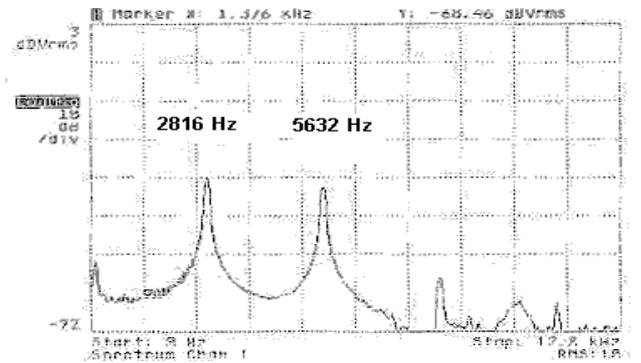


Figure 3: Spectrum of 22.5 kbps 16-QAM waveform

DFT with small number of samples are used in order to keep the solution as simple as possible but as selective as needed. In our simulations a 26-point DFT can well be used for most cases.

### 3 TRANSMISSION of ALERTS

A model for transmission of alerts is the adaptive data transmission demonstrated for the Finnish Defence Forces in an exercise in 2000. The narrow-band field tests have been made in VHF and UHF range, figure 3. The bit rate 22.5 kbps was reported as the result.

Principle of information transmission from sensors directly to people

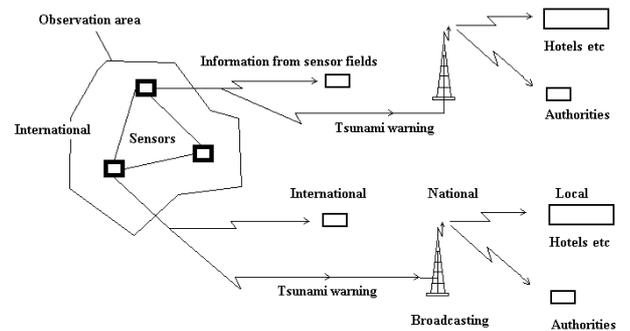


Figure 4: Tsunami warning directly to people

Figure 4 presents the proposed data transmission model for tsunami (or in general any sensor based information source) warning system, which send alerts directly to people. This tsunami warning is a three-tone waveform presented in formula 2. It is proposed to be included in the public broadcasting as a short 3-tone, which triggers the local authorities to action and also the hotels having the 3-tone receivers. A 3-tone receiver is possible to implement in any hand-held FM receiver (GSM or cellular phone). The use of this kind of alerts is so seldom that it does not spoil the broadcasting services. It needs some legislation in order to take the system in use and teaching population of its meaning and functions.

### 3.1 Power line transmission

Also a PLT (power line transmission) is possible as a last mile system with the selected 3-tone transmission. Some manufacturers already make 95 -140 kHz frequency band equipments with 5 kbps bit rates using BPSK modulation according to GENELEC and FCC regulations.

The responsibility to give warnings is obviously given to the authorities, thus commercial FM receivers and PLT systems may be not the sole last mile solution to the early warning system for tsumani alerts. The normal acoustic signal used for fire alarms is probably the best and fastest way if triggered instantly by the 3-tone from the public broadcasting.

## 4 SECURED TRANSMISSION

It is well known that wireless Internet access or LANs are open for recording, interfering etc. There is no standardization for security in the physical OSI reference level. Thus there is a need for securing data.

### 4.1 Generation and Detection of Secured Three-Tone Signals

To generate waveforms in a digital way we have a flexible method in software modem technology, using OFDM (orthogonal FDM) in a band-limited way as presented in formula (Kifle et. al. 2001)

$$x(t) = \sum_{i=0}^{+\infty} \sum_{k=0}^{N_s-1} \sqrt{\{[s_I^2(k) + s_Q^2(k)]\}} p(t - k \frac{T_s}{N} - iT_s) \quad (3)$$

Where the waveform  $x(t)$  is piecewise continuous ( $i=0 \dots \infty$ ),  $N_s$  is the number of samples ( $k=0 \dots N_s-1$ ) in symbol and symbol time is  $T_s$ .

A 3-tone receiver is easily implemented with a software algorithm using a DFT (discrete Fourier transform) method as described in figure 5. In the proposed simple case only 3-DFT-filters are needed for decoding of a 3-tone signal or a MFC (multi frequency code) signal. The detection of the carriers (MFSK-signal,  $M=3$ ) is

made from the sampled signal  $x[n \cdot \Delta(t)]$  presented in figure 5. Generally a soft detection method is used as the symbol decoding system of modem. In this case as a demodulator part of the soft modem gives the amplitude and phase constellation of symbols in the piece-wise continuous data signal stream.

It is easy to mathematically describe and in real time detect continuous or discontinuous waveforms, formulae (2) - (3). In transmission of bits we select the symbol rate  $R_s$  and we set the associated number of samples in a symbol  $N$  together with a proper sample rate  $f_s$  according to the formula 4 as

$$R_s = f_s / N \quad (4)$$

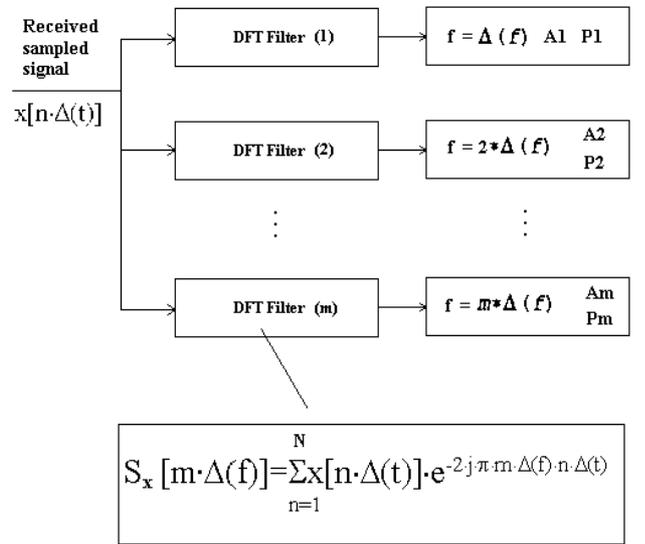


Figure 5: DFT Method

Selecting the proper digital modulation method according to the channel characteristics and BER performance requirements BER vs S/N we get the bit rate

$$R_b = kMR_s \quad (5)$$

Where we use  $k$  carriers,  $M$  bits per symbol and the selected symbol rate is  $R_s$ . This is an adaptive selection of the modulation method (parameters). Detection of  $f$ ,  $A$  and  $P$  is illustrated in figure 5.

Figure 6 (next page) presents the amplitude sensitivity of soft detection method in AWGN channel. At low S/N < 10 dB ratio the normalized amplitude of three orthogonal multi-tones has a high deviation. However, the alarm is detectable by setting the threshold level of the signals properly. Figure 6 presents an orthogonal signal space of three tones. There is AWGN jamming more harmful than other interferences.

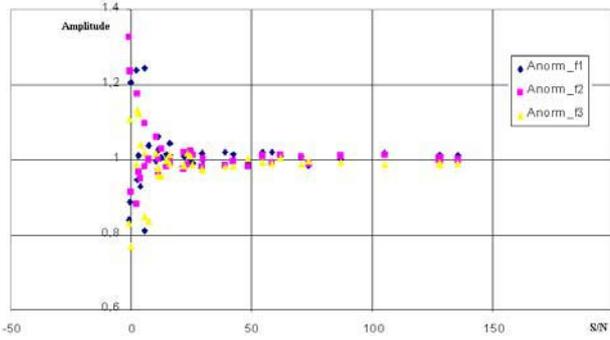


Figure 6: Soft 3-Tone Detection in AWGN Channel

## 4.2 Securing the Transmission

This proposal includes the use of band-limited frequency hopping  $m < (f)$ ,  $m=0 \dots M-1$ . A vector  $F = \{F_M\}$  defines in general the  $M$  frequencies used in the hopping as.

$$\{F_M\} = [f_0, f_1, f_2, \dots, f_k, \dots, f_{M-1}] \quad (6)$$

Let the transmitted signal (tsunami warning signal) be one of a multiplex of  $M$  signals with the hopping code  $\{C\}$  for the particular message signal as

$$\{C_{k,N}\} = \begin{bmatrix} c_{1,1} & c_{1,2} & \dots & c_{1,N} \\ c_{2,1} & c_{2,2} & \dots & c_{2,N} \\ \cdot & \cdot & \cdot & \cdot \\ c_{M,1} & c_{M,2} & \dots & c_{M,N} \end{bmatrix} \quad (7)$$

Where  $N$  message elements  $c_{k,N}$  (one in each column) are 1,  $k=1 \dots N$ , while all other elements are 0. In the similar way hopping codes for other messages can be constructed in a hopping system.

One gets the decoded signal  $S$  in a general case as a vector of signal carriers  $F$  in a matrix operation as

$$S = CF^T \quad (8)$$

Hopping sequence is secret i.e. the carriers are selected with a secure random process known only by the two end users of the secure adaptive end-to-end communication.

All these parameters are well known in DFT theory. The frequency parameter is  $m$  and it is associated with the frequency selectivity  $\Delta(f)$ . This  $\Delta(f)$  can be set with the sampling rate  $f_s$  and the number of samples  $N$  to a minimum value  $f_s/N$ , which is the symbol rate, formula (2). The importance of the adaptive change of  $N$  is obvious. The adaptive software filter can be made as narrow as needed and thus it can be adjusted to the adaptive waveform and the channel. The bandwidth used in the symbol transmission is calculated by DFT during each symbol time only (piecewise continuous system). We should not calculate over any discontinuity points between symbols. In theory using high sampling rate in detection we can use very narrow sub-bands for

multiple carriers. There are a few practical references for narrow-band multi-carrier frequency modems.

## 5 DISCUSSION

We have tested and simulated earlier the use of adaptive waveforms in different channels of radio, telephone and tactical delta-modulated networks. We have calculated some examples over 200 kbit/s bit rates with ten carrier frequencies or channels. In practice some of these high-speed data transmissions may need more bandwidth than a regular voice grade circuit (300-3400 Hz). In modern military VHF radio systems we already have available about 30 kHz wide frequency bands. We believe that older VHF radio equipment, which have 30 kHz bandwidths, are easily modernised for data transmission. They may be then connected to the laptops and used as data communication equipment in alert systems.

## REFERENCES

- NOOA Home Page. 2004. *NOOA Magazine* at <http://www.noaanews.noaa.gov/stories2004/s2357.htm>.
- CCITT. 1989, *IX<sup>th</sup> Plenary Assembly*. Melbourne 14-25 November 1988, Blue Book, ITU, Geneva.
- Lallo P. 1999. "Investigation of Data Transmission over an Adaptive Delta Modulated Voice Channel by Simulations Using a Spreadsheet Program." *Report T45*, ISBN 951-22-4770-4, ISSN 0356-5087, pp 30, Helsinki University of Technology, Communications Laboratory, Espoo.
- Kifle, M.; M. Andro; and M. J. Vanderaar. 2001. *An OFDM System Using Polyphase Filter and DFT Architecture for Very High Data Rate Applications*. NASA/TM—2001-210813, National Aeronautics and Space Administration Glenn Research Center, Prepared for the 19th International Communications Satellite Systems Conference and Exhibit cosponsored by the AIAA, CNES, ESA, and SUPAERO, Toulouse, France, April 17–20, 2001.

## AUTHOR BIOGRAPHY

**PAULI R. U. LALLO** born in 1944 in Joensuu, Finland went in 1963 to the Technical University of Helsinki, where he studied telecommunications and applied electronics. He obtained M.Sc 1969 and Tech.Lic in 1975. First he has worked in Post and Telecommunication Administration (PLH) in Finland a couple of years, in Adult Education Organizations 11 years and in a small Developing Company. Between 1970-1975 and since 1988 he worked in the Finnish Defence Forces (FDF). His rank is LTC in engineering. He has been the Chief Scientist of the Signals and Electrotechnical School since 1.1.2003 (former Signals School), retired on 1.1.2005

# NEW EXTENSIONS OF THE CAYLEY-HAMILTON THEOREM WITH APPLICATIONS

Tadeusz Kaczorek  
 Technical University of Białystok  
 Faculty of Electrical Engineering  
 Street 45D Wiejska, 15-351 Białystok  
 E-mail: kaczorek@isep.pw.edu.pl

## KEYWORDS

Extension, Cayley-Hamilton theorem, rectangular matrix, discrete-time, continuous-time, system, delay, singular.

## ABSTRACT

New extension of the classical Cayley-Hamilton theorem for rectangular matrices, block matrices, discrete-time and continuous-time systems with delays, and singular systems are presented. Some applications of the extensions and illustrating examples are also given.

## INTRODUCTION

The classical Cayley-Hamilton theorem (Gantmacher 1974, Kaczorek 1988, Lancaster 1969) says that every square matrix satisfies its own characteristic equation. The Cayley-Hamilton theorem has been extended to rectangular matrices (Kaczorek 1995, Kaczorek 1988), block matrices (Kaczorek 1995), pairs of commuting matrices (Chang and Chan 1992, Lewis 1982, Lewis 1986, Kaczorek 1995), pairs of block matrices (Kaczorek 1998, Kaczorek 1988) and standard and singular two-dimensional linear (2-D) systems (Kaczorek 1992/1993, Kaczorek 1995, Smart and Barnett 1989, Theodoru 1989).

The Cayley-Hamilton theorem and its generalizations have been used in control systems, electrical circuits, systems with delays, singular systems, 2-D linear systems, etc., (Kaczorek 1992/1993, Busłowicz 1981, Busłowicz 1982, Kaczorek 1994, Lewis 1982, Mertizios and Christodoulous 1986).

In (Kaczorek 2005) the Cayley-Hamilton theorem has been extended to  $n$ -dimensional ( $n$ -D) real polynomial matrices. An extension of the Cayley-Hamilton theorem for discrete-time linear systems with delay has been given in (Busłowicz and Kaczorek 2004).

In this paper an overview of new generalizations of the Cayley-Hamilton theorem with some applications is presented. The classical Cayley-Hamilton theorem will be extended for rectangular matrices, block matrices, discrete-time and continuous-time systems with delays and singular systems. The extensions of the Cayley-Hamilton theorem will be illustrated by examples.

## CAYLEY-HAMILTON THEOREM FOR SQUARE AND RECTANGULAR MATRICES

Let  $C^{n \times m}$  be the set of complex ( $n \times m$ ) matrices.

**Theorem 1.** (Cayley-Hamilton theorem). Let  $A \in C^{n \times m}$  and

$$p(s) = \det[I_n s - A] = \sum_{i=0}^n a_i s^i \quad (a_n = 1) \quad (1)$$

be the characteristic polynomial of  $A$ , where  $I_n$  is the ( $n \times n$ ) identity matrix.

Then

$$p(A) = \sum_{i=0}^n a_i A^i = 0_n \quad (2)$$

where  $0_n$  is the ( $n \times n$ ) matrix.

The classical Cayley-Hamilton theorem can be extended to rectangular matrices as follows (Kaczorek 1988).

**Theorem 2.** (Cayley-Hamilton theorem for rectangular matrices). Let

$$A = [A_1 \quad A_2] \in C^{m \times n}, \quad A_1 \in C^{m \times m}, \quad A_2 \in C^{m \times (n-m)}, \quad (n > m) \quad (3)$$

and

$$p_{A_1} = \det[I_m s - A_1] = \sum_{i=0}^m a_i s^i \quad (a_m = 1) \quad (4)$$

be the characteristic polynomial of  $A_1$

Then

$$\sum_{i=0}^m a_{m-i} [A_1^{n-i} \quad A_1^{n-i-1} A_2] = 0_{mn} \quad (5)$$

where  $0_{mn}$  is the ( $m \times n$ ) zero matrix.

**Theorem 3.** Let  $A = \begin{bmatrix} A_1 \\ A_2 \end{bmatrix} \in C^{m \times n}$ ,  $m > n$  and let the characteristic polynomial of  $A_1$  have the form (4). Then

$$\sum_{i=0}^n a_{n-i} \begin{bmatrix} A_1^{m-i} \\ A_2 A_1^{m-i-1} \end{bmatrix} = 0_{mn} \quad (6)$$

### CAYLEY-HAMILTON THEOREM FOR BLOCK MATRIX

The classical Cayley-Hamilton theorem can be also extended for block matrices (Kaczorek 1998, Kaczorek 1988).

**Theorem 4** (Cayley-Hamilton theorem for block matrices). Let

$$A = \begin{bmatrix} A_{11} & \dots & A_{1m} \\ \vdots & \ddots & \vdots \\ A_{m1} & \dots & A_{mm} \end{bmatrix} \in C^{mn \times mn} \quad (7)$$

where  $A_{ij} \in C^{n \times n}$  are commutative, i.e.

$$A_{ij} A_{kl} = A_{kl} A_{ij} \quad \text{for all } i, j, k, l = 1, 2, \dots, m \quad (8)$$

Let

$$P(S) = \det[I_m \otimes S - A] = S^m + D_1 S^{m-1} + \dots + D_{m-1} S + D_m \quad (9)$$

be the matrix characteristic polynomial of  $A$ , where  $S \in C^{n \times n}$  is the matrix (block) eigenvalue of  $A$ ,  $\otimes$  denotes the Kronecker product of matrices (Kaczorek 1988).

Then

$$P(A) = \sum_{i=0}^m (I_m \otimes D_{m-i}) A^i = 0 \quad (D_0 = I_n) \quad (10)$$

The matrix (9) is obtained by developing the determinant of the matrix  $[I_n \otimes S - A]$  considering its commuting blocks as scalar entries (Kaczorek 1988).

**Theorem 5** (Cayley-Hamilton theorem for rectangular block matrices). Let  $\bar{A} = [A \ A_2] \in C^{mn \times (mn+p)}$  and let the matrix characteristic polynomial of  $A$  have the form (14). Then

$$\sum_{i=0}^m (I_m \otimes D_{m-i}) [A^{i+1} \ A^i A_2] = 0 \quad (D_0 = I_n) \quad (11)$$

The dual theorem has the form

**Theorem 6.** Let

$$\bar{A} = \begin{bmatrix} A \\ A_2 \end{bmatrix} \in C^{(mn+p) \times mn}, \quad A \in C^{mn \times mn}, \quad A_2 \in C^{p \times mn}$$

and let the matrix characteristic polynomial of  $A$  have the form (14). Then

$$\sum_{i=0}^m \begin{bmatrix} A \\ A_2 \end{bmatrix} (I_m \otimes D_{m-i}) A^i = 0 \quad (D_0 = I_n) \quad (12)$$

### CAYLEY-HAMILTON THEOREM FOR SYSTEMS WITH DELAYS

#### Discrete time-systems

Consider the discrete-time linear system with  $h$  delays described by the equation

$$x_{i+1} = A_0 x_i + A_1 x_{i-1} + \dots + A_h x_{i-h} + B u_i \quad (13)$$

where  $x_i \in C^n$ ,  $u_i \in C^m$  are the state and input vectors,  $A_k \in C^{n \times n}$ ,  $k=0, 1, \dots, h$  and  $B \in C^{n \times m}$ .

The characteristic polynomial of (13) has the form

$$p(z) = \det \begin{bmatrix} I_n z - A_0 & -A_1 & \dots & -A_{h-1} & -A_h \\ -I_n & I_n z & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & -I_n & I_n z \end{bmatrix} = \det [I_n z^{h+1} - A_0 z^h - A_1 z^{h-1} - \dots - A_h] = z^N + a_{N-1} z^{N-1} + \dots + a_1 z + a_0, \quad N = n(h+1) \quad (14)$$

Let

$$\Phi_{i+1} = A_0 \Phi_i + A_1 \Phi_{i-1} + \dots + A_h \Phi_{i-h} \quad (15)$$

and

$$\Phi_0 = I_n \quad \text{and} \quad \Phi_i = 0 \quad \text{for } i < 0 \quad (16)$$

Knowing the matrices  $A_k$ ,  $k=0, 1, \dots, h$  and using (15), (16) we may compute the matrices  $\Phi_i$  for  $i=1, 2, \dots$ .

**Theorem 7.** The matrices  $\Phi_i$  for  $i=1, 2, \dots$  defined by (15) and (16) satisfy the equation

$$\sum_{i=0}^N a_{i+k} \Phi_{i+k} = 0 \quad \text{for } k = 0, 1, \dots \quad (a_N = 1) \quad (17)$$

where  $a_i$ ,  $i=0, 1, \dots, N-1$  are the coefficients of the characteristic polynomial (14).

**Proof.** From definition of the inverse matrix we have

$$\begin{aligned} & [I_n z^{h+1} - A_0 z^h - A_1 z^{h-1} - \dots - A_h]_{ad} = \\ & = [I_n z^{h+1} - A_0 z^h - A_1 z^{h-1} - \dots - A_h]^{-1} \times \\ & \det [I_n z^{h+1} - A_0 z^h - A_1 z^{h-1} - \dots - A_h] \end{aligned} \quad (18)$$

From (15) and (16) it follows that





$$\text{Adj}[Es - A] = B_{n-1}s^{n-1} + \dots + B_1s + B_0 \quad (42)$$

be the adjoint matrix of  $[Es - A]$ .

From definition of the inverse matrix and (40), (42) we have

$$\begin{aligned} [Es - A][B_{n-1}s^{n-1} + \dots + B_1s + B_0] &= \\ = I_n(a_r s^r + a_{r-1}s^{r-1} + \dots + a_1s + a_0) \end{aligned} \quad (43)$$

The comparison of the coefficients at the same powers of  $s$  of the equality (43) yields

$$\begin{bmatrix} E & 0 & 0 & \dots & 0 & 0 & 0 \\ -A & E & 0 & \dots & 0 & 0 & 0 \\ 0 & -A & E & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & -A & E & 0 \\ 0 & 0 & 0 & \dots & 0 & -A & E \\ 0 & 0 & 0 & \dots & 0 & 0 & -A \end{bmatrix} \begin{bmatrix} B_{n-1} \\ B_{n-2} \\ B_{n-3} \\ \vdots \\ B_2 \\ B_1 \\ B_0 \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ a_r I_n \\ \vdots \\ a_1 I_n \\ a_0 I_n \end{bmatrix} \quad (44)$$

Premultiplying (44) by the row matrix

$$\begin{bmatrix} A^n & A^{n-1}E & A^{n-2}E^2 & \dots & AE^{n-1} & E^n \end{bmatrix} \quad (45)$$

and using (39) we obtain the equation (41). ■

**Example 3.** Let

$$E = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ a & 0 & 0 \end{bmatrix}, \quad A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (46)$$

The pair (46) satisfies the condition (39).

In this case

$$\det[Es - A] = \begin{vmatrix} -1 & s & 0 \\ 0 & -1 & s \\ as & 0 & -1 \end{vmatrix} = as^3 - 1 \quad (a_0 = 1, a_3 = a) \quad (47)$$

Using (46) and (47) we obtain for  $a \neq 0$

$$\begin{aligned} \sum_{i=0}^3 a_i A^i E^{3-i} &= -E^3 + aA^3 = \\ &= -a \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} + a \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \end{aligned}$$

and for  $a = 0$

$$a_0 E^3 = -E^3 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}^3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

## CONCLUDING REMARKS

New extensions of the classical Cayley-Hamilton theorem are presented with some applications and illustrating examples.

The Cayley-Hamilton theorem has been extended for rectangular matrices, block matrices, discrete-time and continuous-time systems with delays and singular systems.

It has been shown that rectangular matrices satisfy many different algebraic equations and the matrices  $A_k$ ,  $k=0,1,\dots,h$  of the continuous-time systems (25) with  $h$  delays satisfy  $nh+1$  algebraic equations.

With slight modifications the presented extensions for systems with delays can be extended for rectangular matrices and block matrices.

## REFERENCES

- Busłowicz M., 1981. *Algorytm obliczania quasi wielomianu charakterystycznego wielowymiarowego układu liniowego stacjonarnego z niewspółmiernymi opóźnieniami na podstawie równania stanu*, Archiwum Automatyki i Telemekhaniki, Tom XXVI, z. 1, pp. 125-131.
- Busłowicz M., 1984. *Inversion of characteristic matrix of the time-delay systems of neural type*, Foundations of Control Engineering, vol. 7, No 4, pp. 195-210.
- Busłowicz M., Kaczorek T., 2004. *Reachability and minimum energy control of positive linear discrete-time systems with one delay*, Proc. 12<sup>th</sup> Mediterranean Conf. on Control and Automation, Kasadasi-Izmir Turkey
- Chang F. R. and Chan C. N., 1992. *The generalized Cayley-Hamilton theorem for standard pencils*, System and Control Lett. 18, pp 179-182
- Gantmacher F. R., 1974. *The theory of matrices*, vol. 2, Chelsea.
- Kaczorek T., 1988. *Vectors and Matrices in Automation and Electrotechnics*, WNT Warszawa (in Polish)
- Kaczorek T., 1992/1993. *Linear Control Systems*, vol. I, II, Research Studies Press
- Kaczorek T., 1994. *Extensions of the Cayley-Hamilton theorem for 2D continuous-discrete linear systems*, Appl. Math. and Com. Sci., vol. 4, No 4, pp. 507-515.
- Kaczorek T., 1995. *An existence of the Cayley-Hamilton theorem for singular 2D linear systems with non-square matrices*, Bull. Pol. Acad. Techn. Sci., vol. 43, No 1, pp. 39-48.
- Kaczorek T., 1995. *An existence of the Cayley-Hamilton theorem for nonsquare block matrices and computation of the left and right inverses of matrices*, Bull. Pol. Acad. Techn. Sci., vol. 43, No 1, pp. 49-56.
- Kaczorek T., 1995. *Generalization of the Cayley-Hamilton theorem for nonsquare matrices*, Proc. Inter. Conf. Fundamentals of Electrotechnics and Circuit Theory XVIII-SPETO, pp. 77-83.

- Kaczorek T., 1998. *An extension of the Cayley-Hamilton theorem for a standard pair of block matrices*, Appl Math. and Com. Sci., vol. 8, No 3, pp. 511-516.
- Kaczorek T., 2005. *Generalization of Cayley-Hamilton theorem for  $n$ -D polynomial matrices*, IEEE Trans. Autom. Contr. (in Press)
- Lancaster P., 1969. *Theory of Matrices*, Acad. Press.
- Lewis F L., 1982. *Cayley-Hamilton theorem and Fadeev's method for the matrix pencil  $[sE-A]$* , Proc. 22<sup>nd</sup> IEEE Conf. Decision Control, pp. 1282-1288
- Lewis F. L., 1986. *Further remarks on the Cayley-Hamilton theorem and Fadeev's method for the matrix pencil  $[sE-A]$* , IEEE Trans. Automat. Control, 31, pp. 869-870.
- Mertizios B. G and Christodoulous M. A., 1986. *On the generalized Cayley-Hamilton theorem*, IEEE Trans. Automat. Control, 31, pp. 156-157.
- Smart N M. and Barnett S., 1989. *The algebra of matrices in  $n$ -dimensional systems*, Math. Control Inform., 6, pp. 121-133.
- Theodoru N. J., 1989.  *$M$ -dimensional Cayley-Hamilton theorem*, IEEE Trans. Automat. Control, AC-34, No 5, pp. 563-565.



**TADEUSZ KACZOREK**

born 27. 04. 1932 in Poland, received the MSc. PhD and DSc degrees from Electrical Engineering of Warsaw University of Technology in 1956, 1962 and 1964, respectively. In the period 1968-69 he has the dean of Electrical Engineering Faculty and in the period 1970-73 he was the prorector of

Warsaw University of Technology. Since 1971 he has

Research under grant KBN 3TPA 06627

been professor and since 1974 full professor at Warsaw University of Technology. In 1986 he was elected a corresp. member and in 1996 full member of Polish Academy of Sciences. In the period 1988-1991 he was the director of the Research Centre of Polish Academy of Sciences in Rome. In May 2004 he was elected the honorary member of the Hungariar Academy of Sciences. He was awarded by the title doctor honoris causa by the University of Zielona Góra (2002), the Technical University of Lublin (2004), the Technical University of Szczecin (2004) and Warsaw University of Technology (2004).

His research interests cover the theory of systems and the automatic control systems theory, specially, singular multidimensional systems, positive multidimensional systems and singular positive 1D and 2D systems. He has initiated the research in the field of singular 2D and positive 2D systems. He has published 17 books (5 in English) and almost 700 scientific papers in journals like IEEE Transactions on Automatic Control, IEEE Transactions on Neural Networks, Multidimensional Systems and Signal Processing, International Journal of Control, Bull. Pol. Acad. Science, etc. and proceedings of conferences and world congresses. He has given invited lectures in more then 50 universities in USA, Canada, UK, Germany, Italy, Japan, Greece etc. He has been a member of many international committees and programme committees. He supervised 61 Ph.D. theses. He is editor-in-chief of Bulletin of Polish Academy of Scences, Tech. Sciences and editorial member of about ten international journals.

# MODULAR MODELLING AND ANALYSIS OF TIME-DEPENDENT SYSTEMS

Franco Cicirelli, Angelo Furfaro, Libero Nigro, Francesco Pupo  
Laboratorio di Ingegneria del Software  
Dipartimento di Elettronica Informatica e Sistemistica  
Università della Calabria, 87036 Rende (CS) – Italy  
E-mail: *{f.cicirelli,a.furfaro}@deis.unical.it {l.nigro,f.pupo}@unical.it*

## KEYWORDS

Modular modelling, real-time systems, time Petri nets, discrete-event simulation, model checking, UPPAAL.

## ABSTRACT

This paper describes an approach to the analysis of time-dependent systems which combines discrete-event simulation and model-checking techniques. The approach rests on Merlin and Farber's Time Petri Nets (TimePNs) and is supported by a Java toolbox –TPN Designer– which enables graphical modelling, simulation and translation into UPPAAL/Timed Automata, for exhaustive state space verification, of modular TimePN models. The paper discusses the potential of the proposed approach through its application to a real-time system model.

## INTRODUCTION

The design of a time-dependent system (e.g. an embedded real-time system, a communication protocol etc.) must fulfil a set of functional and non-functional requirements. Non functional requirements include *timeliness*, that is the ability of the software system to provide responses to events (e.g., originated in an external controlled environment) which must be functionally correct *and* furnished within precise and predictable timing constraints (e.g. deadlines). To ensure a system development compliant with timing requirements, the use of formal tools is mandatory. A formal language can favour unambiguous specification of system behaviour and, most importantly, can allow reasoning on and assessment of system properties.

In the work described in this paper, Merlin and Farber's Time Petri Nets –TimePNs– (Merlin and Farber, 1976) are chosen as the modelling language. TimePNs permit rigorous specification of a time-dependent system by associating a time interval to transitions (activities) which constrains transition firing. More precisely, let  $t$  be a transition and  $[a,b]$ ,  $0 \leq a \leq b$ ,  $b$  can be  $\infty$ , the time interval associated with  $t$  ( $a$  is said the *earliest firing time*,  $b$  is the *latest firing time* of  $t$ ). Let  $\tau$  be an instant in time when  $t$  gets enabled in the usual sense of Petri nets (Murata, 1989). Provided  $t$  is continuously enabled, it cannot fire before  $\tau+a$  but should fire before or at

$\tau+b$  unless  $t$  is disabled by the firing of a conflicting transition.

TimePNs were recently added to a Java toolbox –TPN DESIGNER (Carullo *et al.*, 2003)– which supports modelling and discrete-event simulation of complex and modular systems. TPN DESIGNER consists of two integrated sub environments: CAD and ENGINE. CAD is concerned with model editing and project management. ENGINE supports compilation, debugging, simulation and statistical data collection. A large system can be split and step-wise refined into modular units called *pages*. A page has an interface of *input/output ports* and hides an internal behaviour expressed by a GSPN-like (Marsan *et al.*, 1984)(Marsan *et al.*, 1987) subnet which in turn can be organized in sub pages and so forth recursively to any arbitrary depth. Pages can be spatially replicated to create linear (pipeline) or bi-dimensional (grid) topologies.

Model configuration requires port interconnections and net parameters (marking of places and attributes of transitions) to be established. Ports can be linked to one another in CAD through the mouse. More in general, a simple but powerful *scripting language* can be used to “program” port interconnections. Scripting can occur at *page level* or at *model* (or *root*) *level*. Page level scripting can be exploited to establish port bindings among instances (*siblings*) of a same page model. Root level scripting can be used to define connections between arbitrary but compatible subjects (places or transitions) possibly belonging to different pages, and to furnish values for model parameters. Root level scripting is essential for model *scalability*. A properly configured model can be scaled (e.g. page multiplicities changed and model parameters redefined) simply by modifying a few values in the root level script. A hierarchical model is compiled into a flat representation which is then managed by ENGINE.

For statistics data collection a flexible *watch system* can be used. Common statistics like gathering minimum, maximum and average number of tokens/firings for selected places/transitions can be achieved by adding one or more watches to selected entities in the graphical ENGINE sub-environment. To cope with general statistics calculations, aspect-oriented *monitors*, i.e.

external Java classes, can be prepared and transparently weaved to TPN DESIGNER control engine. A monitor is event-driven. Specific event occurrences can be observed and application-dependent actions correspondingly executed. Multiple monitors can possibly be jointly used to observe model execution.

General TPN DESIGNER features can be exploited for managing TimePN models. In addition, the ENGINE sub-environment was extended so as to compile a TimePN model in terms of timed automata (Alur & Dill, 1994) in the context of the UPPAAL toolbox (Behrmann *et al.*, 2004), using a template process corresponding to basic TimePN transition. In this way, a TimePN model can be analyzed using simulation and (possibly) model checking (Clarke *et al.*, 2000) i.e. exhaustive verification of system properties against the whole state space of the model. UPPAAL was chosen because of its compact data structures and efficient model checker. This paper argues that simulation can help model checking by suggesting estimates of response time bounds which can then be refined by using a few queries in the context of the verifier. Very often, in fact, the modeller has to tentatively issue many times a given query to the verifier, each time with a different parameter value, in order to determine a worst case response time. All of this can be highly expensive in the wallclock time for complex models. On the other hand, for a large system model, exhaustive verification can be practically prohibitive. In these cases discrete-event simulation, although it can never replace model checking, can anyway furnish important indications about quantitative temporal behaviour of the system.

This paper describes the proposed approach to systems modelling and analysis centred on TPN DESIGNER and UPPAAL, by reporting its application to a real-time modelling example.

## A ROBOT CONTROLLER MODEL

The following considers a distributed robot controller system (Gerber & Lee, 1992) whose purpose is to periodically collect positional data from four tactile sensors, to convert the data into real world coordinates, to integrate the data, to determine the robot next position and to send the new position to the robot. The system consists of four tactile sensors, four conversion processes, four single buffered channels, an integrator and the robot itself.

Communications between interacting components are assumed to be synchronous (rendezvous). Each communication and every internal action are supposed to consume 1 time unit (e.g., 1 ms). Each sensor has a period of 6 time units within which it takes a sample reading from the environment and then attempts to communicate the result to the corresponding converter. Since the environment is supposed to be subject to very rapid changes, the communication between a sensor and

its converter must be made within a hard deadline of 4 time units, able to ensure that each sensor terminates its internal and i/o activities before the next period is commenced. Another deadline of 1 time unit exists between a converter which is ready to transmit its converted data and its associated buffered channel. The integrator must first receive the four communications from the buffered channels, in any order, and then prepares the next positional data for the robot. From when the robot is ready to accept its next command, it must do so within 11 time units (hard deadline). Fig. 1 shows a TPN DESIGNER modular model for the system, under the hypothesis of maximal parallelism, i.e. that each component is mapped onto a distinct physical processor.

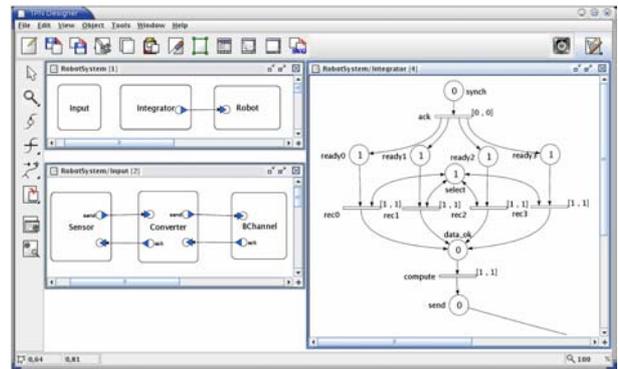


Figure 1: A modular TPN DESIGNER model for the robot controller system

The **Input** page encapsulates **Sensor**, **Converter** and **BChannel** sub pages. To obtain the required system topology, the **Integrator** and **Robot** pages are instantiated only once (default), whereas the **Input** page (and implicitly its contained sub pages) has a monodimensional 4-multiplicity. Interacting page instances within **Input** have input/output ports directly linked by the mouse. Connections between the four instances of the **BChannel** page and the single instance of the **Integrator** are accomplished by the root level script shown in Fig. 2.

```
(1) set Input={4, 1};
(2) link /Input[0,0]/BChannel[0,0]/send to /Integrator[0,0]/rec0 with 1;
(3) link /Input[1,0]/BChannel[0,0]/send to /Integrator[0,0]/rec1 with 1;
(4) link /Input[2,0]/BChannel[0,0]/send to /Integrator[0,0]/rec2 with 1;
(5) link /Input[3,0]/BChannel[0,0]/send to /Integrator[0,0]/rec3 with 1;
(6) link /Integrator[0,0]/rec0 to /Input[0,0]/BChannel[0,0]/reply with 1;
(7) link /Integrator[0,0]/rec1 to /Input[1,0]/BChannel[0,0]/reply with 1;
(8) link /Integrator[0,0]/rec2 to /Input[2,0]/BChannel[0,0]/reply with 1;
(9) link /Integrator[0,0]/rec3 to /Input[3,0]/BChannel[0,0]/reply with 1;
(10) link /Robot[0,0]/receive to /Integrator[0,0]/synch with 1;
```

Figure 2: Root-level script for the Robot model

Line (1) declares the multiplicity of the **Input** page, i.e. 4x1 (4 rows and 1 column). Lines (2) to (5) specify that the **send** place in the various **BChannel** instances (see Fig. 5) has to be linked with the  $rec_j$ ,  $j=0, \dots, 3$  transition of the **Integrator** (Fig. 6). Lines from (6) to (9) serve to

connect the  $rec_j$  transitions of the Integrator with the reply place of the corresponding BChannel instance. Line (10) states that the receive transition of the Robot page (Fig. 7) must be linked to the synch place of the Integrator. All the arcs corresponding to the link instructions have unitary weight (clause with 1). Figg. from 3 to 7 depict the internal sub net of each model page together with its initial marking. As a rule, a communication is followed by an  $ack$  event sent back to the sender at the end of the synchronization. In addition, the communication delay is spent in the receiver so that the  $ack$  event is instantaneous (time interval  $[0,0]$ ).

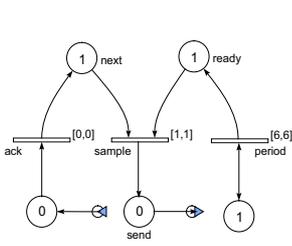


Figure 3: Sensor subnet

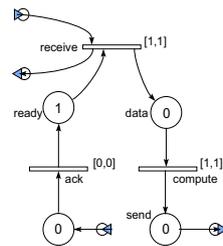


Figure 4: Converter subnet

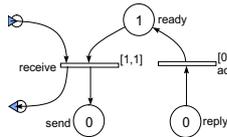


Figure 5: BChannel subnet

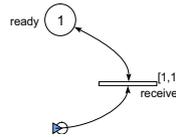


Figure 6: Robot subnet

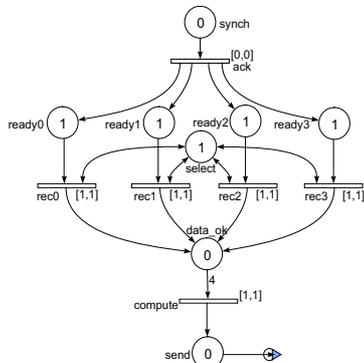


Figure 7: Integrator subnet

Figg. from 3 to 7 depict the internal sub net of each model page together with its initial marking. As a rule, a communication is followed by an  $ack$  event sent back to the sender at the end of the synchronization. In addition, the communication delay is spent in the receiver so that the  $ack$  event is instantaneous (time interval  $[0,0]$ ).

In the **Sensor** subnet, one token in the **ready** place indicates the sensor is ready to sample the environment. However, reading the sample effectively requires one token in the **next** place too, which mirrors the fact, in general, that previous data has already been communicated to the converter. Satisfaction of sensor deadline can easily be related to having no more than

one token at all times in the **ready** place. **Converter** sub net ensures that first a data is received from a sensor (transition **receive**), then the conversion process is accomplished (**compute** transition) after that a communication is attempted with the corresponding buffered channel, whose completion is witnessed by the firing of the **ack** transition. **BChannel** sub net first receives a converted data from a converter then tries to transmit it to the integrator. The **ack** message is generated as soon as the **send** place empties. The **Integrator** sub net is able to receive data from the four buffered channels through the  $rec_j$  transitions.

However, only one communication can take place at each time (due to the **select** place). When all the four data has been achieved, the integrator computes next robot position (**compute** transition) and, finally, tries to communicate it to the robot. At the end of this communication, a token is generated in the **synch** place and then the integrator repeats its behaviour. The **Robot** sub net simply receives next positional information from the integrator (**receive** transition in Fig. 7) and then is again available for the next communication. At each firing of the **receive** transition of the robot, a token is generated in the **synch** place of the integrator through the link established by the root level script (see line (6) in Fig. 2).

A modular and hierarchical TPN DESIGNER model is flattened into a representation made up of concrete subjects (places, transitions and arcs). Meta data like ports are eliminated and port inter-connections and links introduced by scripts are replaced by effective arcs.

## SIMULATION EXPERIMENTS

The ENGINE sub environment of TPN DESIGNER simulates a TimePN model by generating the occurrence time of a transition as uniformly distributed in the transition time interval. Toward this, an unbounded interval like  $[a, \infty]$  is actually managed as  $[a, Tend+1]$  where  $Tend$  is the simulation time limit. Moreover, according to TimePN semantics of transition firing, the race policy is implemented. All the enabled transitions at a given time are scheduled to occur to their proposed firing time. Firing of a transition, though, due to conflicts, can disable some pending scheduled transitions which are then removed from the event list. At each enabling, and also after its own firing, a transition is always scheduled by re-sampling its occurrence time from the associated time interval. Transition firing is regulated by *single-server* policy, meaning that a multiple enabled transition will fire its enablings one at a time sequentially.

The robot system model was simulated for  $10^6$  time units and simulation data were collected to test “satisfaction” of timing constraints (deadlines). For demonstration purposes, both the *watch system* and the

monitor facility were used. In particular, the *between-watch* (see Fig. 8) was employed for estimating the maximum and minimum temporal distance existing between two causally connected events, one of which acts as *cause* (or source) and the other as *effect* (or destination).

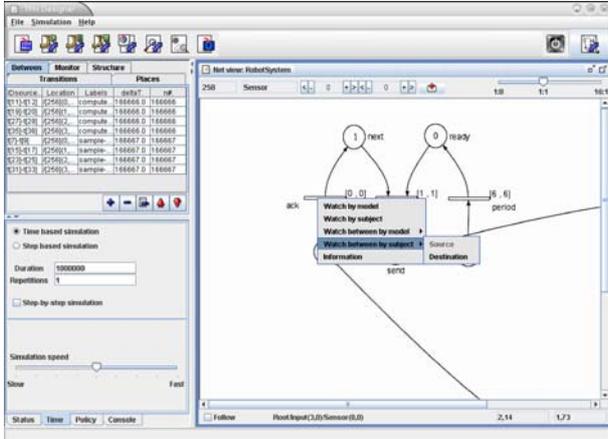


Figure 8. Screenshot of ENGINE showing between-watches in action

A distinct between-watch was associated to the *sample* (cause) and *ack* (effect) pair of events within each *Sensor* page instance, and to the *compute* (cause) and *ack* (effect) pair of events within each *Converter* page instance. The hard deadline of robot was instead estimated by programming a *RobotMonitor* Java class which is dynamically loaded in the context of ENGINE. A monitor class extends the *Monitor* base class, introduces suitable variables for collecting simulation output and redefines the three basic methods: *actionPerformed()*, *calculateStatistics()* and *showResults()*. *actionPerformed* is launched at each transition firing, *calculateStatistics* extracts simulation results at the end of a simulation run and accumulates them during a simulation batch, and *showResults* displays simulation outputs at the end of a simulation batch. Fig. 9 portrays an excerpt of the *actionPerformed* method of the *RobotMonitor* class. *lastFire* variable (initialized to zero) remembers the occurrence time of latest firing of the *receive* transition of the *Robot*. Other details should be self-explanatory.

```

public void actionPerformed( Event e ){
    if( e is Robot receive event ){
        double delay=e.timestamp-lastFire;
        if( delay<minDelay ) minDelay=delay;
        if( delay>maxDelay ) maxDelay=delay;
        lastFire=e.timestamp;
    }
}
//actionPerformed

```

Figure 9. An excerpt of *actionPerformed* method of *RobotMonitor*

After 5 simulation runs of the robot model, it emerged that the between-time in each sensor and in each

converter is bounded to 1 time unit, i.e. when a sensor or a converter is ready to communicate, the partner component is also found ready to engage in the synchronization.

In addition, the inter-receive time window for the robot was estimated to be [6,10], that is two consecutive occurrences of the receive transition of the *Robot* occur in the best case after 6 time units, and in the worst case after 10 time units. As a consequence, from the perspective of simulation, the system “fulfils” all its deadlines.

## UPPAAL TRANSLATION

The UPPAAL toolbox (Behrmann *et al.*, 2004)(UPPAAL on-line) allows to verify systems modelled as networks of timed automata (Alur & Dill, 1994) extended with integer variables, structured data types and channel synchronization (*rendezvous*). To permit model checking activities, TPN DESIGNER is able to translate automatically a TimePN model into UPPAAL by using the template automaton shown in Fig. 10 which reproduces semantics of general TimePN transition, according to the UPPAAL 3.5.3 version which supports C syntax and in particular C function declarations and loop constructs. Let  $T$  and  $P$  denote respectively the set of transitions and the set of places in a model,  $PRE$  and  $POST$  the (maximal) set of input places and output places of any transition. The translation generates the backward  $B_{|T| \times |PRE|}$  and forward  $F_{|T| \times |POST|}$  incidence matrices of a TimePN model, its marking vector  $M_{|P|}$  and the time interval vector  $I_{|T| \times 2}$ . An *Info struct* was defined which contains an *index* in  $M$  which selects a place, and the *weight* of an arc linking the place to a transition or vice versa. Each element of  $B$  or  $F$  is an *Info* value. Inhibitor arcs are allowed. An unrestricted time interval like  $[a, \infty]$  is represented in  $I$  as  $[a, -1]$ . Functions *enabled()*, *withdraw()* and *deposit()* which respectively check transition enabling, withdraw tokens from input places and generate tokens to output places, are declared locally to the *Transition* template and refer to the particular transition through its ID.

A disabled transition stays in the *Disabled* location. An enabled transition starts firing by resetting its clock  $x$  and moving to *Firing* if its time interval is strict, or to *U\_Firing* if its latest firing time is  $\infty$ . An enabled transition can complete its firing as soon as its clock goes beyond its earliest firing time. A time strict transition is obliged to complete its firing at its latest firing time, provided it is still enabled, by the invariant  $x \leq I[ID][1]$  attached to *Firing*. Transition firing, though, is non deterministic among transitions which can complete their firing at a given time.

Firing completion is an instantaneous and atomic process in two steps which involve the committed locations *Withdraw* and *Deposit* in which time is not

allowed to pass. The transition first removes tokens from its input places, then moves to the **Withdraw** location where it sends (*first step*) a broadcast synchronization signal over the **end\_fire** urgent and broadcast channel. This signal forces all the remaining transitions, disabled or under firing, to re-evaluate its enabling status. This is crucial for proper management of conflicts. A no longer enabled transition moves from **Firing/U\_Firing** to **Disabled** and resets its clock. The firing process continues by generating tokens in the output places and then by sending (see **Deposit** outgoing edges in Fig. 10) a second broadcast synchronization through **end\_fire** (*second step*) which permits detection of newly enabled transitions and non persistent transitions which lose their enabling due to inhibitor arcs.

The **Transition** automaton follows the *single server* firing policy. After its own firing, a still enabled transition is always regarded as a newly enabled one and its clock reset.

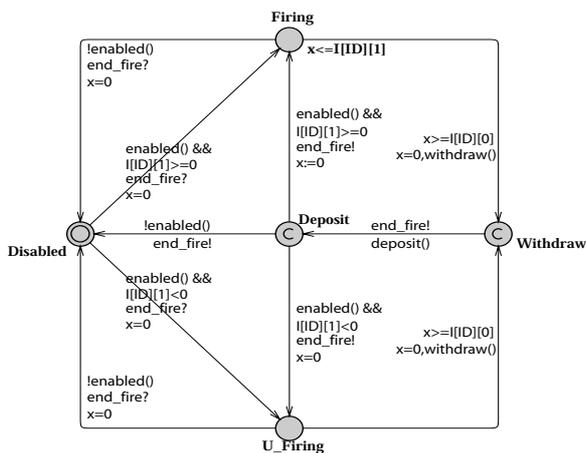


Figure 10: The Transition automaton

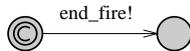


Figure 11. The Starter automaton

A final remark concerns model bootstrapping. A transition starts in its **Disabled** location. An initial broadcast synchronization ensured by the **Starter** automaton shown in Fig. 11 allows transitions enabled in the initial marking of the TimePN model to reach **Firing/U\_Firing** location. After that, **Starter** will take no part in the subsequent evolution of the model.

### About Translation Correctness

Correctness of the **Transition** template follows intuitively from its construction. A TimePN model translates into a collection of **|T|** instances of the automaton in Fig. 10, i.e. the corresponding UPPAAL model is the product of **|T|** automata corresponding to

the **|T|** transitions of the TimePN model, and the **Starter** automaton.

Translation correctness can be formally proved by showing that the semantics of the achieved UPPAAL model is *timed bisimilar* to the original TimePN model. An example of such a formal proof was described in (Cassez & Roux, 2004) which suggested a translation from TimePNs to timed automata based on a transition automaton very close to the automaton in Fig. 10. Cassez and Roux used a separate supervisor automaton to ensure the two broadcast synchronization steps used during the firing completion process. The equivalence of a TimePN model and its corresponding UPPAAL model enables TCTL model checking activities (Alur *et al.*, 1993)(Clarke *et al.*, 2000)(Behrmann *et al.*, 2004). In particular *safety* and *bounded-liveness* properties verified on the UPPAAL model can directly be interpreted on the original TimePN model. Actually, bounded TimePN models can be efficiently model checked using the translation proposed in this paper. A screenshot of TPN DESIGNER/ENGINE during UPPAAL translation of the Robot system is shown in Fig. 12.

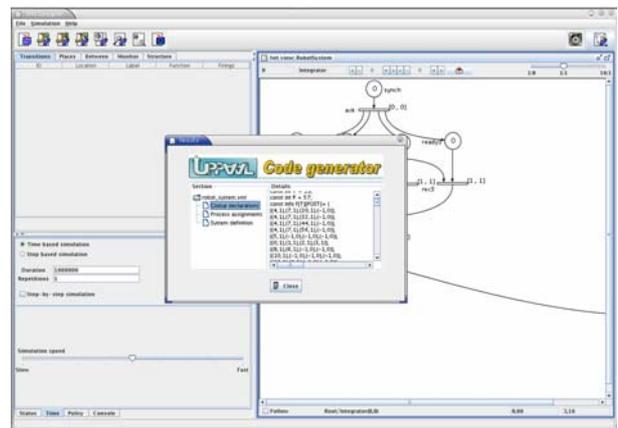


Figure 12. A screenshot of ENGINE during UPPAAL translation of the Robot system

### MODEL CHECKING THE ROBOT SYSTEM

The translation process associates by default a distinct clock to each transition of a TimePN model. However, different clocks are effectively required by concurrent or conflicting transitions. The UPPAAL version of the robot model was verified by using 21 clocks: 2 clocks for each **Sensor** instance (see Fig. 3)(one clock is needed by the **period** transition, the other clock is shared by **sample** and **ack** transitions which operates sequentially), 1 clock for each **Converter** instance, 1 clock for each **BChannel** instance, 4 clocks (one for each **rec<sub>j</sub>** conflicting transitions) for the **Integrator**, 1 clock for the **Robot**.

A first concern in model checking the achieved UPPAAL model of the robot system, was verification of basic safety property, i.e. that the system is live or free of deadlocks. The property, which cannot be demonstrated

in simulation, was verified by launching the following query to the model checker:

$A[] \text{!deadlock}$

The query asks if in *all* the states of the model state graph there is no deadlock. It is worth noting that a deadlock at the TimePN level, i.e. a marking where no transition is enabled, is identically reflected by a deadlock in the state graph of UPPAAL verifier where all the transition automata stay in the Disabled location (Fig. 10). All of this can be observed by telling the verifier to generate a diagnostic trace which leads to the deadlock situation. There is another cause of deadlock, though, for a translated TimePN model which is related to the model being unbounded, a property which cannot be known in advance and which was shown to be undecidable for TimePNs in the general case. The translation process proposed in this paper uses knowledge arising from preliminary simulation to estimate the boundedness degree  $k$  of the model. Accordingly, the translator generates a bounded declaration for the marking vector like this:  $\text{int } [0,k] \text{ M}=\{\dots\}$ . In latest versions of UPPAAL, when a limited integer variable is in the position of being assigned a value out of its admitted range, the model checker generates again a deadlock. Here too the modeller can understand the specific cause of deadlock by consulting a diagnostic trace.

Temporal properties (bounded liveness) of each sensor instance was separately checked by the queries:

$A[] M[\text{Input\_i\_j\_Sensor\_ready}] \leq 1$

$i=0, \dots, 3, j=0$ , which asks if the marking of the *ready* place of each *Sensor* instance is always less than or equal to 1, meaning that never a new period is commenced before the activities of latest period are completed. All these queries were found satisfied.

Verification of converters and robot deadlines was accomplished by *model decoration* (Lindhal *et al.*, 2000). For example, to check a converter deadline, a global clock  $z$  was introduced which is reset at each firing of the *compute* transition (see Fig. 4) and analyzed at subsequent firing of the *ack* transition. Model decoration was simply achieved by adapting the *deposit()* function so that when the transition ID corresponds to that of the *compute* transition, clock  $z$  is reset. The following queries were separately launched:

$A[] \text{tInit\_i\_j\_Converter\_ack.Withdraw imply } z < 1$

$i=0, \dots, 3, j=0$ . Each query verifies if at each complete firing of *Converter ack* transition, clock  $z$  is always found less than or equal to 1 (as suggested by preliminary simulation study). The queries were found

satisfied. Similarly, for the robot deadline the *deposit()* function was adjusted so as to reset clock  $z$  each time the *Robot receive* transition completes its firing, and the following query was issued:

$A[] \text{tRobot\_receive.Withdraw imply } (z \leq 10 \ \&\& \ z \geq 6)$

This query verifies that at each firing of *Robot receive*, clock  $z$  is always in the interval  $[6,10]$  proposed by simulation. The query was found satisfied. However, the query was found not satisfied e.g. in the cases  $(z \leq 9 \ \&\& \ z \geq 6)$  and  $(z \leq 10 \ \&\& \ z \geq 7)$ . In other terms, effectively 6 is the upper bound of best case inter-*receive* time of robot, and 10 is the lower bound of worst case inter-*receive* time of robot.

Model checking the robot model demonstrated that the system *is* temporally correct. In addition, verification exactly confirmed the indications provided by the simulation study. Model checking activities were carried out on a Windows XP platform (Pentium IV, 3.4GHz).

## ANALYSIS OF OTHER SCENARIOS

The robot system was also analysed according to other configurations, when the hypothesis of allocating components to distinct physical processors no longer holds. Alternative system configurations, though, were found temporally unfeasible except for the case the four converters are split into two pairs with each converter within a same pair which runs to completion before the partner converter can engage in synchronization with its corresponding sensor. In this scenario, the worst case time a sensor waits for a communication with its converter was found (both in simulation and model checking) to be 4 time units, the worst case time a converter waits for a communication with its associated buffered channel remains 1 time unit, and the time window of the inter-*receive* time of the robot changes from  $[6,10]$  to  $[6,11]$ .

## CONCLUSIONS

The approach described in this paper is currently being extended in the following directions:

- experimenting with timed Petri nets (Ramchandani, 1974) and the *pre-selection* policy. Under *pre-selection*, conflicts are resolved earlier during the enabling process by a non deterministic choice of the transition to fire. A chosen transition fires in three stages: in the first one (*start-firing*) tokens are immediately withdrawn from input places; in the second step (*firing-in-progress*) tokens are “frozen” and made unavailable to other transitions; in the third step (*end-firing*) new tokens are generated in the output places. TPN DESIGNER already supports the *pre-selection* policy for simulation of TimePN models. Model checking activities rest on a

straightforward adaptation of the Transition template in Fig. 10

- enabling the analysis of very large systems by distributed simulation techniques (Beraldi *et al.*, 2003). A major goal is a systematic exploitation of the temporal uncertainty which comes with a TimePN model through transition time intervals, so as to speed-up the simulator performance (Cicirelli *et al.*, 2005)
- enhancing TPN DESIGNER and the translation into UPPAAL by supporting also Pre-emptive Time Petri Nets (Bucci *et al.*, 2004)(Furfaro *et al.*, 2004) which permit modelling and analysis of real-time tasking sets under application dependent control structures (mutual exclusion on shared data, message passing and so forth) and common practiced priority-based scheduling strategies.

## REFERENCES

- Alur R. and D.L. Dill. A theory of timed automata. *Theoretical Computer Science*, **126**(2), 1994, 183-235.
- Alur R., C. Courcoubetis & D.L. Dill. Model checking in dense real-time. *Information and Computation*, **104**(1), 1993, pp. 2-34.
- Behrmann G., A. David and K.G. Larsen. A tutorial on UPPAAL. In *Formal Methods for the Design of Real-Time Systems*, M. Bernardo and F. Corradini Eds., *LNCS 3185*, Springer, 2004, pp. 200-236.
- Beraldi R., L. Nigro and A. Orlando. Temporal Uncertainty Time Warp: An implementation based on Java and ActorFoundry. *Simulation-Transactions of the Society for Modelling and Simulation International*, **79**(10), 2003, pp. 581-597.
- Bucci G., A. Fedeli, L. Sassoli and E. Vicario. Timed state space analysis of real-time pre-emptive systems. *IEEE Transactions on Software Engineering*, **30**(2), 2004, pp. 97-111.
- Carullo L., A. Furfaro, L. Nigro and F. Pupo. Modelling and simulation of complex systems using TPN DESIGNER, *Simulation Modelling Practice and Theory*, **11**, 2003, pp. 503-532.
- Cassez F. and O.-H. Roux. From time Petri nets to timed automata. *Fourth International Workshop on Automated Verification of Critical Systems (AVoCS'04)*, Electronic Notes in Theoretical Computer Science, Elsevier, 2004.
- Cicirelli F., A. Furfaro and L. Nigro. Exploiting temporal uncertainty in the distributed simulation of Time Petri Nets. *Proc. of SCS 38<sup>th</sup> Annual Simulation Symposium*, 4-6 April, San Diego (CA), USA, pp. 233-240.
- Clarke E.M., O. Grumberg and D.A. Peled. *Model checking* (MIT Press, 2000).
- Furfaro A., L. Nigro and F. Pupo. Model checking pre-emptive tasking sets using Time Petri Nets and UPPAAL. *Proc. of 28<sup>th</sup> IFIP/IFAC Workshop on Real-Time Programming*, 6-8 September, Istanbul (Turkey), 2004.
- Gerber R. and I. Lee. A layered approach to automating the verification of real-time systems. *IEEE Transactions on Software Engineering*, **18**(9), 1992, pp. 768-784.
- Lindhal M., P. Petterson and W. Yi. Formal design and analysis of a gear controller. *Software Tools for Technology Transfer*, **3**(3), 2001, pp. 353-368.
- Marsan M.A., G. Balbo and G. Conte. A class of generalized stochastic Petri nets for the performance evaluation of systems, *ACM Transactions on Computer Systems*, **2**(2), 1984, pp. 93-122.
- Marsan M.A., G. Balbo, G. Chiola and G. Conte. Generalised Stochastic Petri Nets revisited: random switches and priorities, *Proc. of the 2<sup>nd</sup> Int. Workshop on Petri Nets and Performance Models*, IEEE-CS Press, 1987, pp. 44-53.
- Merlin P. and D.J. Farber. Recoverability of communication protocols. *IEEE Trans. Commun.*, **24**(9), 1976, pp. 1036-1043.
- Murata T. Petri nets: properties, analysis and applications. *Proc. of the IEEE*, **77**(4), 1989, 541-580.
- Ramchandani C. Analysis of asynchronous concurrent systems by timed Petri nets. *Technical Report TR-120*, MIT, February 1974.
- UPPAAL on-line, <http://www.uppaal.com>

## AUTHOR BIOGRAPHIES

**FRANCO CICIRELLI** is a computer science PHD student at University of Calabria (Unical), DEIS, making research on agent and service paradigms for the development of distributed systems, parallel simulation, Petri nets, distributed measurement systems. He holds a student membership with ACM.

**ANGELO FURFARO**, PHD, is a computer science assistant prof. at Unical, DEIS, teaching object-oriented programming. His research interests include: multi-agent systems, Petri nets, parallel simulation, verification of time-dependent systems, distributed measurement systems. He is a member of ACM.

**LIBERO NIGRO** is a prof. of computer science at Unical ([www.lis.deis.unical.it/~nigro](http://www.lis.deis.unical.it/~nigro)), DEIS, where he teaches object-oriented programming, software engineering and real-time systems courses. His current research interests include: software engineering of time-dependent and distributed systems, real-time systems, Petri nets, modeling and parallel simulation of complex systems, distributed measurement systems. Prof. Nigro is a member of ACM and IEEE.

**FRANCESCO PUPO**, PHD, is a computer science assistant prof. at Unical, DEIS, teaching introductory programming and computer architecture courses. His research interests include: Petri nets, discrete-event simulation, real-time systems, distributed measurement systems.

# MEMBRANE INITIATED GELSOLIN AMYLOID FORMATION

**Inta Liepina**

Latvian Institute of Organic Synthesis,  
Aizkraukles str. 21,  
Riga, LV1006,  
Latvia  
e-mail: inta@osi.lv

**Cezary Czaplowski**

Faculty of Chemistry, University of Gdansk,  
ul. Sobieskiego 18,  
80-952 Gdansk,  
Poland  
e-mail: czarek@chemik.chem.univ.gda.pl

**Paul A. Janmey**

Institute of Medicine and Engineering,  
University of Pennsylvania,  
Philadelphia, PA 19104,  
USA  
e-mail: janmey@mail.med.upenn.edu

**Adam Liwo**

Faculty of Chemistry, University of Gdansk,  
ul. Sobieskiego 18,  
80-952 Gdansk,  
Poland  
e-mail: adam @chemik.chem.univ.gda.pl

## KEYWORDS

Gelsolin, amyloid, amyloid formation, DMPC membrane, molecular dynamics.

## ABSTRACT

Molecular dynamics method is applied to simulate amyloid formation process.

Mutated gelsolin amyloidogenic fragments 173-243 (G173-243) and 173-202 (G173-202) cause Finnish familial amyloidosis disease (FAF). The mechanism of possible amyloid formation involving gelsolin G173-202 fragments and dimyristoyl-phosphatidylcholine (DMPC) lipid was investigated by means of molecular dynamics (MD). G173-202 fragment adhered to the DMPC membrane in 100 ps, retaining  $\beta$ -sheet structure, suggesting that a lipid membrane could bind G173-202 fragments and serve as a germ for amyloidogenesis.

Afterwards one more amyloidogenic fragment G173-202 was placed in the vicinity of G173-202 adhered to the DMPC membrane. After 700 ps of MD, the newly added amyloidogenic fragment G173-202 was interacting with the membrane attached fragment G173-202, indicating the possible mechanism of membrane initiated amyloidosis.

## INTRODUCTION

Molecular dynamics method is useful computer calculations method to simulate interactions of molecules on a time scale thus simulating physiological and pathophysiological processes. In this work molecular dynamics is applied to simulate gelsolin amyloid formation process. Molecule interactions as

the sum of all atom pair interactions (pair-wise approximation) are described in the same manner as in classic molecular mechanics of empirical force fields (Leach, 1996):

$$E = E_{bond} + E_{angle} + E_{tor} + E_{el} + E_{vdW} + E_{H-bond} \quad (1)$$

where  $E_{bond}$  is the bond-stretching energy,  $E_{angle}$  is the valence-angle-bending energy,  $E_{tor}$  is the torsional energy (which arises because of intrinsic barriers to rotation about the bonds),  $E_{el}$  is the electrostatic energy,  $E_{vdW}$  is the energy of van der Waals interactions, and  $E_{H-bond}$  comprises  $E_{vdW}$  between the protons and electronegative atoms that can form hydrogen bonds with them (it is treated separately, because usually different atom-type-dependent parameters are required than those for non-hydrogen-bonding atom pairs). In the present work the flexible valence geometry force field AMBER was used:

$$E_{bond} = \sum_{bonds} \frac{k_{d_i}}{2} (d_i - d_{ei})^2 \quad (2)$$

where  $d_i$  is the actual length of the  $i$  bond,  $d_{ei}$  is the equilibrium length (the length of a non-strained bond), and  $k_{d_i}$  is the force constant.

$$E_{angle} = \sum_{bond\ angles} \frac{k_{\theta_i}}{2} (\theta_i - \theta_{ei})^2 \quad (3)$$

where  $\theta_i$  is the actual value of the  $i$ th valence angle,  $\theta_{ei}$  is the equilibrium value, and  $k_{\theta_i}$  is the force constant. Electrostatic energy is described as a sum of the interaction of charges localized on atomic nuclei,

$$E_{el} = 332 \sum_{\substack{i < j \\ i, j \text{ nonbonded} \\ \text{and } \theta_i, \theta_j \text{ nonbonded}}} \frac{q_i \cdot q_j}{\epsilon \cdot r_{ij}} \quad (4)$$

The energy of van der Waals interactions is described by the Lennard-Jones potential,

$$E_{vdW} = \sum_{\substack{i < j \\ i, j \text{ not bonded} \\ \text{and not 1,3-nonbonded} \\ \text{and not hydrogen-bonded}}} \frac{A_{ij}}{r_{ij}^{12}} - \frac{B_{ij}}{r_{ij}^6} \quad (5)$$

where  $A_{ij}$  and  $B_{ij}$  are constants depending on the types of interacting atoms.

$$E_{H-bond} = \sum_{\substack{\text{hydrogen} \\ \text{pairs}}} \frac{C_{ij}}{r_{ij}^{12}} - \frac{D_{ij}}{r_{ij}^{10}} \quad (6)$$

The torsional energy,  $E_{tor}$ , is expressed by

$$E_{tor} = \sum_{\substack{\text{torsion} \\ \text{angles}}} \frac{V_{1i}}{2} (1 + \cos\varphi_i) + \frac{V_{2i}}{2} (1 - \cos 2\varphi_i) + \frac{V_{3i}}{2} (1 - \cos 3\varphi_i) \quad (7)$$

In molecular dynamics, successive configurations of the system are generated by integrating Newton's equations. The result is a trajectory that specifies how the positions and velocities of the atoms in the system vary with the time. The trajectory is obtained by solving the Newton's equation: ( $F=ma$ ) for the particle with mass  $m_i$  moving along the coordinate  $x_i$  being driven by force  $F_{xi}$ :

$$\frac{d^2 x_i}{dt^2} = \frac{F_{xi}}{m_i} \quad (14)$$

similarly

$$\frac{d^2 y_i}{dt^2} = \frac{F_{yxi}}{m_i} \quad (15)$$

$$\frac{d^2 z_i}{dt^2} = \frac{F_{zxi}}{m_i} \quad (16)$$

where  $x_i$ ,  $y_i$  and  $z_i$  are the Cartesian coordinates at  $i$ th atom,  $F_{xi}$ ,  $F_{yxi}$ ,  $F_{zxi}$  are the components of the forces acting on that atom and  $m_i$  is the mass of  $i$ th atom.

The forces are derivatives of energy  $E$ , described in (1).

$$F_{xi} = \frac{\partial E}{\partial x_i} \quad (17)$$

The molecule system described by above mentioned method consists of protein molecule fragments - amyloidogenic fragments of gelsolin, and of a membrane bilayer fragment.

Gelsolin is a six-domain dynamic actin-filament binding, severing and capping protein capable of severing and nucleating of the actin cytoskeleton (Kwiatkowski, 1999). By modulating actin network via actin assembly and disassembly gelsolin regulates cell shape and motility (Kwiatkowski, 1999; Sun et al., 1999; Robinson et al., 1999; Laine et al., 1998) in the cytoplasm and cleans blood from stray actin filaments (Lee et al., 1992). In addition to actin regulatory function, gelsolin affects cell growth through the

regulation of apoptosis (Sun et al., 1999). Cellular actin scaffold is continuously reorganized in response to a variety of signals. Apoptosis promotes dismantling the actin cytoskeleton, growth factor stimulation induces actin filament assembly at the plasma membrane, which changes cell shape and regulates cell movement. Gelsolin is a calcium-activated regulator of the actin cytoskeleton (Kwiatkowski, 1999; Robinson et al., 1999). Elevated calcium ion concentration activates actin filament severing and capping activities of gelsolin, which results in shorter actin filaments. Another type of gelsolin activation is performed by caspase-3, which cleaves the  $Ca^{2+}$ -insensitive N-half from the  $Ca^{2+}$ -dependent C-half (Kothakota et al., 1997).  $Ca^{2+}$  opens up gelsolin by inducing conformational changes in the C-half (Kwiatkowski et al., 1989; Patkowski et al., 1990), to expose actin binding sites of the N-half; thus the C-half acts as a regulatory domain to impose calcium regulation on the N-half. Caspase-3 acts cleaves the regulatory C-half, thus giving the N-half calcium uncontrolled actin-severing activity.

Human gelsolin is expressed as an 81 kDa protein in the cytoplasm and an 84 kDa protein in the plasma (Kwiatkowski et al., 1986a,b). The secreted protein differs by a signal sequence required for export and a 25-residue N-terminal extension. Human gelsolin has 87% sequence similarity to horse gelsolin (Koepl & Burtnick, 1998; McLaughlin & Gooch, 1992).

Natural gelsolin exists in two forms – cytoplasmic and plasmic, which are derived from the same gene by alternative splicing (Kwiatkowski et al., 1986a,b). The cytoplasmic or intracellular gelsolin is a 84 kDa protein responsible for the assembly of actin fragments during cytoskeletal formation and disassembly of actin filaments during cytoskeletal rearrangement. The cytoplasmic form of gelsolin modulates the actin cytoskeleton and plays a role in the cell motility and apoptosis (Kwiatkowski, 1999; Azuma et al., 1998; Kothakota et al., 1997). Plasma gelsolin is a scavenger that degrades stray actin filaments, thrown out from apoptotic cells into the blood plasma, and retrieves actin monomers for intracellular use. The cysteines at positions 188 and 201 (of human gelsolin) form a disulfide bond in the plasma gelsolin, whereas the cytoplasmic gelsolin form remains reduced (Wen et al., 1996). The actin severing function enables gelsolin to reduce the viscosity of blood and blood clots (Vasconcellos et al., 1994) or cystic fibrosis sputum (Davoodian et al., 1997).

Gelsolin folding occurs in the cytoplasm, where calcium ion concentration is normally low, and folding produces probably the inactive native protein, which can be activated by an increase in calcium concentration. Folding of gelsolin plasma form could take place in the endoplasmic reticulum, where the free calcium ion concentration is 100-1000 M (Meldolesi & Pozzan, 1998), which should be sufficient to generate the active, relaxed form (Zapun et al., 2000).

Plasma gelsolin is implicated in the familial amyloidosis-Finnish type (FAF), inherited disease. A point mutation in the S2 gelsolin domain causes an

amyloidosis with neurological, ophthalmological and dermatological symptoms. This mutation does not affect the cytoplasmic form, while the plasma form is proteolysed, possibly both in the secretory pathway and in the extracellular medium (Kangas et al., 1996, 1999). Thus in addition to its role of actin binding, gelsolin domain S2 is involved in the pathogenesis of familial amyloidosis-Finnish type caused by inherited mutations D187N and D187Y in domain S2 of gelsolin (Maury et al., 1990). At the genetic level, FAF is caused by a single nucleotide substitution in the genomic DNA sequence of gelsolin where guanine 654 is replaced with adenine or thymine (De la Chapelle et al., 1992a,b). This causes replacement of an aspartate residue Asp<sup>187</sup> with asparagine or tyrosine, which makes gelsolin susceptible to aberrant trypsin-like protease cleavage site between residues Arg<sup>172</sup> – Ala<sup>173</sup> (Kangas et al., 1996). Structural changes unmask an aberrant proteolysis site at Arg<sup>172</sup>-Ala<sup>173</sup> leading to a trypsin-protease sensitive molecule (Kiuru, 1998, Kiuru et al. 1999; Kiuru-Enari et al., 2002). The hydrolysis at the Arg<sup>172</sup>-Ala<sup>173</sup> site results in the formation of a 68 kDa C-terminal fragment, which is further digested at the Met<sup>243</sup> residue, forming an amyloidogenic 8.1 kDa peptide containing residues 173-243 (G173-243) of gelsolin domain S2 (Maury et al., 1997, 1994). The peptides G173-243 or its sequent cleavage fragment Ala<sup>173</sup>-Gly<sup>202</sup>, spontaneously associate into amyloid fibrils. The three-strand peptide Ala<sup>173</sup>-Gly<sup>202</sup> has a greater propensity to form amyloid fibrils than the amyloidogenic fragment G173-243 comprising four  $\beta$ -strands and an  $\alpha$ -helix (Weeds & Maciver, 1993, Burtneck et al., 1997). The structure of the Ala<sup>173</sup>-Gly<sup>202</sup> fragment consists of an antiparallel  $\beta$ -sheet that is stabilized by a disulphide bond, from Cys<sup>188</sup> to Cys<sup>201</sup> (Burtneck et al., 1997). Amyloid fibrils are deposited mainly in the facial regions, in the cornea and cranial nerve. As a result of deposition and accumulation of amyloid fibrils the FAF phenotype is characterized by corneal lattice dystrophy, cranial neuropathy, hyperelastic skin, facial muscle weakness, and renal complications (Kiuru, 1998). Mutated gelsolin has defective actin severing activity (Weeds & Maciver, 1993), but FAF disease phenotype arises from the accumulation of extracellular amyloid tissues rather than the loss of gelsolin functions.

The mechanism of amyloidogenesis is not clear. In our previous work (Liepina et al., 2004) we started to investigate the mechanism of amyloid formation. In this work, using molecular dynamics (MD), we investigate interactions of gelsolin amyloidogenic fragments G173-202 [cleaved from the x-ray structure of gelsolin (1DON)] with dimyristoylphosphatidylcholine (DMPC) membrane bilayers.

## METHODS

Method of molecular dynamics with AMBER force field was used.

Dimyristoylphosphatidylcholine (DMPC) lipid bilayer containing 6 x 6 arrays of DMPC molecules was taken from a previous study (Czaplewski et al, 1999). Mutated gelsolin amyloidogenic fragment 173-202 (G173-202) was cleaved from horse gelsolin crystal structure 1DON.

The G173-202 fragment was placed 9 Å away from a DMPC bilayer at two different starting orientations (Fig. 1a) – start 1, and Fig. 2a) – start2) and subjected to NTP MD simulations at elevated body temperature T=312 K for 150 ps. After 100 ps of MD, the G173-202 fragment at either initial orientation adhered to the DMPC membrane (Fig. 1 b), Fig. 2 b).

Afterwards one more amyloidogenic fragment G173-202 was placed in the vicinity of the G173-202 fragment adhered to the DMPC membrane (Fig. 3 a)), and the new system was subjected to NTP MD simulations at T=312 K for 3744 ps.

Subsequently all systems were placed in a periodic lipid-water box and subjected to MD simulations to investigate of the stability of the structures.

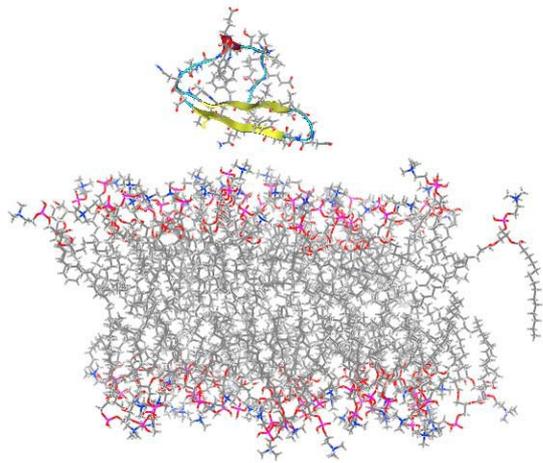
Preliminary calculations on DMPC lipid interactions with amyloid fragments G173-202 were carried out with program MOE, AMBER94 force field. Afterwards the peptide-lipid systems were surrounded by water layer forming periodic lipid-water box, and submitted to AMBER 7.0 molecular dynamics (MD) equilibration to investigate the stability of the systems.

The pictures representing snapshots from molecular dynamics were created by program MOE.

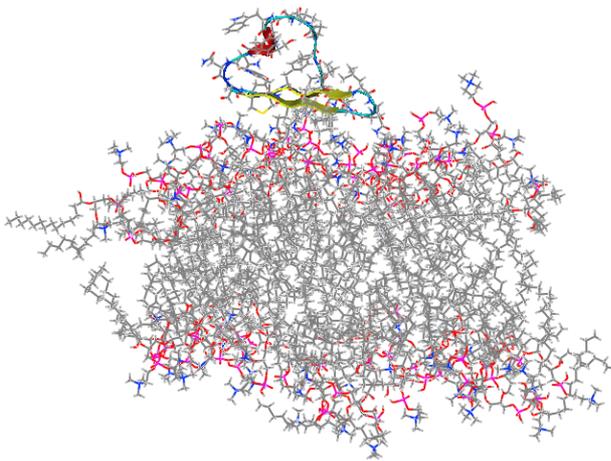
## RESULTS AND DISCUSSION

The gelsolin amyloidogenic fragment G173-202 was placed 9 Å away from a DMPC bilayer at two different starting orientations (Fig. 1 a) – start 1, and Fig. 2 a) – start 2) and subjected to NTP MD simulations for 150 ps at the elevated body temperature T=312 K, which is believed to promote amyloid structure formations. After 100 ps of MD simulations, the G173-202 fragment at either initial orientation adhered to the DMPC membrane (Fig. 1 b), Fig. 2 b), retaining  $\beta$ -sheet structure, suggesting that a lipid membrane could bind G173-202 fragments and serve as a germ for amyloidogenesis (Fig. 1, Fig. 2).

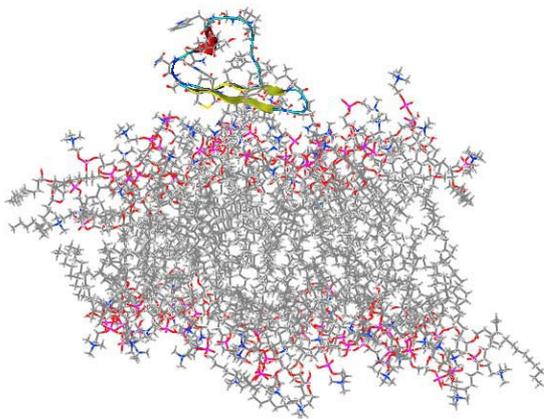
Afterwards one more amyloidogenic fragment G173-202 was placed in the vicinity of G173-202 adhered to the DMPC membrane of the previous system “start-2” (Fig. 3 a)), and the new system was subjected to NTP MD simulations at T=312 K for 3744 ps. After 700 ps of MD, the newly added amyloidogenic fragment G173-202 was interacting with the membrane attached fragment G173-202 and after 1465 ps they were bound with hydrogen bounds, indicating the possible mechanism of membrane initiated amyloidosis (Fig 3). Subsequently all systems were placed in a periodic lipid-water box and subjected to MD simulations to investigate of the stability of the structures.



a) 0 ps

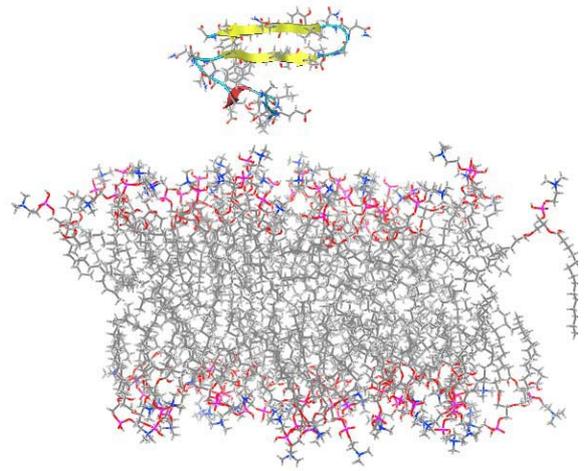


b) 99.5 ps

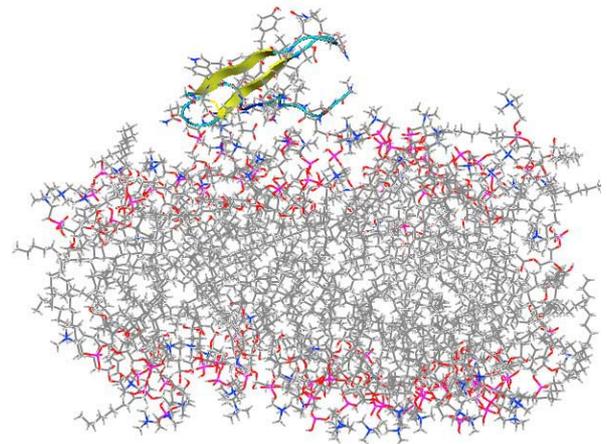


c) 144.5 ps

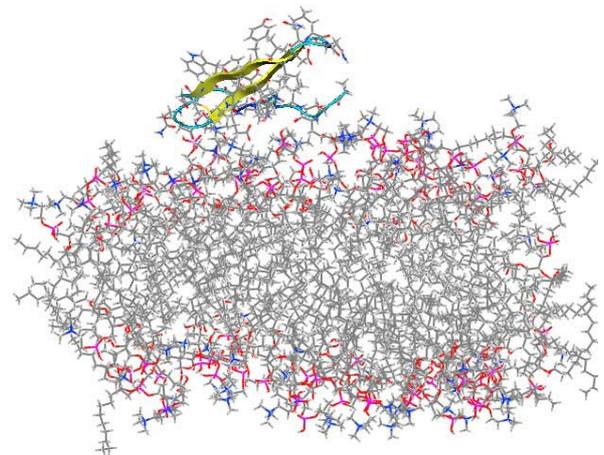
**Fig.1.** Molecular dynamics of the DMPC lipid with G173-202, **start 1**, at a) 0 ps, b) 99.5 ps, c) 144.5 ps.



a) 0 ps

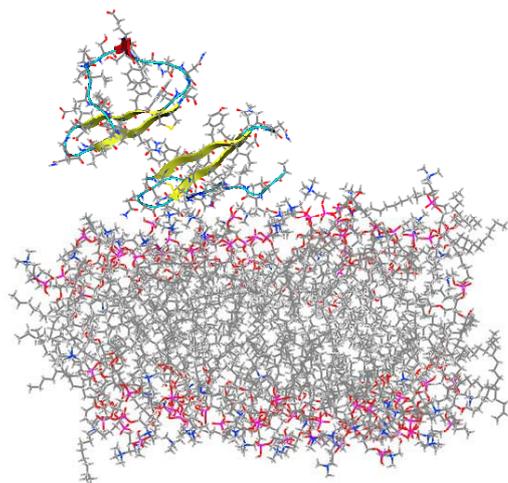


b) 100 ps

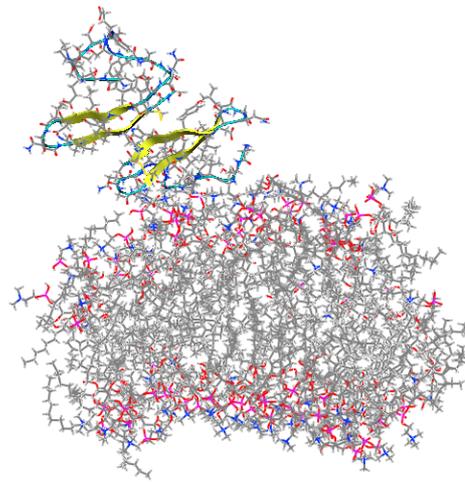


c) 149.5 ps

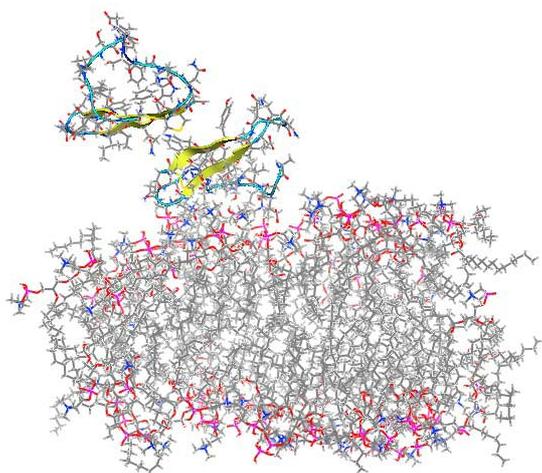
**Fig.2.** Molecular dynamics of the DMPC lipid with G173-202, **start 2**, at a) 0 ps, b) 100 ps, c) 149.5 ps.



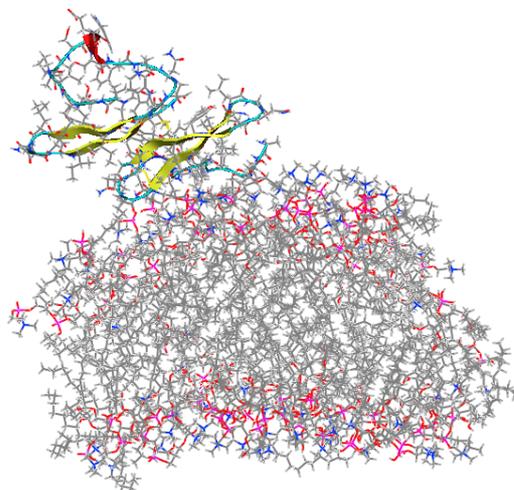
**a) 0 ps**



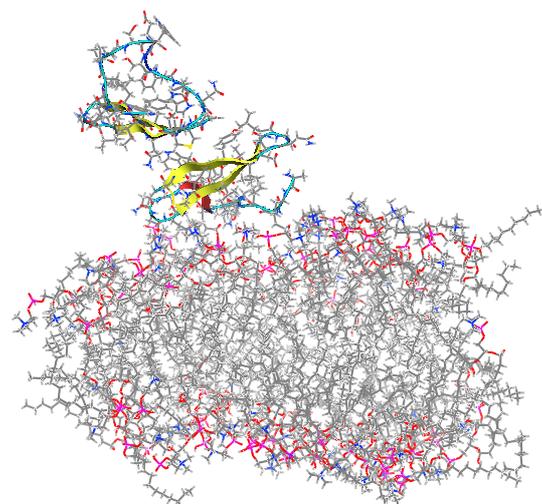
**d) 789 ps**



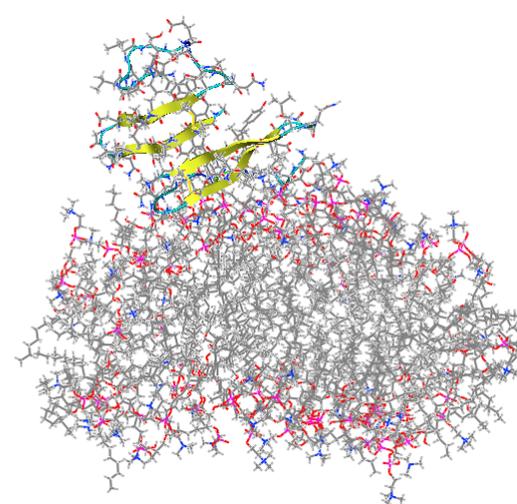
**b) 50 ps**



**e) 1110.5 ps**

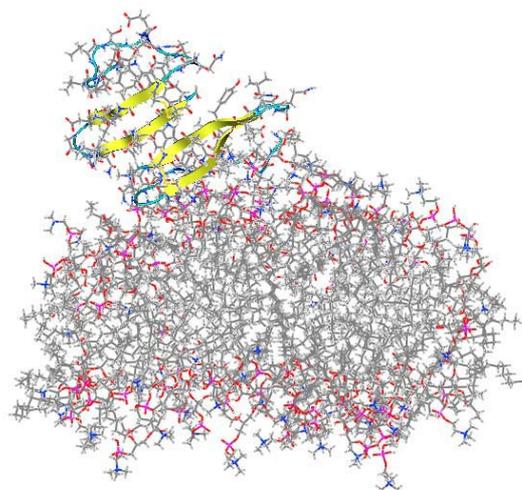


**c) 129 ps**

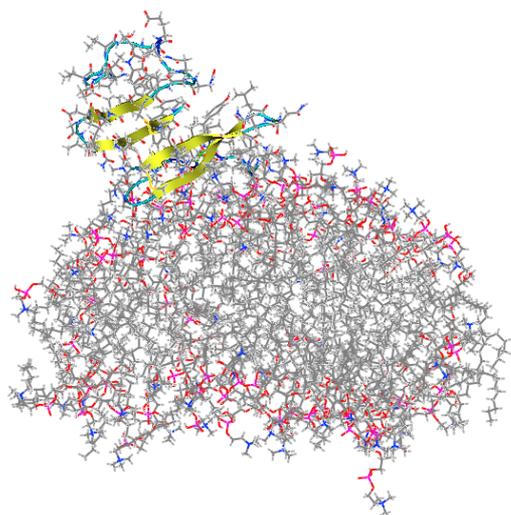


**f) 1978 ps**

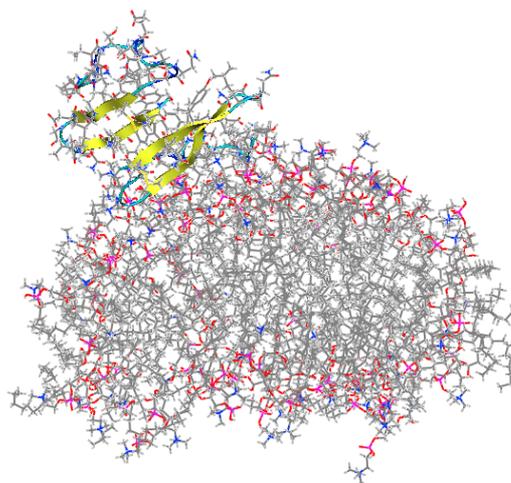
**Fig. 3.** MD of DMPC lipid with two fragments of G173-202, at a) the start, 0 ps, b) 50 ps c) 129 ps, d) 789 ps, e) 1110.5 ps, f) 1978 ps (to be continued).



g) 2492 ps



h) 3017 ps



i) 3744 ps

**Fig. 3. (continued)** MD of DMPC lipid with two fragments of G173-202, at g) 2492 ps, h) 3017 ps, i) 3744 ps

## CONCLUSIONS

Lipid membrane could bind G173-202 fragments and serve as a germ for amyloidogenesis.

Gelsolin fragments G173-202 could stick together forming  $\beta$ -structure.

Results show that such approach of molecular dynamics is reasonable to model the mechanism of amyloid formations.

## REFERENCES

- Azuma, T., Witke, W., Stossel, T. P., Hartwig, J. H. & Kwiatkowski, D. J. Gelsolin is a downstream effector of rac for fibroblast motility. *EMBO J.*, 17, 1362-1370, (1998).
- Burtneck, L.D., Koepf, E.K., Grimes, J., Jones, E.Y., Stuart, D.I., McLaughlin, P.J., Robinson, R.C. The crystal structure of plasma gelsolin V implications for actin severing, capping and nucleation. *Cell* 90, 661-670, (1997).
- Czaplewski, C., Pasenkiewicz-Gierula, M, Ciarkowski, J.. Molecular dynamics of a vasopressin V2 receptor in a phospholipid bilayer membrane. *Journal of Receptor & Signal Transduction Research*, 19, 1999, 355-36
- Davoodian K, Ritchings BW, Ramphal R, Bubb MR. Gelsolin activates DNase I in vitro and cystic fibrosis sputum. *Biochemistry*, 36(32), 9637-9641, (1997).
- De la Chapelle A, Kere J, Sack GH Jr, Tolvanen R, Maury CP. Familial amyloidosis, Finnish type: G654 - a mutation of the gelsolin gene in Finnish families and an unrelated American family. *Genomics*, 13(3), 898-901, (1992a).
- De la Chapelle, A, Tolvanen, R, Boysen, G., Santavy, J. Bleeker-Wagemakers, L, Maury, CP, Kere, J. Gelsolin-derived familial amyloidosis caused by asparagines or tyrosine substitution for aspartic acid residue 187. *Nat. Genet.*, 2, 157-160, (1992b).
- Kangas H, Paunio, T, Kalkkinen, N, Jalanko, A, Peltonene, L. In vitro expression analysis shows that the secretory form of gelsolin is the sole source of amyloid in gelsolin-related amyloidosis. *Hum. Mol. Genet.*, 5, 1237-1243, (1996).
- Kiuru-Enari S, Somer H, Seppalainen AM, Notkola IL, Haltia M. Neuromuscular pathology in hereditary gelsolin amyloidosis. *J Neuropathol Exp Neurol.*, 61, 565-71, (2002).
- Kiuru, S., Oili Salonen, Matti Haltia Gelsolin-related spinal and cerebral amyloid angiopathy *Annals of Neurology*, 45 (3), 305-311, (1999).
- Kiuru, S. Gelsolin-related familial amyloidosis, Finnish type (FAF), and its variants found worldwide. *Amyloid*, 5 (1), 55-66, Review, (1998).
- Koepf, E.K., J. Hewitt, H. Vo, R.T.A. Macgillivray and L.D. Burtneck. *Equus caballus gelsolin.cDNA* sequence and protein structural implications.

- European Journal of Biochemistry, 251, 613-621, (1998).
- Kothakota, S., Azuma, T., Reinhard, C., Klippel, A., Tang, J., Chu, K., McGarry, T.J., Kirschner M.W., Koths, K., Kwiatkowski, D.J., Williams, L.T. Caspase-3-Generated Fragment of Gelsolin: Effector of Morphological Change in Apoptosis. *Science*, 278, 294-298, (1997).
- Kwiatkowski, D.J. Function of gelsolin: motility, signaling, apoptosis, cancer. *Current Opinion in Cell Biology*, 11, 103-108, (1999).
- Kwiatkowski DJ, Janmey PA, Yin HL. Identification of critical functional and regulatory domains in gelsolin. *J Cell Biol.*, 108, 1717-1726, (1989).
- Kwiatkowski D.J., Stossel T.P., Orkin S.H., Mole J.E., Colten H.R., Yin H.L. Plasma and cytoplasmic gelsolins are encoded by a single gene and contain a duplicated actin-binding domain.. *Nature*, 323, 455-458, (1986).
- Kwiatkowski, D. J., Stossel, T. P., Orkin, S. H., Mole, J. E., Colten, H. R. and Yin, H. L. Plasma and cytoplasmic gelsolins are encoded by a single gene and contain a duplicated actin-binding domain. *Nature* 323, 455-458, (1986).
- Laine RO, Phaneuf KL, Cunningham CC, Kwiatkowski D, Azuma T, Southwick FS. Gelsolin, a protein that caps the barbed ends and severs actin filaments, enhances the actin-based motility of *Listeria monocytogenes* in host cells. *Infect Immun.*, 66(8), 3775-3782, (1998).
- Lee, WM, Galbraith, RM. The extracellular actin-scavenger system and actin toxicity. *N. Engl. J. Med.*, 326, 1335-1341, (1992).
- Leach, A.R. *Molecular Modelling. Principles and applications.* Longman. London, (1996).
- Liepina, I., Czaplowski, C, Janmey P. and Liwo A.. Towards gelsolin amyloid formation. *Biopolymers*, 76(6), 543-548 (2004).
- Maury CP, Sletten K, Totty N, Kangas H, Liljeström M. Identification of the circulating amyloid precursor and other gelsolin metabolites in patients with G654A mutation in the gelsolin gene (Finnish familial amyloidosis): pathogenetic and diagnostic implications. *Lab. Invest*, 77, 299-304, (1997).
- Maury CP, Nurmiaho-Lassila EL, Rossi H. Amyloid fibril formation in gelsolin-derived amyloidosis. Definition of the amyloidogenic region and evidence of accelerated amyloid formation of mutant Asn-187 and Tyr-187 gelsolin peptides. *Lab. Invest*, 70, 558-564, (1994).
- Maury, C.P.J., K. Alli and M. Baumann. Finnish hereditary amyloidosis is caused by a single nucleotide substitution in the gelsolin gene *FEBS Lett.*, 260, 85-87, (1990).
- McLaughlin PJ, Gooch J. Crystallization of human gelsolin. *FEBS Lett.*, 302(3), 253-5, (1992).
- Meldolesi, J, Pozzan, T.. The endoplasmic reticulum  $Ca^{2+}$  store: a view from the lumen, *Trends Biochem Sci.*, 23(1), 10-14, Review, (1998).
- Patkowski, A., Seils, J. Hinssen, H, Dorfmüller, T. *Biopolymers*, 20, 427-435, (1990).
- Robinson, R.C., Mejillano, M., Le, V.P., Burtnick, L.D., Yin, H.L., Choe, S. Domain Movement in Gelsolin: A Calcium -Activated Switch. *Science*, 286, 1939-1942, (1999).
- Sun, H.Q., Yamamoto, M., Mejillano, M., Yin, H.L. Gelsolin, a Multifunctional Actin Regulatory Protein, *Journal of Biological Chemistry*, 274, N47, 33179-33182, (1999).
- Vasconcellos CA, Allen, PJ, Wohl, ME, Drazen, JM, Janmey PA, Stossel P. Reduction in viscosity of cystic fibrosis sputum in vitro by gelsolin. *Science*, 263, 969-971, (1994).
- Weeds A, Maciver, S., F-actin-capping proteins. *Curr. Opin. Cell Biol.*, 5, 63-39, (1993).
- Wen, D., Corina, K., Chow, E., Miller, S., Janmey, P., Pepinsky, R. The plasma and cytoplasmic forms of human gelsolin differ in disulfide structure. *Biochemistry*, 35, 9700-9709, (1996).
- Zapun A, Gramatyka S, Deral G, Vernet T. Calcium-dependent conformational stability of modules 1 and 2 of human gelsolin. *Biochem J.*, 350, 873-881, (2000).

## AUTHOR BIOGRAPHIES



**Inta Liepina** was born in Riga, Latvia and went to the Latvian University, where she studied physics and obtained her Master of Science degree in physics. Inta Liepina has worked as a visiting scientist in Friedrich Schiller University of Jena, Germany and in the University of Gdansk, Poland. In 2003 she obtained PhD in theoretical chemistry in the University of Gdansk, Poland. Now Inta Liepina is a researcher at the Latvian Institute of Organic Synthesis, the head of Molecular Biophysics Group. She had calculated conformations of peptides, proteins, lipids, heterocycles and she is interested in molecular modelling of complex systems and protein-ligand, peptide-lipid, protein-protein interactions. [inta@osi.lv](mailto:inta@osi.lv)



**Paul A. Janmey** got his A.B. in Chemistry and Philosophy at the Oberlin College in 1976 and Ph.D. in Physical Chemistry at the University of Wisconsin-Madison in 1982. He worked as an associate professor at the Harvard Medical

School and as a biologist at the Brigham and Women's Hospital from 1985-1999. As a visiting scientist he was at the University of California, Berkeley, as the lecture he was in the Massachusetts Institute of Technology. Now he is a Professor of Physiology at the University of Pennsylvania, Philadelphia, USA and he is the head of Biophysical Lab, dealing on problems of cytoskeleton.

janmey@mail.med.upenn.edu



**Cezary Czaplowski** was born in Gdansk and got his Master of Science degree in the Faculty of Chemistry, University of Gdansk in 1995 and PhD in theoretical chemistry in 1998. After he went for a couple of years to the USA, Cornell University, where he joined Prof. Harold Scheraga's group. Now

he is as an Assistant Professor at the Faculty of Chemistry, University of Gdansk, and his present topic are hydrophobic interactions in protein folding.

czarek@chemik.chem.univ.gda.pl



**Adam Liwo** was born in Gdansk and got his Master of Science degree as well as PhD theoretical chemistry in the Faculty of Chemistry, University of Gdansk. After he went for a couple of years to the USA, Cornell University, where he joined Prof. Harold Scheraga's group. Now

he is a Professor at the Faculty of Chemistry, University of Gdansk, the head of the Molecular Modeling Laboratory, and for the couple of years he is working as visiting scientist in Cornell University again where he is working of protein folding.

adam@chemik.chem.univ.gda.pl

#### ACKNOWLEDGMENTS

This work was supported by NATO Collaborative Linkage Grant LST.CLG.976647, by Fogarty Foundation grant TW006954, by Latvian Science Council Grant 05.1768, by Gdansk Academic Computer Center TASK and by Warsaw ICM Computer Center.

# SUMULATION OF BEHAVIOUR DYNAMICS OF TURBINE DRIVE GENERATING SET

Josko Dvornik\*  
Enco Tireli\*\*  
Ante Munitic\*

\*Faculty of Maritime Studies  
University of Split  
Zrinsko-Frankopanska 38, 21000 Split, Croatia  
e-mail: [josko@pfst.hr](mailto:josko@pfst.hr)

\*\* Faculty of Maritime Studies  
University of Rijeka  
Studentska ulica 2, 51000 Rijeka, Croatia  
e-mail: [tireli@pfri.hr](mailto:tireli@pfri.hr)

## ABSTRACT

The aim of this paper is to show the efficiency of the application of the System Dynamics Simulation Modelling in investigation of behaviour dynamics, one of the complex marine system and process i.e. “steam – turbine generator”. Steam turbine and synchronous generator without contact shall be presented with mental - verbal, structural and mathematical - computing modules, and will simulate working process of ship propulsion complex.

## 1. SYSTEM DYNAMICS SIMULATION MODELS OF THE MARINE STEAM TURBINE

### 1.1. Mathematical model of the Marine Steam Turbine and UNIREG-PID regulator

The steam turbine working process is the conversion of water steam energy to mechanical energy converted to trust on the mechanical units. Therefore, turbine is subjected to various loads transmitted from the units. The steam turbine working system can be derived into two parts: regulating valve and nozzle ring steam space that can accumulate steam energy and rotational part that accumulate kinetic energy. The mathematical model or level equations could be represented as follows:

$$\frac{d\varphi}{dt} = \frac{1}{T_1} (K_1\psi_1 + K_2\psi_2 - \varphi - K_3\alpha) \quad (1)$$

$$\frac{d\psi_1}{dt} = \frac{1}{T_2} (K_0\psi_0 - \psi_1 + K_4\mu) \quad (2)$$

The first differential equation for the first part is defined according to (Siromjatnikov 1983):

$T_1$  - Time constant of rotating parts;  
 $\varphi$  = FI - Relative increment of turbine shaft angular velocity;  
 $\psi_2$  = PSI2 - Relative pressure increment in main condenser;  
 $\alpha$  = ALPHA - Relative turbine load change;  
 $K_1, K_2, K_3$  - Gain coefficients.

The second differential equation is defined:

$T_2$  - Time constant of the steam space;  
 $\psi_1$  = PSI1 - Relative value of the steam pressure increment in the steam space;  
 $\psi_0$  = PSIO - Relative value of the steam pressure increment before regulating valve;  
 $\mu$  = MI - Relative value of regulating valve opening change;  
 $K_0, K_4$  - Gain coefficients.

The PID regulator incorporates in itself proportional (M1), integral (M2) and derivation (M3) regulators. The input function in the regulator is the discrepancy:

Mathematical model of the UNIREG-PID regulator is:

$$UNIREG = PREG + IREG + DREG$$

$$PREG = KPP * X$$

$$IREG = KPI * \int X * dt$$

$$DREG = KPD * \frac{dX}{dt}$$

Where there are:

$UNIREG$  = Output of the Universal-PID regulator,  
 $PREG$  = Proportional regulator,

*IREG* = Integral regulator,  
*DREG* = Derivative regulator,  
*X* = Input Function in the PID regulator,  
*KPP* = Amplification Factor of the Proportional regulator,  
*KPI* = Amplification Factor of the Integral regulator  
*KPD* = Amplification Factor of the Derivative regulator.

In this case, *X*= input function in the first UNIREG-PID regulator is *DISC1*= discrepancy between *CFI*= nominal relative changing of angular velocity and  $\varphi$ = *FI*= relative changing of angular velocity, or exactly:

$$DISC = CFI - FI$$

The *UNIREG*= output of the first Universal-PID regulator is function  $\chi_1 = \chi = KAPA = KAPA1$ = relative shift of high pressure fuel pump.

The UNIREG-PID regulator make connection between angular velocity discrepancy *DISC* and relative shift of high pressure fuel pump variable  $\chi = KAPA$ .

In the reality, this UNIREG-PID regulator self-regulated variable  $\varphi$  to be equal the *CFI*= goal of regulating process of the relative changing of angular velocity  $\varphi = FI$ .

The PID regulator incorporates in itself proportional (*M1*), integral (*M2*) and derivation (*M3*) regulators. The input function in the regulator is the discrepancy-*DISC*.

### 1.2. Structural and Mental-Verbal Models of the Marine Steam Turbine and UNIREG-PID regulator

Fig.1. determinates the Structural Model of Steam Turbine and PID Regulator. It is determined in the accordance with System Dynamics Methodology. Mathematical model could be very suit for determining the mental-verbal qualitative model of the steam turbine and PID regulator.

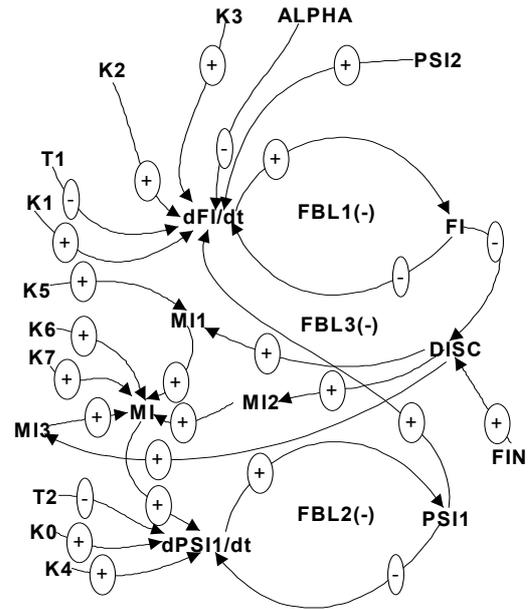


Figure 1. System Dynamics Structural Model of the Marine Steam Turbine and UNIREG-PID regulator

Three self-regulating (-) dominated Feed Back Loops (FBL1, FBL2 and FBL3) are determined in the structural model (Fig. 1.) with a lot of Cause-Consequences Links (CCL).

Mental-Verbal Simulation Model of the FBL1 is:

Link 1. - "If the variable *dFI/dt* (first derivation of *FI* – relative increment of turbine shaft angular velocity, or speed of *FI*), grow up, and the variable *FI* grow up also, then CCL (Cause-Consequences Link) has "positive" (+) dynamics character!";

Link 2. "If the variable *FI* grow up and the variable *dFI/dt* will be drop, then Link 2. will have "minus" (-) dynamics character"! The FBL1 has "minus" (-) global dynamical character, because : sum of negative (-) sign in the FBL1. is odd-number.

In accordance with the System Dynamics (Forrester 1968) quantitative (mathematical) and qualitative (structural) models and POWERSIM-simulation symbols and its program package, it would be possible to work out the System Dynamics Structural Flow Diagram (Fig. 2.)

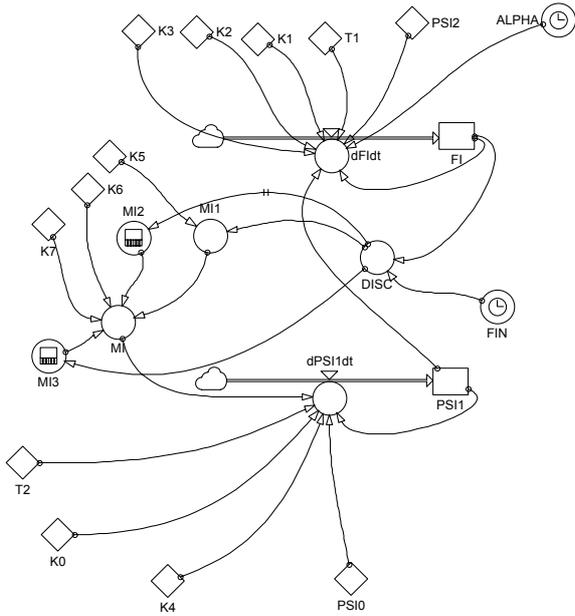


Fig. 2. System Dynamics Structural Flow Diagram of the Steam Turbine and PID Regulator in the PowerSim Symbols

## 2. SYSTEM DYNAMICS SIMULATING MODELLING OF MARINE SYNCHRONOUS GENERATOR

In this short paper, it is impossible to give a complete model; complete model (30 equations) has been presented in IASTED, Pittsburgh, 1998, 372-375, (7).

Where are:

- $\Psi_d = \text{PSID}$  - stator flux linkage in the d-axis,
- $r_s$  - stator resistance
- $x_s$  - stator reactance,
- $\Psi_q = \text{PSIQ}$  - stator flux linkage in the q-axis,
- $\omega = \text{OME}$  - diesel-engine angular velocity (angular frequency),
- $\Psi_{ad} = \text{PSAD}$  - stator mutual flux linkage in the d-axis,
- $u_d$  - stator voltage in the d-axis,
- $\Psi_{aq} = \text{PSAQ}$  - stator mutual flux linkage in the q-axis,
- $u_q$  - stator voltage in the q-axis,
- $u$  - summary stator voltage,
- $\Psi_f = \text{PSIF}$  - rotor exciting flux linkage,
- $r_f$  - rotor exciting resistance,
- $u_f$  - rotor exciting voltage,
- $\Psi_{1d} = \text{PS1D}$  - damping coil flux linkage in the d-axis,
- $r_{1d}$  - damping coil resistance in the d-axis,
- $x_{1d}$  - damping coil reactance in the d-axis,
- $\Psi_{1q} = \text{PS1Q}$  - damping coil flux linkage in the q-axis,
- $r_{1q}$  - damping coil resistance in the q-axis,
- $x_{1q}$  - damping coil reactance in the q-axis,

- $r_L$  - load resistance,
- $x_L$  - load reactance,
- $M_e = \text{MEL}$  - generator electromagnetic moment,
- $i_d$  - stator current in the d-axis,
- $i_q$  - stator current in the q-axis,
- $i_f$  - rotor exciting current and
- $i$  - summary stator current.

System Dynamics Flow Diagram of Synchronous Generator Set with UNIREG-PID regulator shown in Figure 3.

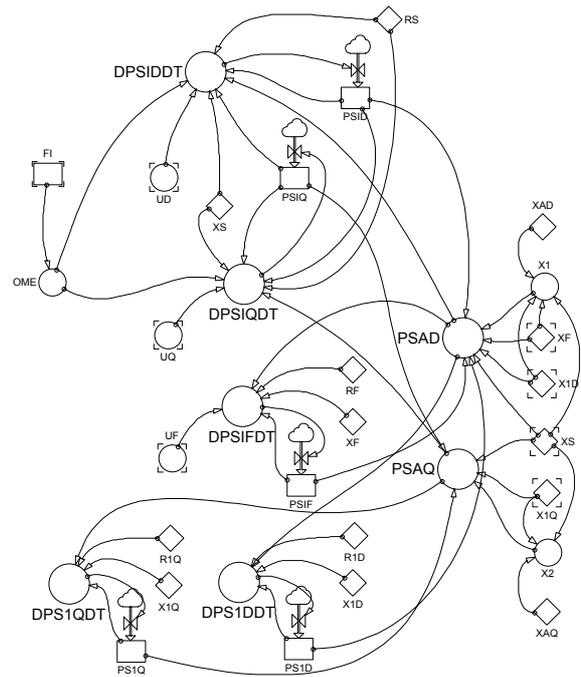


Figure 3. System Dynamics Flow Diagram of Synchronous Generator Set with UNIREG-PID

## 3. SIMULATION SCENARIO OF THE TURBINE DRIVE GENERATING SET

About zero simulation scenario:

The mixed scenario has been implemented in the computer simulation models of the steam turbine and PID regulator:

- steam turbine with PID regulator starts in TIME = 0 and FIN = .05; TIME = 20 and FIN = .05+.45 = .5; and TIME = 40 and FIN = .05+.45+.5 = 1.0 (100%).
- relative turbine load change ALPHA starts in TIME = 60 and ALPHA = .05; TIME = 100 and ALPHA = .05+.45 = .50; and TIME = 140 and ALPHA = .05+.45+.50 = 1.0 (100%)

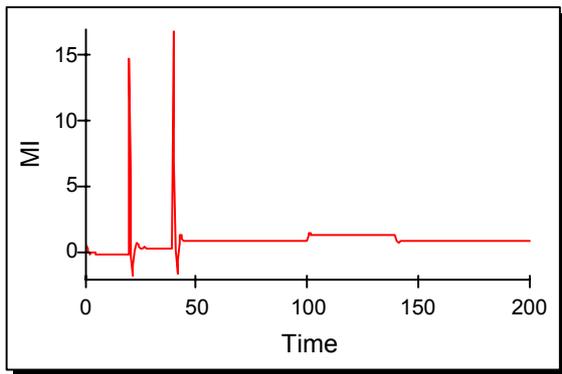
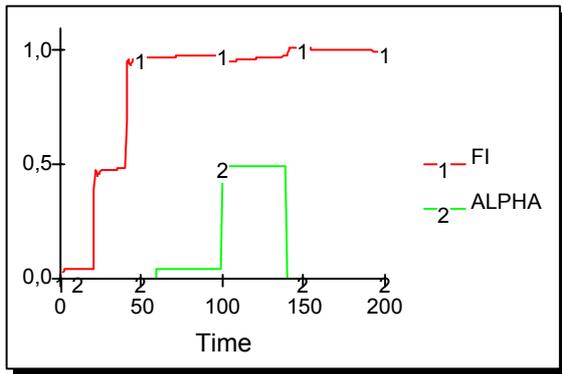


Figure. 4. Graphics Results of Simulation and Heuristics Optimization

#### 4. CONCLUSION

Quality and economical steam turbine functioning depend on many parameters such as steam pressure before and after the regulating valve, condenser pressure, etc. Since successful turbine functioning depends on a large sequence of various parameters, this problem should be solved systematically. By use of the system dynamics in this paper, the complexity of steam turbine dynamics system behaviour has been partially presented. The system dynamics mathematical model, dynamics continued computer simulation model and structural dynamic model of the steam turbine and automatic PID-regulator are presented. Therefore interaction links between each parameter and variables can be analyzed. A simulation model is used to enable optimization of all parameters of the steam turbine system and transient and steady state simulation according to the stated scenario. The most difficult operation conditions can be investigated, even those which in reality are not physically possible.

#### REFERENCES

- Forrester, Jay W. 1973/1971. *Principles of Systems*, MIT Press, Cambridge Massachusetts, USA, 1973.
- Munitic, A., Milic M. and Milikovic M. 1997. System Dynamics Computer Simulation Model of the Marine Diesel-Drive Generation Set, *World Congress on Scientific Computation, Modelling and Applied Mathematics*, Berlin, 1997.
- Munitic, A. 1989. *Computer Simulation with Help of System Dynamics*, BIS, Croatia, 1989.
- Miler, J., 1955, *Parne i plinske turbine*, Zagreb, Croatia, 1955.
- Richardson, George P. and Pugh III Aleksander L. 1981. *Introduction to System Dynamics Modelling with Dynamo*, MIT Press, Cambridge, Massachusetts, USA, 1981.
- He: eB4ha, P. A., 1975, *Abtomat4La\*4β cP\*ob.: ix* *lhepnet4Peck4x Pctahobok*, 1975.,
- Munitic, A., Kuzmanic, I., Krcum, M., 1998., System Dynamic Simulation Modelling of the Marine Synchronous Generator, *Modelling and Simulation, IASTED*, Pittsburgh, 1998, 372-375.
- Munitic, A., et al. 2002. System Dynamics Modelling of Complex Electro Mechanical System, *IASTED, AMS 2002*, Cambridge, USA, 2002, 511-515.
- Munitic, A., Kulenovic, Z., Dvornik, J., 2003., Computing Simulation of Driving System- "Ship/Piston Compressor-Electric motor", *IASTED MS 2003*, California, USA, 2003, 515-520.
- Siromjatnikov, V. F., *Osnovi avtomatiki i kompleksnaja avtomatizacija sudovih paroenerge-tičeskih ustanovok*, Transport, Moskva, (in Russian). 1983.

#### BIOGRAPHY



Josko Dvornik was born 1978. at Split Croatia, where he finished elementary and high Maritime school. He graduated in 2000. year on theme "Application on computer simulation dynamics of behavior of ship propulsion system: windlass – asynchronous engine" at Faculty of Maritime studies, University of Split. Today, he is Master of technical science. He has published over 30 scientific papers on system dynamics simulation modelling. He is a member of the SCS, *The Society for Computer Simulation International*, *IASTED*, *The International Association of Science and Technology for Development and System Dynamics Society*.

# MODELLING RISK MANAGEMENT FOR UNIFIED THREAT MANAGEMENT SYSTEMS

**Vladislavs Minkevics**

Ministry of Finance, Smilšu 1,  
Riga LV 1919, Latvia, Mg.sc.ing,  
Vladislavs.Minkevics@fm.gov.lv

**Jans Slihte**

Ministry of Finance, Smilšu 1,  
Riga LV 1919, Latvia, Dr.sc.ing,  
Jans.Slihte@fm.gov.lv

**Girts Vulfs**

Riga Technical University Kaļķu  
1, Riga, Latvia, Prof,Dr.sc.ing,  
vulfs@itl.rtu.lv

## KEYWORDS

Risk, risk management, automatic risk management, effective risk management, Unified Threat management neuron networks.

## 1. ABSTRACT

**Minkevics Vladislavs, Šlihte Jans, Ģirts Vulfs Riska menedžmenta modelēšana unificētajām draudu apstrādes sistēmām.**

Rakstā pētītas unificētās draudu apstrādes sistēmas, to pielietošanas iespēja informācijas sistēmu riska analīzes nodrošināšanai un piedāvāts risinājums riska menedžmenta efektivitātes uzlabošanai. Lai uzlabotu riska pārvaldību nepieciešamas šādas aktivitātes: pirmkārt, ekspertam, kas novērtē informācijas sistēmu risku jābūt ar lielu pieredzi; galvenā problēma ir definēt, kuri apdraudējumi ir vissvarīgākie, ir jānosaka efektīvi risku mazinājoši līdzekļi. Otrkārt, lai pietiekami operatīvi sekotu apdraudējumiem mūsdienu IT vidē, riska analīzei jābūt nepārtrauktai, bet tā kā parasti to veic eksperts šis process aizņem daudz laika, nereti riska analīze tiek veikta tikai reizi gadā. Šajos apstākļos risku mazināšanai un rezerves nodrošināšanai jau sākotnēji jāiegulda daudz lielāki līdzekļi.

Darbā piedāvāts riska mazināšanas veids, izmantojot unificētās draudu pārvaldības sistēmas, kas balstīts uz automātisku riska analīzi, izmantojot sistēmu ģenerētos audita pierakstus. Apskatīts, kādā veidā, izmantojot standarta audita pierakstus un citas sistēmu atskaites var izveidot sistēmu, kas automātiski noteiktu riska pakāpi dažādiem apdraudējumiem un pamatojoties uz apdraudējuma nopietnības pakāpi, pieņemtu attiecīgu lēmumu par korigējošām darbībām. Riska apgabala noteikšanai var izmantot dažādas metodes, viena no tām ir neironu tīklu modeļi, ar kuru palīdzību apdraudējumi tiek klasificēti apgabalos un noteikts kāds apdraudējums kādam riskam pieder. Ņemot vērā visas iespējamās varbūtības un apdraudējumus, sistēma būs spējīga pieņemt lēmumus savlaicīgāk un efektīvāk pasargājot informācijas sistēmas no apdraudējumiem.

**Minkevics Vladislavs, Slihte Jans, Ģirts Vulfs. Modelling risk management for Unified Threat Management systems**

This paper addresses Unified Threat Management systems and ability to use it to analyze information risks. A solution for effective risk management is proposed. To improve risk management two basic

activities are required: first of all expert who is evaluating information risk should be very competent and experienced. Secondly, to follow threats in today's IT environment, risk analysis should be continuous, but as the matter of fact that usually it is done manually by an expert, this process is time consuming and usually risk analysis is performed once a year. In such circumstances, to minimize risk and to provide information protection and backup more funds are required.

A risk minimization method using unified threat management system which provides automatic risk management based on systems generated audit logs is provided. A review is shown on which, using system's audit log files and other system's reports, it is possible to build a system which is able to automatically evaluate risk for different vulnerabilities and according to risk level, make a decision to minimize risk. There are many methods to define risk area, one of them is neuron net's models, by using which, vulnerabilities are classified into areas and system determines which vulnerability belongs to which risk.

If all possible probabilities and vulnerabilities are put in count, system will be able to make right decisions very quickly, which will more effectively save information systems from threats.

## 2. INTRODUCTION

There are many ways to protect information assets. One of the most important activities is to perform comprehensive risk analysis and to define effective risk mitigation methods. Effective risk mitigation requires expert who is performing risk analysis to be very competent. Sometimes there is not enough information for an expert to evaluate one or another risk. Therefore it may be a good method to perform double risk analysis – one by the expert and another by the risk management system. Risk management system can use system's audit logs and based on pre defined risk descriptions, make a decision if it is an information risk or no.

### 3. PROBLEM

To create an automatic risk management system which would be able to analyze systems alarms and log files and evaluate risks, there are three main activities required:

1. system should have as much defined risk descriptions as possible;
2. system should have a defined action if there is no such defined risk in systems database, for example it may be able to teach itself;
3. decision should be made according to faults and log files.

Automatic risk management system should answer the main risk management questions and immediately inform responsible authorities if any risk exceeds the predefined value.

### 4. THEORETICAL ASPECTS

#### 4.1. Risk and Automatic risk management systems

Risk is a function of the consequences (or impact) of an undesirable event and the likelihood of that event occurring. Risk assessment is the process whereby risk relationships are analyzed, and an estimate of the risk of asset compromise is developed. Compromise includes unauthorized disclosure, destruction, removal, modification, or interruption. Options for managing risk include reduction, transfer, avoidance, and acceptance. A risk assessment produces an estimate of the risk to an IT system at a given point in time. It answers the following questions:

- What can go wrong?
- How bad could it be?
- How likely is it to occur? [1]

Risk management is the process of identifying exposure risks, defining controls and requirements to manage risks, and implementing controls in a cost-effective manner. Ideally risk management should answer those questions:

- What is the risk level of each application?
- How can critical vulnerabilities be found and mitigated?
- How do infrastructure changes impact security levels?
- What is the right priority of remediation actions?

Automatic risk management system is network based system which is capable to analyze traffic, audit logs, alarms from the systems in network and by using mathematical models, make online risk assessment.

ADE risk management system:

ADE is one of the risk management systems that provide a bespoke and individual solution, which has initially been used for clients in the pharmaceutical market where good decision-making is vital but often has to be conducted against a background of incomplete information, assumptions and uncertainties. ADE combines the company's mathematical experience, skills and techniques with clients' culture and

infrastructure for efficient implementation of tailored solutions.

ADE can be designed and tailored for any situation, in any market place where portfolio management of several projects is required and where risk is associated with the projects. Solutions have been successfully implemented in a number of different companies and the organization is currently in discussions with more potential clients.

Using innovative statistical and rigorous mathematical techniques, ADE provides an environment in which to collect, process and present data through creative outputs. All this greatly helps improve client efficiency in development strategies, risk analysis and portfolio management.

ADE allows the client to make critical decisions with complete visibility of the impact of risks and uncertainties as a result of:

- Clearly mapping out different options in a structured way
- Identifying strategies which maximise financial return
- Assessing and balancing risk and return
- Focussing on high level issues
- Having confidence in the mathematical analysis [2]

Risk view system:

RISK-VIEW is another known risk management tool for the process industry, developed and patented by 7-Technologies. Knowledge of past, present and expected events forms the basis for RISK-VIEW predictions of how your process will perform in the future. At the same time RISK-VIEW will give you advice on how to solve potential problems arising tomorrow, in a week or even a month from today. RISK-VIEW can predict and inform, on the basis of information from the pumping stations, when the extra load will reach the plant. RISK-VIEW will warn the operator in advance of how to prepare for the situation. RISK-VIEW's capability of predicting future problems is obtained by combining a traditional fault tree analysis with data that are already fully available in your company.

The traditional fault tree analysis is a static method for calculating the consequence of certain events. The analysis provides an overview of complex systems and defines the combinations of faults leading to an unwanted consequence.[3]

#### 4.2. Unified Threat Management systems

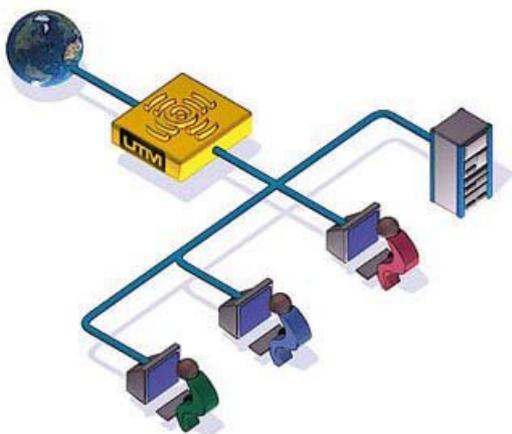
Unified Threat Management is the name for an emerging trend in the firewall appliance security market. You might say that Unified Threat Management is a system that performs content filtering, spam filtering, intrusion detection and anti virus duties that traditionally are handled by multiple systems.

When hackers were the primary focus of the IT Enterprise, a good solid firewall was sufficient to

protect a network. Then as viruses became more prevalent we installed Anti Virus Gateways that scanned for viruses and soon web content filtering and then spam filtering. This resulted in a mess of systems that were costly to administer and consumed valuable rack space.

As the hardware that powered today's enterprise firewalls became more robust it became viable to add these off box functions right into the firewall. Firewalls became "Firewall Appliances". This is where Unified Threat Management comes in. Rather than administer multiple systems that handle Anti Virus, Content Filtering, Intrusion Detection and Spam Filtering, companies can purchase a Unified Threat Management Firewall Appliance that integrates all of the above into a single rack mountable network appliance. The greater functionality that the Unified Threat Management Firewall Appliance provides can be the justification for the replacement of older more basic Firewalls in favor of a Unified Threat Management firewall appliance that does it all. [4]

For instance UTM intrusion prevention system uses over 2500 rules and signatures to identify attacks. The system actively intervenes in the data stream and blocks attacks before they can infiltrate the network. A special Auto-Prevention function simplifies configuration and thereby enables rules and rule groups to be quickly adapted to different security needs in the protected systems. UTM Vulnerability Scanner specifically checks protected systems for vulnerabilities. PacketAlarm continuously runs tests and lists the vulnerabilities it finds. In addition to being well structured, these lists present detailed information on any vulnerabilities found and recommend how they can be removed. [5]



Picture 1 „UTM’s location in network”

The emerging Unified Threat Management Security Appliance market transforms single function appliances into a more flexible environment for deploying multiple security features on a single platform. Unified Threat Management systems are quickly gaining popularity

because they offer security application performance, operating cost savings, and capital cost preservation.[6] It is understood that nowadays systems become more integrated and available to communicate between each other. Unified Threat Management systems can provide a valuable information for automatic risk management system, because viruses and intrusions are the main threats that are affecting information systems.

## 5. WHY INFORMATION TECHNOLOGY AUTOMATIC RISK MANAGEMENT SYSTEM

Organizations that are taking care of their information systems, usually are taking risk assessment. Manual risk assessment has some disadvantages:

- this method of risk assessment requires experts to be very competent;
- it is hard to create one list of vulnerabilities to include needs of all systems;
- the risk assessment is based on subjective thoughts of the members of expert group;
- risk assessment is time-consuming procedure, if risk assessment has to be done for 5 information systems, it may take a few days;
- because it is time consuming, usually it is done once a year, which may cause much vulnerability to be unnoticed, before they happen.

To avoid these disadvantages, risk assessment could be divided into two parts:

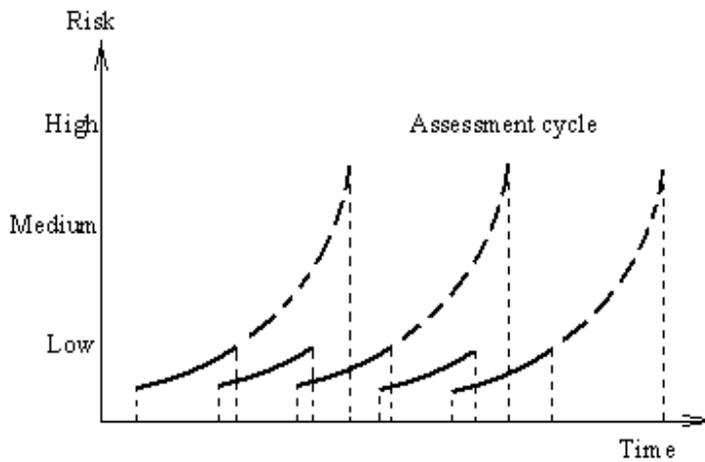
- risk assessment using experts;
- risk assessment done by the system.

Because automatic systems are not able to evaluate all possible risks, it would be very useful to leave manual risk assessment and combine it with reports from automatic risk assessment systems. For instance if Contingency and recovery plans are being evaluated. All aspects should be covered, including:

- back-up practice and policy;
- the contents of the recovery plan;
- the status of the recovery plan;
- the recovery location;
- general contingency practice, procedure and policy;
- network contingency;
- application contingency.[7]

Picture 2 shows that there is direct correlation between the risk assessment cycle time and risk level. The longer the assessment cycle time the more exposed is the organization is to attacks on critical information assets. Therefore, the most straightforward way to reduce risk is by completing the assessment cycle much faster . It will shrink the window of exposure.

By using automatic risk assessment it is possible to shrink exposure to a single day.[8]

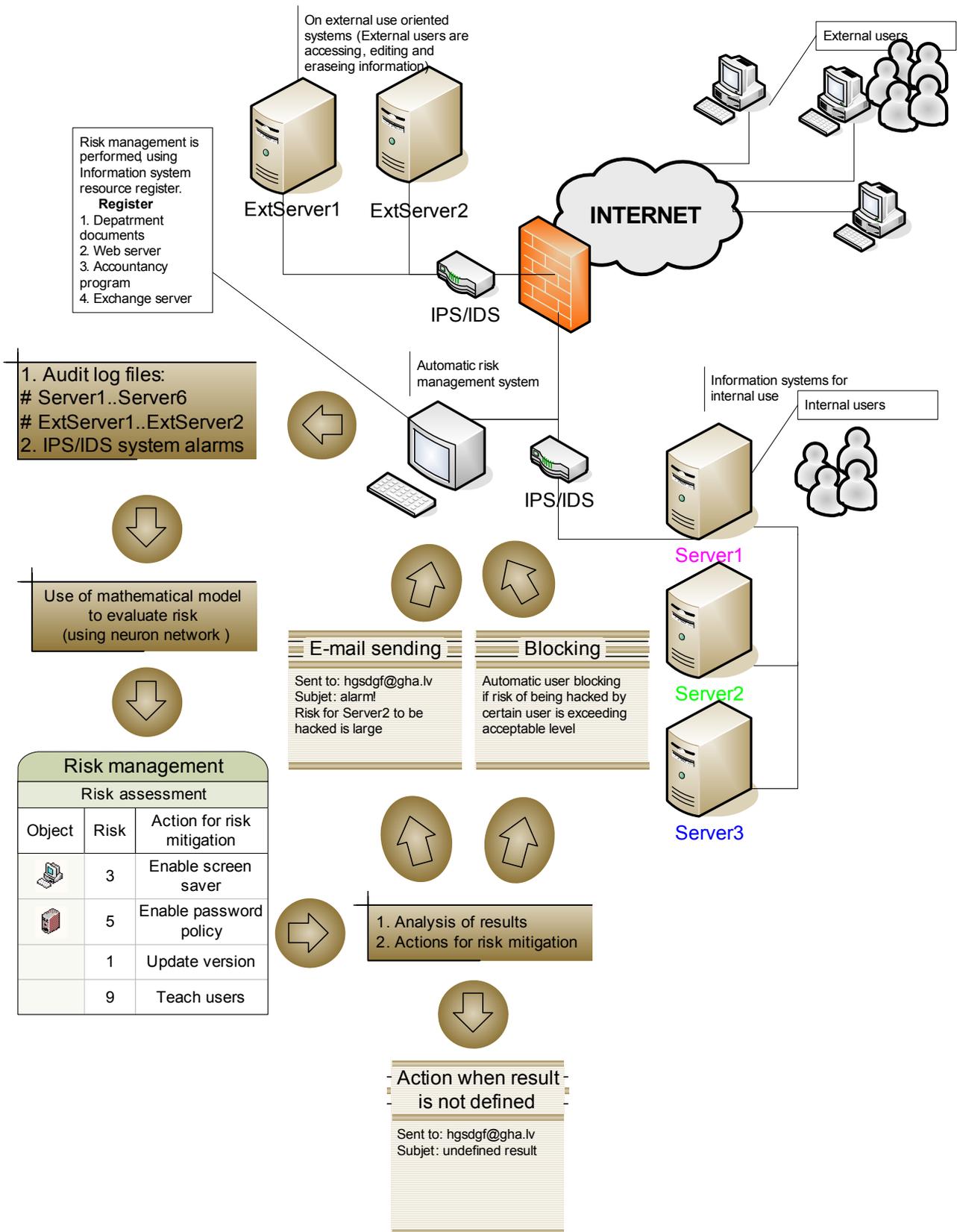


Picture 2 "Risk assessment cycle"

As it was previously mentioned, there are a few automatic risk management systems known but information system risk management system is not available for organizations yet. Information systems automatic risk management system would be a very helpful tool for information system holders and security managers to get a full picture of security risks in organization and to improve security before unwanted security event occur.

On picture 3 information systems automatic risk management system's location in the network is shown. Main parts of the system are mathematical analyzer. One of the options for mathematical analyzer may be neuron network, such as Delta learning rule and decision making algorithm. The main advantage of delta learning rule is that expert who is creating all rules for automatic risk management system may define main properties of each vulnerability and system will automatically sort them into classes from which we can gain risk level of one or another threat.

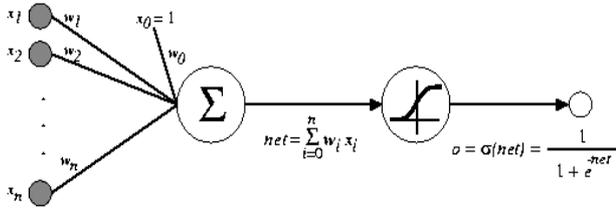
## 6. MODEL OF AUTOMATIC INFORMATION TECHNOLOGY RISK ASSESSMENT SYSTEM.



Picture 3 „Scheme of information systems automatic risk management tool”

## 7. MATHEMATICAL MODEL (NEURON NETWORKS & DELTA LEARNING RULE)

Neuron network seems to be very sufficient solution to sort threats by different criteria's. Delta learning rule seems to be suitable for this task. On picture 4 it is delta learning rule shown.



Picture 4 „Delta learning rule”

learning signal is calculated as follows:  
 $r = [d_i - o_i] f'(net_i)$ , where  $d_i$  is purposed neuron reaction.  
 $o_i$  is real neuron reaction.

$$o_i = f(net_i) = \frac{2}{1 + e^{-\lambda net}} - 1$$

$f'(net_i)$  – result of activation function, which is calculated for  $net = w_i^t x$

Algorithm will stop when error

$$E = \sum_{i=1}^I E_i = \sum_{i=1}^I \frac{1}{2} (d_i - o_i)^2$$

had reached an acceptable level.

To minimize the error level, changes in weights will be made this way:

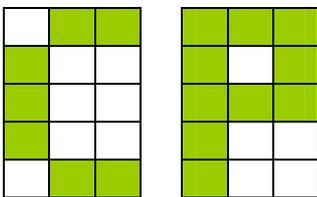
$$\Delta w_i = c(d_i - o_i) f'(net_i) x$$

Where  $c$  is randomly selected constant value.[9][10]

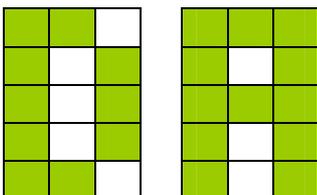
For instance:

There are 2 classes with 2 symbols in each class

1. class



2.class



Symbol classification is shown in Table 2. Using this classification rules will be created.

Table 2 “Symbol classification”

	Y1	Y2	Y3	Y4	Y5	Y6	Y7	Y8	Y9	Y10	Y11	Y12	Y13	Y14	Y15	Y16
C	0	1	1	1	0	0	1	0	0	1	0	0	0	1	1	1
P	1	1	1	1	0	1	1	1	1	1	0	0	1	0	0	1
D	1	1	0	1	0	1	1	0	1	1	0	1	1	1	1	0
A	1	1	1	1	0	1	1	1	1	1	0	1	1	0	1	1

Teacher's answers are:

C	1	1	-1
P	1	-1	1
D	1	-1	1
A	-1	1	1

Beginning weight values are:

$W_1 = (0,2;0,2;0,2;0,2; 0,2;0,2;0,2;0,2; 0,2;0,2;0,2;0,2; 0,2;0,2;0,2;0,2)$

$W_2 = (0,1;0,1;0,1;0,1; 0,1;0,1;0,1;0,1; 0,1;0,1;0,1;0,1; 0,1;0,1;0,1;0,1)$

$C = 1;$

$\lambda = 1;$

$E_{min} = 0,01$

C symbol

$Net_{11} = 1,6$

$o_{11} = (2/(1+\exp(1,6)))-1 = 0,664$

$\Delta w_{11} = \lambda(d_{11} - o_{11}) * 0,5 * (1 - o_{11}^2) = 0,28$

$E = 0,5 * (d_{11} - o_{11}) = 0,5(1 - 0,28) = 0,056$

Iterations will continue for each symbol until Error level reached acceptable level. For our example it is 0,01.

When error level will be 0,01 or less the algorithm will stop. In this case algorithm stops at 26<sup>th</sup> iteration.

As a result we gained weights:

$W_1 = -0.088; 2.639; 0.147; 0.809; 0.300; -0.088; 0.809; -3.046; -0.088; 0.809; 0.300; -2.856; -0.088; -4.155; -2.622; 0.809$

$W_2 = -1.686; -1.860; 1.462; 0.308; 0.200; -1.686; 0.308; 2.871; -1.686; 0.308; 0.200; 1.188; -1.686; -2.363; 4.336; 0.308$

This means that by putting these weights in system, it will recognize the object (which class it belongs to).

## 8. PRACTICAL USE USING SELF LEARNING NEURON NETWORKS

Practically this automatic online risk evaluation will be tested on the system where many different signals are analyzed. The main signal to be tested is audit log files from the server with very important information in it. The most important audit files that should be analyzed online is file server's security audit log files (Table2).

Table 2 “Example of audit log file”

Audit log ID	Description	User	Comments
560	Security: Object Access	Pa-upite	ReadData (D:\DATA_PA\vd_sa n.doc
560	Security: Object Access	En-lipsa	DELETE D:\Users\En\En-lipsa\Twin_Status.xls
540	Security: Logon/Logoff	BJ-vitol	Kerberos; 121.210.15.23
3006	Application: Warning	EvntAgnt	Error reading log event record
532	Security: Logon/Logoff	Ai-vaiva	The specified user account has expired 121.210.15.223
576	Security: Privilege Use	Kj-guran	SeChangeNotifyP rivilege

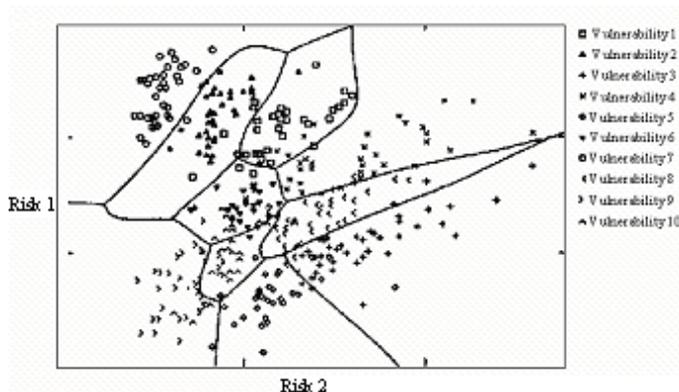
The main problem is to sort important audit logs from not so important ones. Very skilful experts are needed to prioritize logs. When it is done, using advantages of neuron network it should be put in a system in numeric format:

Table 3 “Classes of risk evaluation”

Class	#1	#2	#3	#4	#5	#6	#7	#8
1	1	0	1	0	1	1	0	1
2	0	1	1	1	0	1	0	0
3	0	1	0	0	1	0	1	1

Risk is calculated by dividing possible vulnerabilities into classes. For example in Table 3, there are two classes or two risk definitions: 1) Possibility of being hacked. 2) Password limitation is too strict. As to numbers #1 to #8: they mean different vulnerabilities, like - #1 user tried to enter password three times; #2 very limited access object has been accessed; #3 network traffic is more than 50% for more than 15 minutes; etc.

After using Delta learning rule, we have got weights to recognize an object. By putting these weights into formula  $net = w_i \cdot x$  it is possible to tell which vulnerability belongs to certain risk object.



Picture 5 “Risk identification using neuron networks”

The identification of risks is shown on Picture 5, here, vulnerabilities are defined as small dots on xy axis. Each vulnerability, according to delta learning rule’s defined classification belongs to one or another risk (risk is shown as areas). The main task is to create the correct rule and to classify vulnerabilities according to defined algorithm. Even if vulnerability is not defined, system will classify it by the knowledge it gained before, which means it will be able to classify new vulnerabilities.

## 9. CONCLUSIONS

The proposed system will be able to teach itself and effectively make decisions which vulnerability can be addressed to which increase of risk value.

For a system where this solution is proposed, there are a few problems that might slow the implementation of automatic risk management:

- Not all systems are creating log files (some of them are old),
- Not all audit files are the same format (program or tool is needed to make log files understandable for automatic risk assessment system).

This means that at first, the most important systems should be able to create audit files.

At the moment automatic risk analysis system is in developing phase and there are many things that are not noticed yet, and will be slowing its integration in organizations risk management procedure. If everything goes well, it is possible to get a system that will start really effective risk management in organization and save money by preventing risks rather than fighting with consequences.

## References

1. V.Minkevics and J.Slihte Riga 2004 “Computer science” Science proceedings of Riga Technical university; Series 5; ISSN 1407-7493;
2. St. John’s Innovation Centre Ltd [http://www.ircnet.lu/matching/completetec.cfm?BBS\\_ID=13s843&COMPANY=102381](http://www.ircnet.lu/matching/completetec.cfm?BBS_ID=13s843&COMPANY=102381)
3. Seven technologies <http://www.7t.dk/riskview/download/rv-brochure.pdf>
4. Unified Threat Management Firewall Appliances <http://www.unifiedthreatmanagement.com/>
5. 2003 FORTINET INC <http://www.fortinet.com/news/pr/2004/pr092204.html>
6. VarySys Technologies [http://www.packetalarm.com/packetalarm/index\\_utm.php](http://www.packetalarm.com/packetalarm/index_utm.php)
7. 2003 C & A Security Risk Analysis Group <http://www.security-risk-analysis.com/cobkbs.htm>
8. “Information Systems Control” Volume1 2005 “The Role of Attack Simulation in Automating Security Risk Management” by Gidi Cohen
9. Erik Postima IKAT Universiteit Maastrich [www.cs.unimaas.nl/~postma/DMcourse/DM.ppt](http://www.cs.unimaas.nl/~postma/DMcourse/DM.ppt)
10. <http://www.abo.fi/~rfuller/nfs10.pdf>

# REPRESENTATION OF COMPLEX AGENTS BY FRAMES FOR SIMULATION OF INTERNAL RELATIONSHIPS IN STRUCTURAL MODELLING

Ieva Valkovska  
Janis Grundspenkis  
Department of Systems Theory and Design  
Riga Technical University  
1 Kalku str., LV-1658, Latvia  
E-mail: {ieva, jgrun}@cs.rtu.lv

## KEYWORDS

Agents, Agent-based Simulation, Structural Modelling, Morphological Model, Frame Model, Ontology

## ABSTRACT

This paper presents a novel approach that uses an agent-based modelling to improve the dynamic relations between the objects of modelled system. The set of different frames as the representation of agent's internal structure is considered in the structural modelling domain. Main accent is paid on the usage of the created frame system skeleton that is viewed from two aspects: agent representing a system as a whole and representing its internal structure as multi-agent system.

## INTRODUCTION

During decades of research done in Artificial Intelligence one of the main research objects is knowledge representation. Four categories of the knowledge representation schemes, namely, logical, procedural, network and structural representation schemes have been developed (Luger and Stubblefield 1998). Each of these representation schemes has its own advantages and drawbacks but, in the aggregate, all of them lack dynamics, that is, extensive capabilities to represent dynamic relationships between objects and, as a consequence, they lack advanced techniques to support various kinds of reasoning. It is worth to point out that recently widely used techniques in another field – information systems development – based on various diagrams, like UML or other approaches implemented in CASE tools, as a rule, do not support any reasoning at all. They are only declarative knowledge representation schemes. In fact, the only knowledge representation scheme where the dynamic relationships are simulated is the object-oriented frame based systems. The dynamic relationships between objects are simulated by the message sending or by the use of facets (Durkin 1994). Structural modelling approach emerges in early 70-ies and was targeted towards diagnosis problem solving in complex technical systems (Grundspenkis 1997). In structural modelling two causal models are used, e.g., morphological structure and functional structure. These

models in the knowledge base are represented as a frame hierarchy. Up till now the captured knowledge helps to represent only a static view of the investigated system. Moreover, such static representation allows to support only one of the four kinds of reasoning, namely, structural reasoning (Grundspenkis 2004). Concepts of other three kinds of reasoning that are supported by structural modelling and called behavioural, diagnosis and predictive reasoning (Grundspenkis 2004) have not been implemented yet due to the lack of dynamism of the represented knowledge.

In this paper we present an ongoing research which final goal is to expand reasoning capabilities of the structural modelling. We suggest not only to use facets and message sending for simulation of dynamic relationships but, in addition, to use intelligent agent paradigm to provide dynamic relationships more effectively. Our research is based on the assumption that there should be similarities between knowledge representation in a human agent and that in an intelligent software agent. In other words, we are following Minsky's description of a frame: "When one encounters a new situation (or makes a substantial change in one's view of the present problem) one selects from memory a structure called a *Frame*. This is a remembered framework to be adapted to fit reality by changing details as necessary" (Minsky 1975). So, we are using a set of frames as problem domain knowledge representation in the agent's knowledge base. Interpretation of the whole causal model or its primitives as agents, from our point of view has at least two advantages. First, it provides collaboration between agents and their environment. Second, when internal structure of knowledge is considered, in agent communication a common knowledge base is used by all system's agents that in turn, allows to implement different kinds of reasoning, using one knowledge representation scheme.

## THE BASIC PRINCIPLES OF STRUCTURAL MODELING

An abstract causal domain model built within the framework of structural modelling consists from three models, namely, a model of morphological structure (MSM, in brief) and two kinds of models of functional

structures (FSM). Building of these models is essentially a method for encapsulating domain knowledge into small, independent, composable and decomposable units of knowledge (Grundspenkis 1997). Objects are basic units of the MSM. These primitives have input and output contacts. If interpreted in an application domain, abstract objects correspond to the components of the given system, and contacts represent their inputs and outputs. The connection of one object's output to another object's input is the only path by which the components may interact. Interactions are called flows. Flows are determined and connected to contacts by the Automated Structural MOdelling System (ASMOS). Each contact is characterised by its behaviour state. Behaviour states specify how flows "act" at corresponding inputs and outputs. A MSM is visualised as a diagram or a digraph. The MSM represents a physical structure of a given system, i.e., it represents structural relationships that can be reasoned in logic. The essence of this kind of reasoning (structural reasoning) is the exploration of paths and cycles between objects (Grundspenkis 2004). A frame hierarchy is used to represent all primitives of the MSM into knowledge base. Up to this moment structural modelling helps to create only a static view of a system under investigation. To overcome this drawback we need to encapsulate the procedural knowledge into the knowledge base to support behavioural, diagnosis and predictive reasoning (Grundspenkis 2004) as well. The aim of this study is to integrate the static frame based knowledge representation scheme used in structural modelling using the agent-based approach.

## FRAME MODEL AND ONTOLOGY

Traditionally a frame system (Minsky 1975) is understood as a class – instance relationships, but in our case we are using a set of different frames, which is called a frame model. The frame model is a structure of four types of frames. Each of them is included for different purposes. There are a typical class frame and a procedure, contact and behaviour frames in the suggested structure. The frame model architecture is shown in Figure 1.

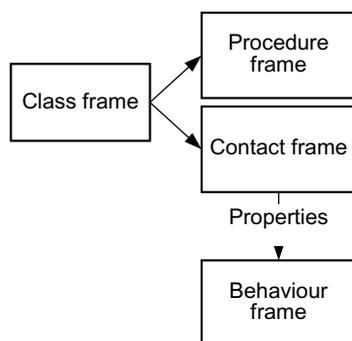


Figure 1: The main elements of the Frame Model Architecture

The special ontology (Gruber 1993) has been designed

to define a common vocabulary of basic concepts and relations among frames for the created structure.

The designed ontology contains descriptions of the frame model components using categories. Categories constitute taxonomy and are described by logical expressions, which define frame structures in a formal way. Due to the scope of this paper the details about the created ontology are not presented. We give only some additional information of the objects in the frame model architecture.

Each class frame model structurally includes frame name/ identifier, supper class frame name/ identifier, properties, contact frame and procedure frame. The descriptions of these elements are as follows:

*Name/ identifier* – shows the meaning of some concepts that represent the existence of identity and is given to the active frame.

*Supper class frame name/ identifier* – corresponding on Name/identifier; allows to find the location of an active frame in the hierarchy. For the first frame (root frame) in the hierarchy of the frame system this object value is given "Haven't". This object value is predefined because the first frame doesn't have any supper class frame.

*Properties* – the attributes that represent qualities and quantities and allow to describe an active frame and the class related with it. There are 3 predetermined attributes for class frame {function, number of contact frames, number of procedure frames} and user defined that allows detailed representation of the information.

Contact and behaviour frames provide active components, actions and/or reactions. These frames support needed dynamic connections and the qualitative and quantitative measures that are described by categories of the created ontology.

*Contact frame* – data structure where information about the active components of specific frame is stored and behaviour of an object is defined. The notion of the behaviour state is introduced that represents effects on the considered active frame or effects provided by the considered active frame on other frames from its environment. These effects are characterised by matter, energy or information flows. Every frame can have more than one contact frames. Contact frame structure includes contact name/identifier, flow name, type {input, output}, connection with another class frame contact and behaviour frame.

Procedure frame structure includes procedure name/identifier, procedure and properties. The procedure frame in our approach makes it possible to separate dynamic information descriptions from static ones that are represented in typical class frame. It stores information about state changes of the frame attributes.

*Procedure frame* – data structure, which consists of stored information about behaviour state changes of the frame in the system and the consequences of given causes. It inspects the rules, which affects an active frame and the frames that depend on active frame, directly operating with data structures – making the

changes in the system.

*Function* – an attribute, which value represent the purpose or goal of an active frame that is received historically or conventionally and can be the base for a whole class. Here the purpose/goal is given which had to be done but without a warranty that it will be reached.

*Number of Contact frames* – an attribute whose value is calculated by counting the contact frames, those defined by the user.

*Number of Procedure frames* – an attribute whose value is calculated by counting of the procedure frames, defined by the user.

*User defined* – an attribute whose value, name and quantity are defined by the user, considering personal or expert given knowledge about each frame and/or class.

*(Further C denotes Contact frame) Name/identifier* – corresponding on Name/Identifier – allows to identify the adherence of contact to the defined frame and to determine the connection with another contact frame.

*(C) Properties* – corresponding on Properties – the properties of contact frame display the possible parameters of contact: flow name, contact type, connections and identification of the behaviour frame.

*Flow name* – attribute of contact frame that defines the names of ingoing or outgoing flows connected with another frame contact. If active frame's contact type is ingoing (IN), then flow connects it with another contact, whose type is outgoing (OUT).

*Type (ingoing or outgoing)* – contact frame attribute that define the type of contact, can be ingoing (IN) or outgoing (OUT). If the flow is going into an active contact then the type is defined as IN, if the flow is going out of the contact then the type is defined as OUT.

*Connection with another class frame contact* – attribute of the contact frame, whose value is specific identifier or name that can be recognized in the problem domain and which points on the contact frame.

*Behaviour frame* – data structure that stores the information about observed behaviour and is realized by the contacts and flows. Behaviour is defined as activity or reaction. The behaviour frame structure includes behaviour frame Name/Identifier, properties and parameter.

*(Further B denotes Behaviour frame) Name/Identifier* – corresponding on Name/Identifier – allows to identify behaviours frame connection with contact frame.

*Parameter* – component, that represents the grade of efficiency in the particular state for the flow and the contact. Parameter structure includes external link, internal link, parameter value and parameter Name/Identifier.

*(B) Properties* – corresponding on Properties – the properties of contact frame consists of one predefined attribute: the quantity of parameters.

*Number of parameters* – Attribute of behaviour frame, which value is calculated by counting user defined number of parameters that is marked as enforceable. It

means that the attribute should be defined or calculated by user's defined function.

*Name/Identifier of the parameter* – corresponding on Name/Identifier – the attribute of parameter, which clearly, completely and briefly shows the meaning of the parameter and connection with the active frame and related flow.

*Parameter value* – attribute of parameter that shows importance, efficiency and quality or quantity of parameter, related with the name of given parameter.

*External link* – defines the relationships between current frame parameter values and another external frame parameter values for the current state. Used to sent or receive resources from the external structures.

*Internal link* – defines the relationships between internal frame parameter values for the current state.

*(Further P denotes Procedure frame) Properties* – corresponding on Properties – at the properties of procedure frame one attribute is located – the quantity of procedures.

*(P) Name/Identifier* - corresponding on Name/Identifier – procedure frame name is given to identify that the upper (lower) level frame procedures are included.

*Procedure* – structure, which shows the activity, steps of the task and instructions, and which is realized by the users defined criteria. Procedure consists of the users selected procedure type, action and description. Procedure do not specify the best solution but executes the user's defined one.

*Number of procedures* – attribute of the procedure frame, which value is calculated by counting users selected number of procedure types, marked as enforceable.

*Type* – the attribute of the procedure, which value defines semantic meaning of enforceable action and in this case allows a certain type of classification. It is expected that there are 4 types:

“If added” – procedure is executed in case when the property or the value of the property is added to the frame. Procedure can affect the properties of an active frame and a subclass frame (name and/or value);

“If needed” – procedure is executed in case when the property or the value of the property need to be added to the frame. Procedure can affect the properties of an active frame and a subclass frame (name and/or value);

“If deleted” – procedure is executed in case when the property or the value of the property is deleted from the frame. Procedure can affect the properties of an active frame and a subclass frame (name and/or value);

“If changed” – procedure includes 2 types of procedures – “If added” + “If deleted”. Procedure is executed in case when the property or the value of the property is deleted and then added to the frame. The procedure can affect the properties of an active frame and a subclass frame (name and/or value);

*Action* – the attribute of the procedure, which shows the reaction on any selected type of procedure in current a frame or in the operating area that shows the work mechanism.

*Description* – the attribute of the procedure, which

describes the actions, meaning, relations and consequences of the procedure.

The detailed frame model architecture is shown in Figure 2.

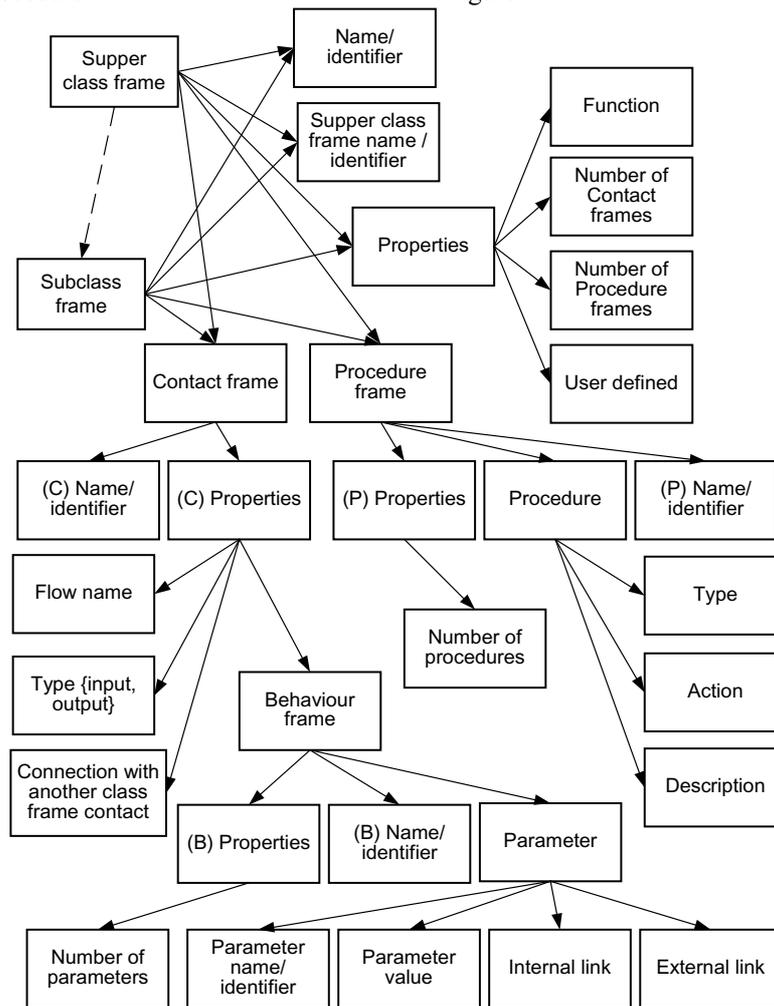


Figure 2: Detailed Frame Model Architecture

Proposed generalisations of the frame system is applicable to a variety of systems that include different types of objects. We suggest to use a frame model as a base for an agent representation. In this case the object of the morphological model is replaced by the created frame model. Replacement makes changes in the frame hierarchy developed in structural modelling.

### FRAME SYSTEM AND INTELLIGENT AGENTS

“An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through effectors.” (Russel and Norvig 2003)

Agents usually are represented without internal structure. We suggest to use frame system as an internal structure of an agent to provide more sophisticated kinds of reasoning.

We assume that it is possible to join several frame models and as a consequence to create a unified system. The result is a frame system skeleton. This skeleton can be considered from two aspects. First, the skeleton can be interpreted as a set of included frame

models and their internal links. In this case a system is comprehended as one unified component of an intelligent agent. Second, the skeleton can be interpreted as a multi-agent system considering each frame model as a separately functioning agent. The first aspect can include the second one and they can be viewed as two different elaboration levels.

From the first viewpoint, the structure of frame models may be viewed as a component in the intelligent agent. Below a special defined standard notation for a component is given, which conditions are determined by the created ontology with its logical definitions and categories. The frames are divided into two groups – the frames that belong to the external and the frames that belong to the internal level. The external level frames interact with the environment and with frames that belong to the internal level. The internal level frames interact only with frames that belong to the internal level. Flows from more than one frame model reach the external environment. Thus the resulting flow is gained from the reaction of the whole network of frames. In this case every frame model has its determined meaning and importance level in a system

as whole. The common understanding of this kind of agent architecture is shown in Figure 3., but decomposition of the frame system skeleton can be viewed in Figure 3 while the internal structure of the frame system skeleton obtained by the decomposition is presented in Figure 4. In point of fact, Figure 4 depicts a multi agent system in which each  $F_i$  is interpreted as an individual intelligent agent represented by the corresponding frame model.

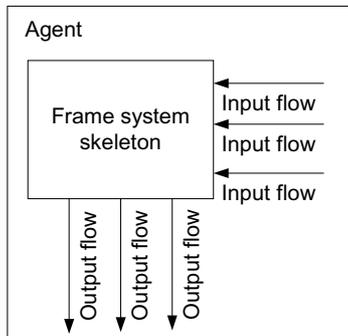


Figure 3: System of Frame Models representing one agent

Each frame model as an individual agent receives several input flows and acts by sending output flows to some other agents according with the functionality of the system under investigation. The frame model that represents the internal structure of an agent's knowledge base is shown in Figure 2. Communications between frames are based on the developed ontology.

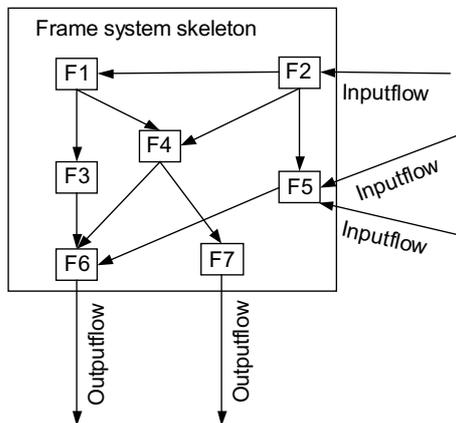


Figure 4: Frame Model representing the agent architecture

### INTERPRETATION OF THE FRAME MODEL AS AN AGENT

Proposed approach provides agents with percepts mater, energy and/ or information and their quantitative and qualitative measurements. Following the idea that agents usually can be viewed according to their type, perceptions, actions, performance measures and they are embedded in different types of the external

environment, in the proposed frame system skeleton one can draw the following parallel:

1. It is possible to define the agent type with class frame. We propose to define the type using one (if looking from the second aspect) ore more frames (from the first aspect side).
2. Perceptions can be represented by input flows described in the contact frame. It is possible to represent also output flows that deliver the information to executive mechanisms. In the first aspect input conditions have been provided by external environment components and common reasoned outputs affect an external environment. Internal processes on the highest level of abstraction can are hidden. In the second aspect input conditions have been provided by external environment components and/ or another agent from multi-agent environment.
3. Actions can be specified with the frame structure that is provided by the procedure frame and in behaviour frame parameters that determine the function results and quality. In the first aspect actions will be specified with several frame models that have specifically similar meaning in a system. In second aspect actions will be specified only in terms of one frame model.
4. Performance measures are provided by the behaviour frame. The created ontology defines the meaning of measures.
5. In the first aspect the experts can play the role of the external environment. In addition to the frame model an extra structure can be provided. It is used to save the identifier of the frame system creator. In the second aspect the external environment are experts and other agents from multi-agent environment. Information about other agents is stored using contact frame attributes.

To demonstrate the first aspect we use the wumpus world example (Russel and Norvig 2003) where agent is specified by a proposed frame system skeleton. The agent lives in some area (4\*4) with pits, wumpus and gold. The goal of the agent is to grab the gold, not to meet the wumpus and not to fall in a pit. It can move in all directions and perform some actions like "Grab" and "Rotate by 90°". The agent can feel the breeze if nearby is the pit, can feel the stench if nearby is the wumpus and can see the glitter if nearby is the gold.

For this example in the frame model architecture the frame name/identifier is "Gold agent", the supper class frame name/identifier is "Logical agent". The function of the Gold agent is "To grab the gold". In this case the number of contact frames and procedure frames is "1", but, in general, there can be any other positive number of frames if needed. The user-defined properties is **step** with value "1", **possible rotation angle\_1** with value "90°", **possible rotation angle\_2** with value "180°", **possible rotation angle\_3** with value "270°", **area** with value "4\*4". The number of

procedures is "1". The procedure name/identifier is "Change the step value". The type of procedure is "If changed" and the action is "IF step>1 THEN step=1. The description of the procedure will be "If some user changes the step value it is changed to previous". The name/identifier of contact frame can be "square [2,2]" that means agent is located in the square [2,2]. The flow name is "breeze" that means agent's sensors have detected the breeze in the square [2,2] and type value is "Input", because agent feels breeze. Connection with another class frame contact can be "pit" to denote that adjacent squares can contain pits. Behaviour frame name/identifier is "move to square [2,3]". The number of parameters can be "1". In this example we suppose that parameter name/identifier is "state costs", parameter value is "-0,4". The internal link is "2,2\_to\_2,2" if agent doesn't move and the external link is 2,2\_to\_2,3 if agent moves to square [2,3]. In this paper we don't give interpretation of the second aspect.

## CONCLUSIONS AND FUTHER WORK

In this paper is discussed an ongoing research that uses an agent – based modelling to improve the dynamic relations between the objects of the modelled system. The combination of different frames as the representation of agent's internal structure is considered in the structural modelling domain. Main accent is put to the usage of the created frame system skeleton that is viewed from two aspects. In both cases, as a consequence, the automatic reasoning is effectively supported. All provided assumptions are only conceptual ones now but according to them we will try to create a real system prototype that will provide possibilities to represent necessary knowledge and to realize simulation of communications by implementation of different kinds of reasoning.

## REFERENCES

- Durkin, J. 1994. „Expert System (Design and Development).” New York, Macmillian Publ. Comp.
- Gruber, T. R. 1993. „A Translation Approach to Portable Ontology Specifications.” *Knowledge Acquisition*, 5, (2), 199-220.
- Grundspenkis, J. 1997. „Structural modelling of complex technical systems in conditions of incomplete information.” *Modern aspects of management science*, No.1, Riga, Latvia, 111-135.
- Grundspenkis, J. 2004. „Reasoning in structural Model – Based Diagnosis.” *Proceedings of 4<sup>th</sup> International conference on Quality, Reliability and Maintenance, QRM 2004*, Oxford, March 21-22, G.J. Mc Nulty (ed.), Professional Engineering Publicity, London, UK, 295-298.
- Luger, G.F. and W. A. Stubblefield. 1998. „Artificial Intelligence. Structures and Strategies for Complex Problem Solving”. Addison Wesley Longman, Harlow, England.
- Minsky, M. 1975. „A framework for representing knowledge.” *The Psychology of Computer Vision*, P. Winston (ed.), New York, McGraw Hill, 211-277.
- Russel, S.J. and P. Norvig. 2003. „Artificial Intelligence: A

Modern Approach.” Prentice Hall, Englewood Cliffs, N.J.

## AUTHOR BIOGRAPHIES

**IEVA VALKOVSKA** is a master student at Riga Technical University. Her scientific interests are associated with system analysis, design and development of the knowledge representation systems and the systems for structural modelling. Contact e-mail address is [ieva@cs.rtu.lv](mailto:ieva@cs.rtu.lv) and contact address is Riga Technical University, 1 Kalku str., LV-1658, Latvia.

**JANIS GRUNSPENKIS** is the professor at Riga Technical University. He also is the dean of the Faculty of Computer Science and Information Technology, the director of the Institute of Applied Computer Systems and the head of the Department of Systems Theory and Design. His research interests are agent technologies, knowledge engineering and management, and structural modelling. He has published more than 150 scientific papers. Contact him Riga Technical University, 1 Kalku str., LV-1658, Latvia; [jgrun@cs.rtu.lv](mailto:jgrun@cs.rtu.lv).

# INTEGRATION OF SIMULATION INTO IT SYSTEMS OF PORT OF GDANSK

Krzysztof Amborski and Andrzej Dzieliński  
Institute of Control and Industrial Electronics  
Warsaw University of Technology,  
00-662 Warszawa, Poland  
E-mail: ambor@isep.pw.edu.pl  
adziel@isep.pw.edu.pl

Jerzy Sukiennik  
Port of Gdansk Authority,  
Gdansk, Poland

E-mail: Jerzy.Sukiennik@portgdansk.pl

## KEYWORDS

Simulation, ITC maritime systems, container terminal.

## ABSTRACT

Management of all activities in seaport is always supported by ICT (Information and Communication Technology). The ICT applications can be different depending on financial and technical possibilities in given location. Port Management and Information System is one of the many ICT applications supporting maritime economy (maritime administration and harbor master offices, shipping industry, ports and their business partners). In the paper the influence of ICT used on various sea port facilities and simulation of port processes with special emphasis on container terminal is analyzed.

The paper deals with the integration of simulation-based method of analysis and improvement of logistics processes in the container terminal at the Port of Gdansk (Poland). The real-life data of the terminal have been fed into the simulator and the obtained model has been validated through comparison of its results with the operational data of the terminal. The model built has been used for further simulations to check the scenarios for future development of the terminal.

## INTRODUCTION

Port of Gdansk is a medium size Baltic seaport with a total cargo handling of over 17 millions tons. Information and communication technology (ICT) applications form important part in Port of Gdansk management and operation infrastructure.

Port Management and Information Systems are some of the many ICT applications supporting maritime economy (maritime administration and harbor master offices, shipping industry, ports and their business partners) in the case of Port of Gdansk. List of ICT applications in broad scope is very long and includes - for example:

- Marine Information & Navigation Systems (including VTS and AIS - automatic identification system);
- Port Community System;
- Port Management Information System (including EDI);
- Port Monitoring & Reporting System (including safety

and security);

- Port Data Communication Network & Broadcasting System;

- Port Terminal Stevedoring Systems;

- Port Customer Systems (Shipping Agents, Forwarding Companies, Customs Agencies etc.)

- Radio Voice Communication and Wireless Data Transmission/Access;

- Simulation systems.

Application of new hand-held devices in port logistics - being in the scope of many on-going activities - is connected directly with information systems used on the port authority level and port stevedoring and port service companies, for example mooring/tugs/towage, pilots etc.

It determines that computerized information systems used on those levels should be analyzed for any integration activity projected in the port area.

There are three main applications in Port of Gdansk interesting from the integration of simulation system point of view:

- DYSPOORT - Vessel/Cargo Information & Clearing System;
- "GTK" - Gdansk Container Terminal System;
- "PGE" - Gdansk General Cargo Terminal System (including Port Free Zone Information System).

Other "internal" PGA (Port of Gdansk Authority) and "external" systems are not included in this paper.

## GDANSK CONTAINER TERMINAL INFORMATION SYSTEM – GTK IS

Gdansk Container Terminal (Dzieliński A. et al. 2002, Dzieliński A. et al. 2003) is one of the main terminals in the port. Its present throughput is about 40000 TEU, but it is still under development as the container traffic is increasing. Moreover, there are plans to build a completely new container terminal with a much higher handling capacity. For the purposes of development of an existing terminal a simulation-based approach seems to be an attractive way of predicting, planning and design (Amborski K. and Dzieliński A. 2002). Therefore, it is interesting to show the interaction of the existing GTK IS with the simulation software used in

the terminal. In this section the main features of the GTK IS are presented.

The existing GTK IS has the following main functions: database supporting terminal management and terminal partners; container handling software system for container pickups, deliveries and customs clearing. It delivers information for terminal management, planning, control and connectivity with the port community. The system provides terminal operator with the simple tools they need to efficiently and cost-effectively manage all areas of a terminal operation, including container inventory, import and export transactions, gate operations, statistical reporting, and Web-based customer service (in the near future).

Main users - partners:

Forwarders:

- notification of containers
- container service status control

Terminal dispatcher's office:

- all documents concerning container operations entry and processing;
- status of containers control: notified containers, IN and OUT operations, storage control etc.
- uses container details like size, weight, vessel, and discharge port to automatically select the most logical storage location for each container. This helps to organize the terminal yard in a way that reduces the need for re-handling;
- almost real time data entry to drive the processes accurately. Decision on where to put a box should be made as it comes in through the gate when all the information is available to make the right decision;
- dispatcher can match containers with owners and consignees, specify container slots, issue sequential work assignments to yard equipment operators and give truckers selected points for pickup and drop-off;
- optimizes time and yard (depot) space;

Port Chief Dispatcher's Office and Terminal Management Requirements:

- automatic data transfer (once a day reporting): loaded and unloaded containers;
- cargo in containers
- statistics & analysis.

Customs Office Requirements - data exchange.

EDIFACT Messages for/from shipping lines: generating & processing

**Future development:**

- transmission of standard electronic messages from various port users - replacing paper documentation currently received and reviewed upon delivery or release of containers at the port gates.
- forwarders, customs agents and trucking companies will send advance notice of transactions, including details of the driver and truck that will handle the transaction, so that

the system will have the necessary information available to process before the truck arrives at the port terminal;

- new handling and ICT technology makes the port or terminal's container volume going up, and up. In one year (last year of being involved in this business) container load went up from few thousand TEU to more than 40,000 TEU;
- new functions of the system for further development.

**GDANSK GENERAL CARGO TERMINAL OPERATIONS – PGE IS**

Other important terminal at Port of Gdansk is the one which handles general cargo - PGE. Main PGE Terminal features are as follows:

- handles every type of non - containerized cargo including general cargo, ro-ro (+ personal vehicles), bulk commodities, steel, grain, dry bulk materials, heavy lift and special vehicles;
- can handle any type of forest products, palletized cargoes, bagged cargoes, and more (has been developed to handle break-bulk and general cargo);
- PGE is a very large (in terms of area covered) terminal with storage and warehousing space for light manufacturing, an active foreign trade free zone etc.

Main functions of the PGE Information System: data base supporting terminal management and terminal partners. The system provides terminal operator with the simple tools they need to efficiently and cost-effectively manage all areas of a terminal operation, including cargo inventory, import and export transactions, gate/rail operations, statistical reporting, and Web-based customer service (in the near future). "PGE-IS" operating system is to coordinate the loading and discharge of cargo with the right workforce, equipment, trucker/rail wagons, warehousing, empty storage/sheds and documentation.

Cargo Information Services will provide EDI and web services on cargo bookings, freight releases, delivery data and import/export information

**Main users - partners:**

Shipping agents:

- notification of vessels and cargo (Import)

Forwarders:

- notification of vessels and cargo (Export)
- cargo loading/unloading status control

Terminal dispatcher's office:

- all documents concerning cargo operations & entry to the port and processing;
- status of cargo control: notified cargo, storage In and Out operations, inventory control etc.
- Dangerous Cargo storage and inventory;
- uses cargo details like size, weight, vessel, and discharge port to select the most logical storage

or vessel location for each cargo. This helps to organize the terminal yard in a way that reduces the need for re-handling;

- almost real time data entry to drive the processes accurately. Decision on where to put cargo or moor the vessel should be made as it comes in through the gate or through the approach channel when all the information is available to make the right decision;
- dispatcher can match cargo with owners and consignees, specify trailers (ro-ro), issue sequential work assignments to yard equipment operators and give truckers or railways dispatcher selected points for pickup and drop-off;
- optimizes time and storage space;

#### Port Chief Dispatcher's Office and Terminal Management Requirements:

- automatic data transfer (once a day reporting): loaded and unloaded cargo;
- dangerous cargo;
- statistics & analysis.

#### Customs Office Requirements - data exchange.

#### XML/EDIFACT Messages for/from shipping agents and forwarders: in future - generating & processing.

#### **Future development:**

- transmission of standard electronic messages from various port users - replacing paper documentation currently received and reviewed upon delivery or release of containers at the port gates.
- forwarders, Customs agents and railways/trucking companies will send advance notice of transactions, including details of the driver and truck that will handle the transaction, so that the system will have the necessary information available to process before the truck/ train arrives at the port terminal;
- new functions of the system for further development:

It is worth mentioning that the simulation software thus far used mainly in GTK can be without too much effort also integrated into the PGE IS environment.

### **GDANSK CONTAINER TERMINAL SIMULATION SYSTEM**

In this section we describe the use of port Process Simulator, to simulate the operation of Gdansk Container Terminal. As mentioned above also operation of other types of terminals can be easily simulated using the same tools and environments. Main activities so far were aimed at simulation-based analysis of the current container terminal operations, identification and evaluation of possible improvement variables and

criteria, and simulation-based improvement of logistics processes (Amborski K. and Dzieliński A. 2002).

The work on simulation of real port operations started with Ulster's Port Process Simulator described in (Dzieliński A. et al. 2002, Dzieliński A. et al. 2003). As it is generic in nature, the simulator needed quite a lot of work in order to customize it for the use at a particular harbor environment, i.e. Port of Gdansk.

Upon completion of customization, the simulation program represented the port operations in general and Gdansk Container Terminal in particular. This way we could come to the main issue of our work, i.e. the simulation of Gdansk Container Terminal operation and simulation-based improvement. However, at first it has been necessary to evaluate the simulator itself in terms of its accuracy in representing the terminal real operations.

One of the issues in question here was whether the simulator can represent real state of Container Terminal operations. To answer this question we assumed a hypothesis that the Port Simulator can accurately describe the operation of the terminal, provided the simulator with real data obtained from container terminal, and made two statistical verification tests: Pearson's and Kolmogorov's test.

#### **Gdansk container terminal – a case study of simulation-based approach**

Container Terminal at Port of Gdansk is a large and complex, living organization. Inside this structure we can see logistic processes at many levels of terminal operation. For us, important processes are located at the highest level – main logistic processes.

Main logistic processes of Gdansk Container Terminal are described at the simple scheme presented on the Fig.1.

In the simulations we have considered five main processes (total number of simulated processes is hard to evaluate, because some of them are composed of several sub-processes making a complete model of simulator):

- import / export cargoes from sea to land side,
- moving cargoes between ships and yard,
- moving cargoes between yard and land vehicles,
- process of ships service,
- process of land vehicle service.

These processes are operating with large number of variables, mostly invisible for simulator user (because these variables base on simulator internal events, not important for the user).

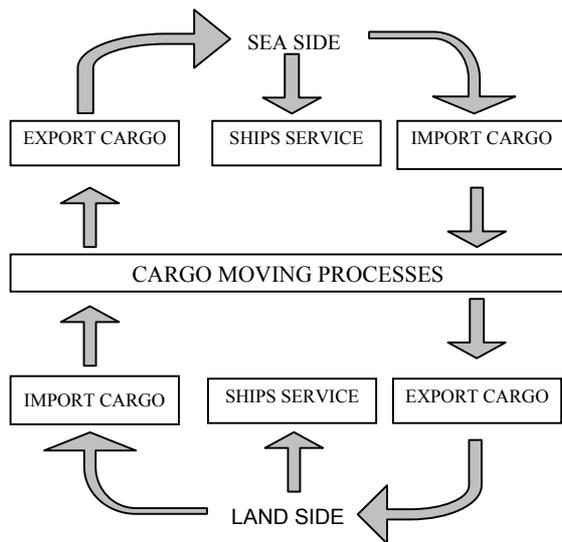


Figure 1. Container terminal logistics processes.

Simulation results were analyzed from the point of view of possible recommendations to GTK management. These recommendations were meant to improve the operations of the terminal with most cost-effective measures.

Process of analysis is achieved using MS Excel application. Output data from simulation (after post processing) are imported to a spreadsheet. In the next step the sheets to visualize change of the variable under consideration are prepared. At the beginning graphical sheets are best suited for general analysis. Trends and dependencies between variables can be easily seen and interpreted. More in-depth analysis can be achieved using numerical capabilities of MS Excel formulae for better accuracy.

In summary, the simulations performed give some guidelines as to how to proceed with port operation improvement. Several factors, like number of berths, number of port basins, number of gantry cranes etc. were taken into account and one may notice the influence of new investments in the port infrastructure, aqua structure and equipment on the operation capabilities. The results obtained are preliminary, but even they show clearly that some investments in the existing terminal equipment and set-up can improve the situation significantly in terms of loading/unloading time, ship queue or number of shipments (Amborski K. 2004).

The partners from Gdansk Container Terminal developed business strategy for GCT. In near future they will plan several investments. We expect that simulator will be also a useful tool to analyze these future investments. GCT Authority is especially interested in following results from simulations:

- Amount of import cargo
- Amount of export cargo
- Vehicle turnaround time
- Number of shipments in
- Total amount of cargo

- Mean number of units of resources allocated
- Number of customers who had to wait

These factors are useful especially from practical point of view. Cargo flow and ships movement is very important for every manager who works at container terminal.

There are two possible investment areas in the near future:

Area one – Quay.

Current length of quay is suitable to serve any type of vessel (in Port of Gdansk). But at one time terminal can serve only one ship. To save customers time and expand overall terminal operability Port's Authority plans to enlarge the length of quay. The simulation confirms that longer quay will be able to serve two ships at one time.

Area two – Handling equipment

Currently in the terminal there are three 'ship to shore' type cranes. This number is enough for current quay and cargo movement. However, as simulations show the increased length of quay force the terminal operator to increase the number of cranes. Two cranes at two berths will be optimal.

Area three – Yard

Higher level of cargo movement determines higher level of yard usage. Since, according to simulation results the current yard area will not be enough it will need expansion. Fortunately this is not a serious problem to expand this area. Terminal Authority reserved suitable land for future expansion of yard.

## CONCLUSIONS

The simulation-based approach to analysis and improvement of logistics processes in the Port of Gdansk has been used. The part of port under consideration was the Gdansk Container Terminal, and the simulations were related to reflect the traffic of containers in the terminal. The simulation software used is a part of information system used by GTK. The Port Process Simulator adopted, after the customization turned out to be a useful tool for assessing and predicting the behavior of the container terminal. After the verification of the simulator using the real data, the tool has been used to simulate the present operation of the existing terminal. These simulation results formed a reference point for further simulation-based considerations. Basing on this reference, several operation scenarios have been prepared and fed to the simulator. Each simulation scenario took into account changes in one of important terminal variables at a time, without changing other parameters. This way the analysis of the simulated scenarios gave the justification to possible changes in organization or new investments plans of the terminal. This made possible to prepare the recommendations to the terminal management on how to improve the operation and logistics of the terminal. This sort of simulation-based approach may also be

used as a design support tool for the new container terminal which is now under preparation in Port of Gdansk (Amborski K. 2004). This terminal with a significantly greater throughput and situated in the Northern Port part of Port of Gdansk will eventually replace the existing one.

## ACKNOWLEDGEMENTS

The work has been partially supported by EU projects: BALTPORTS-IT funded under the 5<sup>th</sup> Framework IST Program and eLOGMAR-m funded under the 6<sup>th</sup> Framework IST Program of the European Commission.

## REFERENCES

Amborski K., Dzieliński A. 2002. "Simulation of a Container Terminal: A General Approach." *Proceedings of 16<sup>th</sup> European Simulation Multiconference on Modelling and Simulation "ESM 2002"*, Darmstadt (Germany), June 3-5, 2002, pp. 385-388

Dzieliński A.; Amborski K.; Kowalczyk P.; Sukiennik J.; Pawłowski A. 2002. "Simulation of Operation of a Medium-sized Seaport. Case Study: Port of Gdansk." *Proceedings of 14<sup>th</sup> European Simulation Symposium ESS'02, Dresden (Germany), October 23-26, 2002*

Dzieliński A., Amborski K., Kowalczyk P., Sukiennik J. 2003. "Simulation-Based Approach to the Analysis of Logistics Processes at Gdansk Container Terminal". In: *Proceedings of HMS-2003 "International Workshop on Harbour, Maritime and Multimodal Logistics Modelling and Simulation"*. Riga (Latvia), September 2003

Amborski K., Dzieliński A. 2003. "Some Remarks on Simulation-Based Analysis of Maritime Container Terminal." 15 Polish-German Seminar "Research and Development in Engineering and Studying in United Europe", Koeln (Germany), 17-18 November 2003.

Amborski K. 2004. "Simulation Supported Optimization of Container Storage Place." *Proceedings of International Conference on Advances in Internet Technologies and Applications, CAITA-2004, Purdue University, West Lafayette, Indiana (USA), July 8-11, 2004, pp.13-18.*

Merkuryev Y., Bardachenko V., Solomennikov A. and Kamperman F. 2003. "Simulation of logistics processes at the Baltic Container Terminal: Model validation and application." *Proceedings of the International Workshop on Harbour, Maritime and Multimodal Logistics Modelling and Simulation. HMS-2003. September 18-20, 2003, Riga, Latvia.*

## AUTHOR BIOGRAPHIES



### KRZYSZTOF AMBORSKI

Born in 1941. Graduated from Warsaw University of Technology (Poland) in 1964 as a M.Sc. in Telecommunications. In 1971 received a M.A. in mathematics from University of Warsaw. He obtained a Ph.D. in Electrical Engineering from Warsaw University of Technology in 1972. In 1998-2004 he was a professor at the University of Applied Sciences in Darmstadt (Germany) and visiting professor at the University of Wisconsin Platteville (USA) teaching simulation and programming. Currently he is back at Warsaw University of Technology teaching control theory, optimization and simulation.



### ANDRZEJ DZIELIŃSKI

Born in 1959. Graduated from Warsaw University of Technology in 1983 as a M.Sc. in Electrical Engineering. Obtained a Ph.D. in Electrical Engineering in 1992 and a D.Sc. in Automatic Control in 2002. Currently he is an associate professor at Warsaw University of Technology. His main professional interests are in computational intelligence, modelling and simulation and automatic control.

**JERZY SUKIENNIK.** Graduated from Academy of Economics in Prague (Czech Republic) as a M.A. in Computer Science. Currently is a Chief IT Officer at Port of Gdansk Authority.



# **Simulation and IT-Based Modelling in Logistics and Transport**



# OPTIMIZATION AND DEVIATION WITH THE TRAVELING SALESMAN PROBLEM IN REVERSE

William Conley  
University of Wisconsin at Green Bay  
Business Administration and Statistics  
2420 Nicolet Drive  
Green Bay, Wisconsin 54311-7001 U.S.A.  
email: conleyw@uwgb.edu

## KEYWORDS

VTSPA statistic, multivariate deviation test, longest routes

## ABSTRACT

Analysis of variance (if normality conditions hold) and the Kruskal Wallis test present ways to test multiple populations for equal averages. Presented here is a new approach to testing multiple populations for equal deviation or variability. Its VTSPA statistic calculations are based on finding the longest routes to connect the data points rather than the well known reverse of that concept, finding the shortest routes to connect points, the so called "traveling salesman problem" or TSP. The mathematically difficult TSP and its reverse can be handled with approximation techniques made viable by the fast computers currently available. Various versions of the traveling salesman problem (TSP) can do much more than route delivery trucks or sales people to customers in an optimal fashion. They can form the basis for the TSP class of statistics that help to analyze large multivariate data sets.

## INTRODUCTION

A researcher, engineer, scientist or business person is frequently confronted with large data sets. When this happens the question that arises initially is, does this data mean anything? Can it be reduced to a useful piece of information? Has production increased? Has quality improved? Are the factors correlated? In fact, an old name for statistics was "data reduction." The idea was to reduce the data to one or two useful conclusions that could be advantageously acted upon.

The CTSP statistic discussed in (Conley 2003) showed how to use an adaptation of the TSP problem with multi stage Monte Carlo optimization (MSMCO) to see if  $k$  factors were correlated using  $n$  data points for the statistical test. Also, the TSP and MSMCO based MTSP statistics in (Conley 2003) presented a general purpose multivariate test for averages. Continuing with this theme, the VTSPA statistic tests  $k$  populations to see if they have equal deviation or variability. However, variability or deviation is concerned with how far apart data points are, so we will use the TSP in reverse and find the longest routes (instead

of the shortest routes) connecting the  $k$  dimensional data points.

Let us look at a five and a nine variable example from business and industry.

## FIVE SALES TEAMS

A supervisor of five sales teams looks at the number of units of product ABXC sold in each of the last fifty-five weeks (Table 1) by the sales teams working for her. She will, of course, study the means and total sales production by each team separately and together. However, the supervisor wonders if some subtle or large differences in deviation or variability in sales production could also be tested for. Therefore, the null hypothesis of equal deviation of the five populations will be tested with the VTSPA=A/B statistic on the data (given below) using the farthest available point algorithm (MSMCO (Conley 1993) and (Conley 1994) could also be used adjusted for maximizing instead of minimizing).

Table 1: Five Sales Teams Data

Week Number	Team 1	Team 2	Team 3	Team 4	Team 5
1	3	27	2	42	31
2	83	49	50	66	95
3	37	40	36	49	50
4	37	27	88	70	15
5	13	48	100	42	57
6	60	21	55	70	39
7	38	28	100	57	6
8	90	38	100	58	40
9	57	35	32	65	32
10	51	45	86	70	8
11	6	24	21	70	84
12	34	50	50	49	26
13	93	52	7	57	31
14	25	27	3	41	95
15	34	37	96	66	95
16	32	39	52	63	39
17	35	24	42	51	17
18	7	30	27	64	69
19	90	46	41	70	29
20	68	24	34	67	46

21	87	42	99	54	75
22	83	49	53	70	95
23	53	39	58	70	21
24	21	47	16	46	70
25	37	33	98	69	43
26	18	41	20	47	50
27	85	30	89	44	12
28	33	33	35	47	21
29	23	50	52	53	78
30	9	50	73	68	23
31	36	40	31	46	43
32	65	40	74	56	27
33	40	38	55	43	91
34	28	51	9	70	57
35	19	34	99	60	77
36	49	39	88	66	67
37	46	29	34	51	91
38	42	48	19	48	70
39	3	28	17	66	80
40	18	31	43	46	34
41	21	21	69	63	78
42	89	49	8	45	23
43	74	21	19	67	82
44	0	52	92	52	14
45	50	31	35	70	18
46	5	24	22	44	82
47	61	47	20	61	74
48	1	44	54	45	21
49	77	32	96	43	33
50	86	48	15	70	28
51	7	31	41	46	60
52	71	34	98	60	90
53	25	47	78	66	60
54	92	46	15	59	77
55	73	22	41	69	23

Briefly, one of the 55 five dimensional points is selected at random. Then, this point's distance to the point furthest away from it (in five dimensional space) of the other 54 points is calculated and added to the A total. Then from this second point, the distance to the point furthest away from it, of the remaining 53 points is calculated and added to the A total. Then from this third point, the point (of the remaining 52 points) furthest away from it is identified and this distance is calculated and added to the A total. This process continues until all 55 points are connected with a total distance of  $A=5010.41$  for the data given here.

Then in the range of the data (0-100 in this case) five sets of 55 points were drawn at random and their longest route distances were calculated to be 6386.74, 6152.23, 6694.64, 6410.67 and 6527.27. Taking the median of these values for B, we get  $VTSPA=A/B=5010.41/6410.67=.7816$ . The  $5 \times 4 = 20$  A/B quotients from the five random sets of data are all greater than  $6152.23/6694.64=.91898$ . Therefore, the null hypothesis of equal deviation can be confidently rejected, because the  $VTSPA=.7816$  did not occur by chance. It happened because the data was much more compact in some of the five dimensions than in the others.

Therefore, the supervisor can look into this for keys to improving or steadying the performances of her sales teams.

Geometrically, the idea is, if for example, data were collected on three factors of equal variability, the graph of these points in three dimensions could be covered by a cardboard box of fairly equal dimensions in each direction (like a cube). However, if three factors were such that one factor's points were in the range of the length of the cube, but the other two were much more compact (less deviation) then their graph could be covered by an object with different dimensions in the three directions, such as a cricket bat. Therefore, the longest route connecting the points covered by a cricket bat shape will be much shorter than the longest route connecting the points covered by the cube. VTSPA exploits this difference to test for deviations.

Let us look at another example.

### NINE INVESTMENT ADVISORS

The world of investments sometimes looks at deviation or variability as a measure of risk. The idea being if one would earn a six percent rate of return, which is guaranteed on an investment, the risk is virtually zero. However, other investments might be estimated to earn six percent, but any amount of actual return is possible. Then the risk is considered greater.

A manager of a large investment house recorded the yield on investments that nine financial advisors (working for him) obtained for each of sixty individual clients (Table 2) that the manager assigned to them. The data is given below in yields of tenths of a percent. The manager wants to know if there is equal or unequal deviation (or risk) between the nine financial advisors working for him. The idea being that a top performer with low risk might be in the future assigned to the most important projects or clients and rewarded more.

Table 2: Nine Investment Advisors Data

	X <sub>1</sub>	X <sub>2</sub>	X <sub>3</sub>	X <sub>4</sub>	X <sub>5</sub>	X <sub>6</sub>	X <sub>7</sub>	X <sub>8</sub>	X <sub>9</sub>
1	21	81	16	63	34	31	44	42	34
2	24	37	93	57	85	62	53	52	13
3	36	24	36	70	40	80	45	51	4
4	40	25	24	67	37	40	30	42	51
5	10	24	96	47	95	65	31	62	74
6	45	32	100	61	49	58	49	45	85
7	89	27	82	67	39	53	31	51	31
8	85	41	85	41	90	75	44	64	56
9	65	27	29	58	42	73	51	55	20
10	88	47	67	48	66	61	40	43	77
11	29	22	79	68	74	49	44	55	35
12	81	20	47	58	94	20	58	51	43
13	44	20	96	52	88	80	35	50	100
14	15	23	17	43	17	72	56	54	85
15	84	49	41	48	83	67	41	54	86

16	18	39	98	45	50	44	36	57	22
17	30	51	1	50	52	47	38	57	63
18	58	26	68	41	49	33	37	52	80
19	47	50	12	50	71	48	48	56	38
20	51	50	76	70	31	73	49	45	69
21	10	38	56	70	95	27	33	53	19
22	48	31	58	42	48	80	59	61	39
23	4	46	49	56	49	65	37	42	24
24	65	51	64	70	24	58	56	56	25
25	76	40	57	60	36	23	43	43	100
26	45	23	42	54	68	45	48	41	98
27	11	24	91	70	28	71	50	61	82
28	52	22	60	65	92	33	39	65	100
29	14	27	94	54	35	40	35	42	16
30	93	36	22	63	89	50	35	58	76
31	10	21	28	46	74	56	32	47	17
32	32	38	42	45	21	70	39	54	80
33	55	36	79	70	85	50	48	44	78
34	93	52	17	40	63	44	60	51	88
35	93	26	43	63	95	34	52	47	23
36	58	42	70	64	95	65	52	63	93
37	39	30	17	46	50	37	46	58	49
38	36	50	34	52	9	40	47	49	29
39	29	28	85	59	81	79	52	63	80
40	62	50	77	70	22	45	54	44	98
41	6	46	33	58	16	32	33	62	33
42	11	46	85	61	22	72	39	50	71
43	65	48	8	62	90	69	58	52	74
44	1	38	37	70	50	40	55	64	31
45	31	27	48	61	31	54	57	48	3
46	27	35	30	53	63	36	38	49	59
47	23	30	66	70	34	71	34	54	11
48	68	31	61	54	11	77	57	41	49
49	35	50	36	59	13	67	51	49	4
50	8	25	16	59	76	61	42	53	47
51	19	22	66	42	79	76	48	54	100
52	78	52	57	50	84	33	31	50	3
53	48	45	13	66	14	33	42	55	95
54	7	25	30	50	20	40	44	40	74
55	27	28	100	53	54	69	34	62	96
56	80	43	73	63	56	46	54	65	72
57	53	22	46	70	31	69	37	49	13
58	48	49	81	61	48	54	37	56	66
59	94	26	23	59	18	24	38	63	100
60	80	30	25	64	84	75	60	40	33

The same previously described algorithm (always connecting points furthest away, this time in nine dimensional space) is used. The numerator of VTSPA was calculated to be  $A=6642.09$  for the data collected by the manager. Then five sets of random data ( $n=60$  points in nine dimensions) in the same range as the original data yielded longest routes of 9164.53, 9385.34, 9296.84, 9155.97 and 9071.01. Therefore,  $VTSPA=A/B=6642.09/9164.53=.72476$  which is considerably outside the area of  $9071.01/9385.34=.96651$  and  $1/.96651=1.03465$  where the  $5 \times 4 = 20$  quotients from the random data are. Therefore, the hypothesis of equal risk can be confidently rejected. The manager can then use

this information to perhaps improve customer and employee satisfaction. The goal of reducing variability is usually acceptable to all.

### FTSP FORECASTING EXTENSION OF CTSP

Given a large multivariate data set, testing it for means (MTSP), variance or deviation (VTSP), an underlying distribution (with the DTSP statistic), and possible correlation among variables (with CTSP) are four important areas of statistical data analysis. However, let us say that a future CTSP statistical test for correlation among the variables represented by the data in question, is successful in showing that a correlation exists. That may be sufficient in some cases. However, in selected applications, the researcher may desire to fit an equation to the highly correlated data. A variation of the CTSP approach for correlation could also be used for curve fitting or so called "forecasting." Let us call this the FTSP approach (F for forecasting with a TSP algorithm).

Briefly, given  $k$  columns of  $n$  rows of data (representing  $n$  samples of  $k$  variables), they would first be tested to see if a correlation exists. A shortest route connecting the  $n$  points in  $k$  dimensional space would be calculated with an appropriate TSP algorithm (MSMCO or other suitable ones). Then repeated random samples of size  $n$  in the ranges of the  $k$  columns of the original data, would have their shortest routes also calculated. Then if the shortest route from the original data was statistically significantly less than these random data sets' shortest routes, that would indicate the original data was more compact and hence correlated in some way. Keep in mind that this correlation could be functional or not represented by a function. Therefore, in some cases proceeding with a curve fit would be a good idea (but not always).

However, let us say that it is thought that variable one (represented by the first column of data) is dependent on the other  $k - 1$  variables. Then for whatever functional form is to be used in the forecast or curve fit, the FTSP approach would be as follows. Use multi stage Monte Carlo optimization (MSMCO) to vary the betas (or constants) in the functional form and close in on the best fit for the data. For each set of betas under consideration (in the MSMCO simulation), random sample an  $n$  by  $k - 1$  array in the ranges of the  $k - 1$  independent variables and evaluate the function to get the new  $Y$  values (for column one). Then attach this new  $n$  by  $k$  array to the  $n$  by  $k$  original data and find the shortest route connecting these  $2n$   $k$  dimensional points. If this distance is less than the comparable distance of the "best answer so far" store these betas. Then proceed centering the rest of the simulation around these betas until a better yet answer occurs. Then store that one and keep proceeding with the MSMCO simulation until the betas are fixed after dozens of subsequent MSMCO stages (hence multi stage optimization).

This FTSP approach should allow the fitting of any type of function (linear or nonlinear) and even some relations. It will require a lot of calculation, but our computer age provides us with powerful, inexpensive desktop computers.

Computers today are used to efficiently carry out the massive calculations necessary to apply the traditionally accepted mathematics solution techniques that have been developed over the last several hundred years. This is obviously an important area of computing. However, let us also develop new statistics and new solution techniques that were unavailable and not possible before the computer age became a reality. Examples of these might be MSMCO and the TSP statistics.

## CONCLUSION

Desktop computers are so powerful and available in the 21<sup>st</sup> century that new statistics can be used to as they say “mine large data sets” effectively and quickly. The TSP class of statistics is an entry into this area of so called data mining. The VTSPA statistic to test for deviation in multiple populations was featured here. An additional seven dimensional example is in (Conley 2003). Also, the new DTSP statistic (Conley 2005) will test multivariate data to see which distribution it may have come from. It makes heavy use of the multistage Monte Carlo simulation shortest route (TSP) adjusted for  $k$  dimensions and the standard normal approximation to the runs test for randomness.

Our computer age creates the possibility of the world being overwhelmed by large masses of data. However, it also creates new opportunities for us to “mine it” so to speak to find “precious” pieces of useful information, using the new FTSP, CTSP, MTSP, VTSP and DTSP multivariate simulation based TSP statistics for forecasting, correlation, means, variances and distributions.

## REFERENCES

- Conley, W.C. 1993 “Multi Stage Monte Carlo Optimization Applied to a Six Hundred Point Traveling Salesman Problem.” *Int J. of Systems Sci.*, Taylor & Francis, Vol. 24, 609-626.
- Conley, W.C. 1994 “Multi Stage Monte Carlo and Nonlinear Test Problems” *Int J of Systems Sci*, Taylor and Francis, Vol. 25, 155-171.
- Conley, W.C. 2003 “The TSP Statistics and Total Quality Management and Shipping.” *Proceedings of the International Workshop on Harbour, Maritime and Multimodel Logistics Modelling and Simulation*, HMS 2003 (September 18-20, 2003, Riga, Latvia) Riga Technical University, 53-57.
- Conley, W.C. 2005 “A New Multivariate Distribution Statistic Applied to Health Care Research.” *Proceedings of the 2005 International Conference on Health Sciences Simulation HSS 2005*, (January 2005, New Orleans, LA), SCS San Diego, 55-60.

## AUTHOR BIOGRAPHY

**William C. Conley** was born in Lansing, Michigan, U.S.A. and went to Albion College where he studied mathematics and obtained a degree with honors in 1970. He then received a masters degree in mathematics from Western Michigan University in 1971 and an M.Sc. and Phd in mathematics and computer statistics from the University of Windsor in 1973 and 1976. He joined the faculty of the University of Wisconsin at Green Bay in 1977 and is now professor of business administration and statistics. He teaches introduction and advanced business statistics courses and does statistical optimization and computer statistics research. The developer of multi stage Monte Carlo optimization and the TSP class of statistics he is the author of five books and more than 185 publications worldwide. His favorite research problems are the shortest route or traveling salesman problems (TSPs), especially the ones adapted to higher dimensions ( $k = 3, 4, 5, 6, \dots$ ). He is a member of the American Chemical Society and a senior member of SCS since 1994. He is a fellow in the Institution of Electronic and Telecommunication Engineers of India, a member of Phi Beta Kappa (national academic honorary), Omicron Delta Kappa (national leadership honorary), Omicron Delta Epsilon (national economics honorary), Sigma Beta Delta (International Honor Society in Business, Management, and Administration), and a Michigan Scholar in College Teaching. He was elected to the Albion College Athletic Hall of Fame in 1995 and was recently named to the 2005 Edition of Marquis's Who's Who in America.

# A BDI APPROACH TO AGENT-BASED MODELLING OF PEDESTRIANS

Nicole Ronald and Leon Sterling  
Department of Computer Science and Software Engineering  
The University of Melbourne, Victoria 3010, Australia  
Email: {naron,leon}@cs.mu.oz.au

## KEYWORDS

Transport modelling, agent-based simulation, BDI architecture

## ABSTRACT

The modelling of pedestrian behaviour in a real-world environment is a complex problem, mainly due to the unpredictable nature of human decision making. Agent-oriented simulation moves away from the traditional all-knowing and “controlling” simulations and towards reality, where pedestrians exhibit different behaviours depending on their knowledge of the environment and other personal characteristics. We look the behaviours that pedestrians may exhibit and the different techniques used for pedestrian modelling. We then explore whether the belief-desire-intention (BDI) architecture is appropriate for this domain. The simulation requirements and constraints are presented, followed by a high-level conceptual design using Prometheus, an agent-oriented design methodology, and a discussion of the implementation using JACK, an agent-oriented programming language based on the BDI architecture. Although the BDI architecture is useful for high-level decision making, further work is required in representing and updating the environment.

## INTRODUCTION

There is a need to model pedestrian behaviour for a range of applications including event planning, resource usage, and urban planning. For example, the organisers of a large sporting event require information on what areas are likely to be congested so that management strategies can be developed and tested before the event. Similarly, the designers of a shopping mall might be interested in how people move around their intended design so they can place shop entrances and seating in useful locations.

Most of the models developed so far fall in one of two categories: large-scale models producing aggregate results and smaller, disaggregate models. Less work has been done for pedestrian modelling in between these two extremes. It is useful to have an understanding of how people will behave in certain situations, such as at a large sporting event or after a change to their environment, during the planning process.

Agent-based simulation appears to have potential

for pedestrian modelling. Each pedestrian could be modelled as an autonomous agent with its own knowledge and goals. This representation is closer to reality than traditional simulation methods as it requires less abstraction. The purpose of our research is to compare different agent-based approaches to modelling pedestrian behaviour, in particular the benefits to disaggregate modelling. The approach we discuss in this paper is based upon belief-desire-intention agents.

This paper is divided into several sections. Firstly, we examine at the need for pedestrian modelling and the properties of the pedestrian system. We then review some existing models and techniques. The design and implementation of our model is covered, followed by a discussion.

## BACKGROUND

We discuss some of the interesting properties of pedestrian systems and the history and need for transport modelling, followed by a survey of pedestrian modelling approaches.

### System properties

The Australian Pedestrian Council defines a pedestrian as “any person wishing to travel by foot, wheelchair or electric scooter, throughout the community” (Australian Pedestrian Council 2004). There are many reasons for walking, and the manner in which we walk changes depending on the purpose.

“Commuters scurry; shoppers meander; bush-walkers trek; power-walkers stride; lovers stroll; tourists promenade; protesters march ... But we all walk.” (Australian Pedestrian Council 2004)

Transport systems are constrained, sometimes weakly. For instance, people cannot cross the road whenever they feel like it - they should find a suitable place (such as an intersection) and wait until it is safe. They also should travel on the pedestrian network (eg. designated paths) at all times, however if it becomes too congested, pedestrians may overflow onto the road or surrounding parkland. A stricter constraint is that pedestrians cannot walk through solid objects or on water.

Most transport modelling techniques have focused

on the modelling of cars and vehicles on the road network, as congestion and environmental effects of car travel are pressing problems for cities. The recent interest in environmentally sustainable transport modes, however, has led to an interest in providing better infrastructure and facilities for cyclists and pedestrians and therefore a need for improved methods of modelling their behaviour.

Pedestrian behaviour is usually individual-based and autonomous. In most cases, we decide where we want to go and how to get there without being told explicitly. This behaviour lends itself nicely to the agent-oriented paradigm in principle.

### **Previous work**

There are many approaches to modelling pedestrian behaviour, which can be divided into two schools. The first school is the “civil engineering” approach. This is concerned with forecasting demand so that decisions can be made about provision of new infrastructure. The main outputs of these models are numbers of people travelling along various routes and the algorithms used are frequently based on traditional vehicle modelling algorithms. They are generally macroscopic or aggregate models, where the smallest detail of a pedestrian’s movement is the locations they visited and the paths they used to get there.

The second school is the “architecture/urban geography” approach. This group is interested in how people move around areas, in particular how design and location of certain attractions influence their movements. These models are usually microscopic, in that they model a pedestrian’s path in more detail, usually in terms of steps or small grid squares. They are usually developed for small areas only, although some have been expanded to cover entire cities. Some models combine both approaches and as a result are very flexible regarding the type of areas they can model.

Mathematical models, such as regression models and Markov models, have also been used to model pedestrian behaviour at aggregate levels (Harney 2002). Regression models are of limited use as their only output is pedestrian volume at a particular time and place. Markov models are also of limited use due to their complexity.

Physical models have also been used to model pedestrians. Helbing (Helbing et al. 2001) used the notion of attraction and repulsion to model microscopic behaviour and has developed complex equations to model a range of pedestrian behaviours, commonly referred to as the “social force” model. Hoogendoorn and Bovy (Hoogendoorn and Bovy 2003; Hoogendoorn and Bovy 2004) used the same starting point of basic mechanics formula and developed a three-layered model encompassing

activity choice, wayfinding, and walking. This model attempts to minimise the cost of walking and was applied to a multi-modal transfer station.

A similar approach is the use of cellular automata (CA), where pedestrians occupy cells on a grid and move according to some simple rules (Dijkstra et al. 2001; Henein and White 2004; Narimatsu et al. 2004). Most of these models used the Schreckenberg-Nagel approach to modelling vehicle traffic using CA as a starting point (Nagel and Schreckenberg 1992). This has been shown to be useful for disaggregate models with minimal activity choice. AlpSim (Gloor et al. 2004) combines a cellular automata approach with a graph-based representation of the environment to take advantage of the benefits of both map representations, specifically higher-level planning which is very complex using a grid.

Traditional time-based simulation has also been used in industry. PAXPORT, developed by the consulting firm Halcrow, has been used to model pedestrian movements in airports, train stations, and sporting venues. It provides aggregate measures of flow and level-of-service in a graph-based environment. It was recently used to model behaviour in the Sports and Entertainment Precinct, Melbourne in order to select a design for a new bridge to be built for the 2006 Commonwealth Games (Ronald 2004).

## **MODELLING ASPECTS**

### **Overall Architecture**

Transport systems can be broken down into three main concepts: user, vehicle and environment. The user has a perception of attributes of the environment and their vehicle, and needs to guide their vehicle through the environment. The vehicle interacts with and changes the environment. The environment is constantly updated with the new locations of vehicles and provides perceptions to vehicles and users.

For a driver, the user is the driver in the car, the vehicle is the car, and the environment is the road network and the other vehicles. For pedestrians however, the user and vehicle are essentially the same object: a human. However, most of our walking is done subconsciously and therefore it is permissible to separate these two concepts. We can define the user as the human’s brain and the vehicle as the human’s legs.

Given the high-level decision making role of the user, the belief-desire-intention architecture would be useful for modelling user behaviour in transport systems.

### **The BDI Architecture**

The philosophical component of BDI is based upon

practical reasoning. Practical reasoning is defined as reasoning toward actions, as opposed to theoretical reasoning, which is reasoning about beliefs. Practical reasoning can be broken down further into two activities: deliberation (deciding what goals to achieve) and means-end reasoning (how to achieve a goal) (Wooldridge 2000). Another nice feature of the BDI architecture is the ability to act in both a reactive and proactive manner, however there is a danger of being too reactive or too proactive.

The key concepts in the BDI architecture are:

- beliefs: what I know or don't know about the world;
- desires: what I want to do;
- intentions: how I plan to do what I want to do.

Previous work, as described earlier, has used cellular automata and other methods to develop their agent-based models of pedestrian behaviour. BDI fits our problem well, in that:

- People have beliefs about the environment that affect their decisions (eg. "The main street is always crowded at lunchtime - I will take another route.");
- People have desires to do something or to visit somewhere. This is more obvious with vehicle travel as people do not drive around the city just for something to do, whereas people will sometimes walk somewhere just because it is there. If people are wandering "just because", then that is still a desire;
- People have plans or procedures of deciding where to go first, how to get there, and how to create a path to follow;
- If a route is blocked due to congestion or temporary infrastructure, a new plan can be formulated and a new path taken to reach a location.

### Designing with Prometheus

Prometheus is a methodology developed for specifying agent-oriented software systems. Although there are several design methodologies that could be used for this system (Bergenti et al. 2004), Prometheus was chosen because of its maturity (a book was recently published (Padgham and Winikoff 2004)) and because the concepts used in the methodology tie in with the concepts used in JACK Intelligent Agents, our chosen implementation language. Prometheus consists of three design phases:

- system specification phase: identify functionalities, inputs, outputs and shared data sources;
- architectural design phase: determine agents required and their interaction; and
- detailed design phase: internal design of agents.

The resulting design is a combination of forms and diagrams, which clearly describe the percepts, action, environment, agents, capabilities and plans in the system. Although not tied to any implementation, it fits nicely with JACK Intelligent Agents. It would not be unfamiliar to those familiar with UML for object-oriented design (Rumbaugh et al. 2004).

### System specification

The system specification involves identifying system goals and functionalities, developing the interface between system and environment, and developing use case scenarios.

Firstly, the system goals and possible subgoals need to be established. An example of this is shown in Figure 1, which shows three goals (visit attractions, arrive at the stadium at a reasonable time, move through the environment) and their subgoals using an oval shape. One of the goals is to move through the environment, with the subgoals of satisfying network constraints and taking a reasonable path.

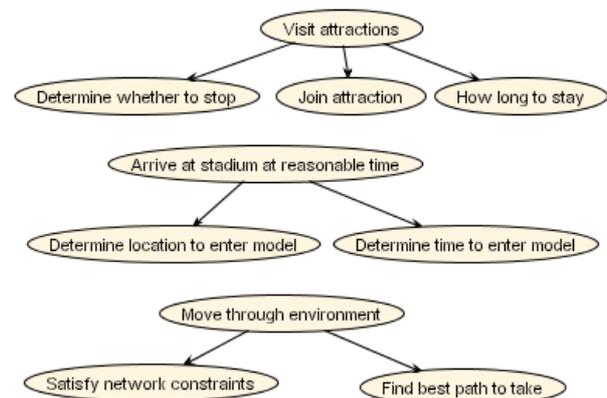


Figure 1: Displaying system goals using Prometheus.

Goals can then be grouped together to create functionalities. Scenarios can also be developed. These consist of steps such as percepts, messages, goals and actions. The system interface involves determining the actions and percepts of the interface to the environment.

### Architectural design phase

The architectural design phase involves grouping the functionalities into similar areas, developing agents to control each area and specifying the interactions between agents.

The functionalities can be grouped into areas by investigating the data that is present in the system and which data is required by each functionality. This is then used to determine the agents required and what functionalities they should have. Figure 2 shows an example of a coupling diagram. Each data source is represented by a yellow cylinder and the functionalities by rectangles. The arrows signify whether the functionalities read

(arrowhead at the functionality end) or write (arrowhead at the data end) data.

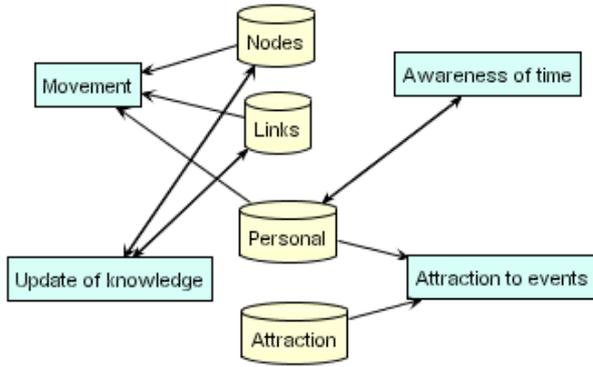


Figure 2: A coupling diagram in Prometheus.

As part of this phase, communication between agents can also be specified at a high level.

#### Detailed design phase

In the detailed design phase, the agent's capabilities, events, plans and data structures are developed in more detail. An example of a detailed design for the Pedestrian agent is shown in Figure 3. This shows the events (percepts: stars shapes; actions: arrow shapes; messages: envelope shapes), capabilities (rounded rectangles), and plans that the agent requires. The arrows again signify the incoming and outgoing nature of events.

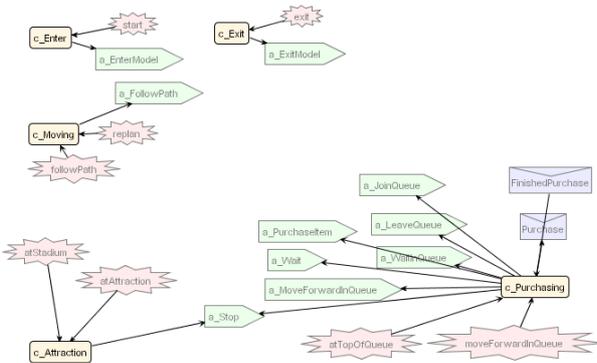


Figure 3: A detailed design for a Pedestrian agent.

Capabilities are similar to modules in that related plans, events, and data can be combined together in a coherent manner. In this model the pedestrian has a capability for each of its main activities.

Events can be actions (affecting the environment in some way), percepts (knowledge coming from the environment), and messages (to and from other agents). For each of these concepts, several parameters need to be designed including the information carried by the percept/message, the effect of the action, and what to do

in case of failure.

Descriptions of data usage are also required. Plans need to specify whether they are reading and/or writing data. Figure 4 shows two plans (p\_Startup and p\_UpdateEnvironment) that use three data sources (Links, Nodes, and Attractions). p\_Startup has write-only access, whereas p\_UpdateEnvironment has both read and write access to the data sources.

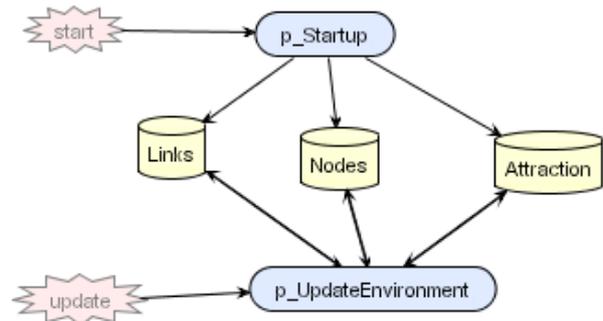


Figure 4: A diagram showing the data usage.

In this stage, plans are described at a high level, including their name, the percepts that trigger them (as shown in Figure 4 by the star shapes) and the actions that occur during the plan. They will be designed in more detail depending on the implementation platform.

The main issue with using an agent-oriented methodology is that it designs the simulation only i.e., what the agents are doing. It cannot design the core of the simulation i.e., how the clock will tick over, the graphical user interfaces required to set up the simulation, the methods to collect outputs. Therefore Prometheus needs to be combined with another methodology to design the whole of the simulation.

#### Implementation

We constructed a prototype model in JACK Intelligent Agents which involved agents entering a sports precinct and moving towards a stadium. Several "distractions" were located on the way to the stadium, such as food stands and street performers. In our prototype, we attempted to implement the entire model architecture (user-vehicle-environment) in JACK to avoid complex interfacing.

JACK is based on the BDI architecture and was purpose-built for simulations, in particular defence simulations. The aim of the package was to develop a stable, lightweight and practical agent-based programming language that would not be superseded quickly and would facilitate further research. It is based on Java with a few syntactic extensions, and when compiled compiles to Java code (Busetta et al. 1999). The product has been used for several applications, mainly within de-

fence, however it has a strong reputation worldwide both in research and industry (Agent Oriented Software 2005).

JACK supports the concepts in the BDI architecture and Prometheus: agents, events, beliefs (data), capabilities, and plans. Appendix A shows a plan (p\_Stopping.plan) written in JACK code. As the JACK files are compiled into Java before execution, normal Java statements can be embedded in JACK files.

It is straightforward to implement goal-directed behaviour, such as moving towards the stadium. The belief system, however, is similar to facts in a logic programming language such as Prolog and does not handle complex beliefs well. For example, it is difficult to represent an environment in detail using JACK belief-sets. Ideally an interface to the environment should be developed and then any environment format (eg. graph, cells, shapes) could be used behind that.

The decision-making used in BDI and in JACK cannot elegantly handle continuous events, such as stepping. It is also difficult to define the subconscious decisions behind walking. Therefore the vehicle model in our architecture would be better suited to an object representation rather than an agent one. JACK has the ability to interface with both Java and C++ code, however we are still experimenting with this interface.

An advantage of agent-based technology is that agents are capable of doing several things concurrently without trouble eg. walking and looking about. So, if you are walking to the post office and you see a shop that has a sale, you continue walking as you make a decision whether to detour or not. If you decide to keep going, you continue with your existing walking plan. However, if you decide to detour, you stop your current walking plan and construct another to get to the sale. We found JACK had problems interrupting one plan before starting another and in our model, agents sometimes found themselves in two places at once. A solution is to increase the lookahead of the pedestrian agent, so that they make decisions and construct new plans earlier.

## CONCLUSION

In this paper, we explored the need for pedestrian modelling, the nature of pedestrian behaviour and techniques for modelling it. In particular, we investigated using the belief-desire-intention (BDI) architecture to model pedestrian behaviour using the design methodology Prometheus and the agent-oriented language JACK Intelligent Agents.

We found that the user-vehicle-environment architecture is an appropriate separation for transport models, and applies to pedestrian models even though the user and vehicle are physically the same. The BDI architecture is appropriate for the user model only, as that is

where the decisions are made. Prometheus is useful for designing the BDI concepts required. For the vehicle and environment model, an object approach is more suitable than using an agent language such as JACK.

The work is continuing as part of a larger project to evaluate approaches and methodologies for modelling pedestrian behaviour. The next stage is to learn from our prototype and use JACK Intelligent Agents to develop a model of the user, following the Prometheus specification, and connecting it to environment and vehicle modules written in Java.

## ACKNOWLEDGEMENTS

The first author thanks Dr. Michael Kirley and Julien Fischer for their assistance.

## APPENDIX A: SAMPLE JACK CODE

```
package pedestrian;
import data.*;

public plan p_Stopping extends Plan {
    #handles event m_Stopping att;

    #posts event m_Replan replan;

    #uses data b_Personal personal;
    #uses data b_Attraction attraction;

    #uses interface a_Pedestrian self;

    body() {
        logical int node, id, m, n, type,
            open, close, distract, l,
            s, swing;
        logical String name;

        attraction.get(id, att.where, n,
            type, name, open, close,
            swing);

        personal.add("move",0);
        personal.add("seenAttraction",0);
        System.out.println(self.name()
            + ": " + "stopping");
        personal.get("currentNode", n);
        personal.get("currentLink", l);
        personal.get("currSpot", s);
        @subtask(replan.post(n.as_int(),
            l.as_int(),s.as_int(),
            att.where));
        System.out.println(self.name()
            + ": " + "stopping now");
    }
}
```

## REFERENCES

- Agent Oriented Software. 2005. "Agent Oriented Software (AOS) - JACK Intelligent Agents, software agent system." <http://www.agentoriented.com/shared/home/index.html>, accessed 4 April 2005.
- Australian Pedestrian Council. 2004. "The Pedestrian Council: Pedestrian Council Home". <http://www.walk.com.au>, accessed 9 December 2003.
- Bergenti, F.; M.-P. Gleizes; and F. Zambonelli. 2004. *Methodologies and Software Engineering for Agent Systems*. Springer.
- Busetta, P.; R. Rönnquist; A. Hodgson; and A. Lucas. 1999. "Light-Weight Intelligent Software Agents in Simulation". Technical Report 3, Agent Oriented Software.
- Dijkstra, J.; A. Jessurun; and H. Timmermans. 2001. "A multi-agent cellular automata model of pedestrian movement". In *Pedestrian and Evacuation Dynamics*, M. Schreckenberg and S. Sharma, (Eds.), Springer-Verlag, Berlin, 173–181.
- Gloor, C.; P. Stucki; and K. Nagel. 2004. "Hybrid techniques for pedestrian simulation". In *Cellular Automata: 6th International Conference on Cellular Automata for Research and Industry, ACRI 2004*, P. Soot; B. Chopard; and A. Hoekstra, (Eds.), volume 3305 of *Lecture Notes in Computer Science*, Springer-Verlag, 581–590.
- Harney, D. 2002. "Pedestrian modelling: current methods and future directions". *Road & Transport Research* 11, no. 4, 2–12.
- Helbing, D.; P. Molnár; I. J. Farkas; and K. Bolay. 2001. "Self organizing pedestrian movement". *Environment and Planning B: Planning and Design* 28, 361–383.
- Henein, C. M. and T. White. 2004. "Agent-based modelling of forces in crowds". In *Multi-Agent and Multi-Agent-Based Simulation*, P. Davidsson; L. Gasser; B. Logan; and K. Takadama, (Eds.), volume 3415 of *Lecture Notes in Computer Science*, Springer-Verlag, 173–184.
- Hoogendoorn, S. and P. H. L. Bovy. 2003. "Simulation of pedestrian flows by optimal control and differential games". *Optimal Control Applications and Methods* 24, 153–172.
- Hoogendoorn, S. and P. H. L. Bovy. 2004. "Pedestrian route-choice and activity scheduling theory and models". *Transportation Research B* 38, 169–190.
- Nagel, K. and M. Schreckenberg. 1992. "A cellular automaton model for freeway traffic". *Journal de Physique I* 2, no. 12, 2221–2230.
- Narimatsu, K.; T. Shiraishi; and S. Morishita. 2004. "Acquisition of local neighbor rules in the simulation of pedestrian flow by cellular automata". In *Cellular Automata: 6th International Conference on Cellular Automata for Research and Industry, ACRI 2004*, P. Soot; B. Chopard; and A. Hoekstra, (Eds.), volume 3305 of *Lecture Notes in Computer Science*, Springer-Verlag, 211–219.
- Padgham, L. and M. Winikoff. 2004. *Developing Intelligent Agent Systems - A Practical Guide*. John Wiley and Sons.
- Ronald, N. 2004. "The role of pedestrian modelling in planning for special events". In *Proceedings of the 27th Australasian Transport Research Forum*.
- Rumbaugh, J.; I. Jacobson; and G. Booch. 2004. *The unified modeling language reference manual*. Addison-Wesley, Boston, MA.
- Wooldridge, M. 2000. "Intelligent agents". In *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, G. Weiss, (Ed.), MIT Press.

## AUTHOR BIOGRAPHIES

**NICOLE RONALD** graduated with undergraduate degrees in computer science and civil engineering from the University of Melbourne in 2002. She is currently a postgraduate student in the Department of Computer Science and Software Engineering at Melbourne. Between December 2002 and January 2005 she worked as a graduate transport modeller for an global engineering consulting firm in Melbourne. Her webpage is [www.cs.mu.oz.au/~naron](http://www.cs.mu.oz.au/~naron) and her email address is [naron@cs.mu.oz.au](mailto:naron@cs.mu.oz.au).

**LEON STERLING** is Adacel Chair of Software Innovation and Engineering at the University of Melbourne. He is a co-director of the Intelligent Agent Lab ([www.cs.mu.oz.au/agentlab](http://www.cs.mu.oz.au/agentlab)) and his research interests include agent-oriented software engineering. His email address is [leon@cs.mu.oz.au](mailto:leon@cs.mu.oz.au).

# SIMULATION MODEL OF THE LOGISTIC DISTRIBUTION IN A MEDICAL OXYGEN SUPPLY CHAIN

Francesco Costantino  
Giulio Di Gravio  
Massimo Tronci  
Department of Mechanical and Aeronautical Engineering  
University of Rome "La Sapienza"  
Via Eudossiana 18, 00184 - Rome, Italy  
E-mail: [francesco.costantino@uniroma1.it](mailto:francesco.costantino@uniroma1.it)

## KEYWORDS

Logistic supply chain, distribution, simulation, vehicle routing problem, service level, decision support system.

## ABSTRACT

Research activities on operations management in the last years are always more dedicated to supply chain and logistics optimization models.

The study belongs to this branch and describes the problems related to a re-configuration of the distribution net in a company that produces medical oxygen cylinders for Italian market. The enterprise is particularly sensible to the optimization of supplying processes due to the characteristics of its product, as any delay in the delivery could create dangerous health situation for patients.

The work has the objective to realize a software for supply chain management that could be a decision support system, analyzing strategic impacts that changes in distribution system create. In details, the model shows the differences in service level in case of closing one or more factories and the relative necessary changes in logistics net.

The paper is articulated in the following parts:

- analysis of company and construction of simulation model;
- study of classic operation research techniques to solve dynamic vehicle routing problems;
- description of possible scenes derived by strategic decision in closing factories;
- analysis of experiments and global conclusions and developments.

## RESEARCH CONTEXT

Logistic distribution represents a key factor in the competitive market. In fact, fidelization of customers and market share are necessarily related to the service level, as an increase in capacity allows higher sales, and to reduction of costs, that's to say:

- an optimal inventory management, avoiding stock-outs and minimizing immobilization;
- production programs adapted to distribution requirements;
- focused investments on supply system.

In our case study, distributive logistic is not only an operative function but a real strategic activity so, it has to be managed at direction level to be effective on company's objectives.

The study has the purpose to define a dedicated informative tool to be a support on decisional process related to the optimization of logistic chain.

In details, the proposed solution gives the possibility to stop production processes of one or more factories with an automatic optimal assignment of customers to the other production centres. The products in analysis are oxygen cylinders in both gaseous and liquid form.

Supply chain frame (fig. 1) and production process (fig. 2) are constituted of many different parts, here described:

1. primary transportation for gas distribution, from the unique raw material supplier to finished product plants. Primary transportation is different from the one from plants to customers and has independent means;
2. secondary distribution is structured to give the possibility of connecting any plant with any customer and all the customers themselves;
3. distribution links different customers categories: patients, hospitals, agencies and depots. All these customers are grouped in selling centres that place orders, depending on the demand of the single customers;
4. about cylinder filling process, it's to consider the different problems related to liquid and gaseous products;
5. at the same time, the cylinders are to be differentiated because of their coupling system and technical specifications;
6. the activation of a transport happens when a vector exhausts its load capacity or when the evaluated delivery time is higher than a fix maximum time;
7. all the customers are to be served, if possible, minimizing the distance for each single transport, so to reduce logistic costs;
8. the plants produce also industrial oxygen as raw material for other customers; but, as the logistic problems of this sector are limited, this process is not considered in the simulation model.

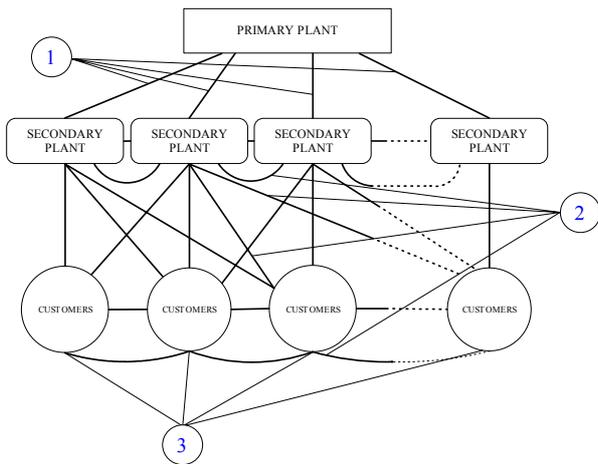


Fig. 1 – Medical oxygen supply chain

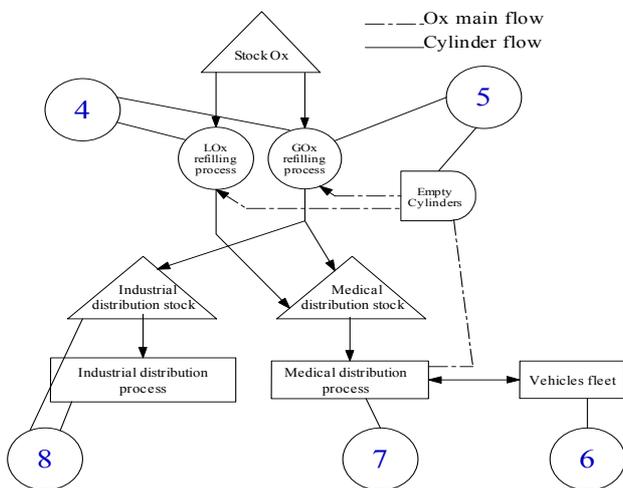


Fig. 2 – Secondary plant production process

Two further elements define the problem. First, cylinders for medical use need very short delivery time due to critical situation that a customer stock-out generates. Then, as the value of the cylinder is ten times higher than its contents, it's necessary to define a circuit of empty ones to come back. Production plants and selling centres are principally located in northern Italy, with decreasing density going to south (fig. 3). As a consequence, in the city of Trieste there is the production of raw materials. Its particular geographic position it's to consider in establishing reorder levels of the different secondary plants. It's obvious that the different distances influence supplying delivery time and so the availability of products.

### CUSTOMERS MODEL

As described, the simulation represents a high level national logistic, without modelling local distribution, and the customers are grouped and summed up in city centres. The customer is modelled as a FIFO buffer and

its stock level is continuously updated. For this particular product, a service level of 100% is needed and this explains a punctual monitoring to absolutely avoid stock-outs, together with average delivery time established by strict contractual clauses (maximum delivery time is 24 hours after a placed order).



Fig. 3 – Company territorial distribution

Following this requirements, order management is arranged with a double kanban flow (fig. 4): each full cylinder in stock has a production kanban associated that is sent as a production order anytime the respective cylinder is delivered, so to maintain a constant level of finished product in the warehouses. A movement kanban follows the cylinder to the customer.

The model represents the real transportation system, simulating the cylinders gathered together to form a load unit of 16 elements in standard dimensions, so to be easily moved by forklift in the factory. The transport to customers is effectuated by trucks of 10 load units, for a total of 160 cylinders (fig. 5).

As first step, the model assigns each customer to a particular production plant with a geographic criterion, having inside a function to redirect the orders to the other centres if the original is closed. The demand curve for selling centre is defined as the sum of the demands of each customer, based on historical series, transformed into statistic distribution. So, the customers (fig. 6) are composed of:

- empty cylinders buffer;
- full cylinders buffer;
- aggregated demand curve;
- order management activity.

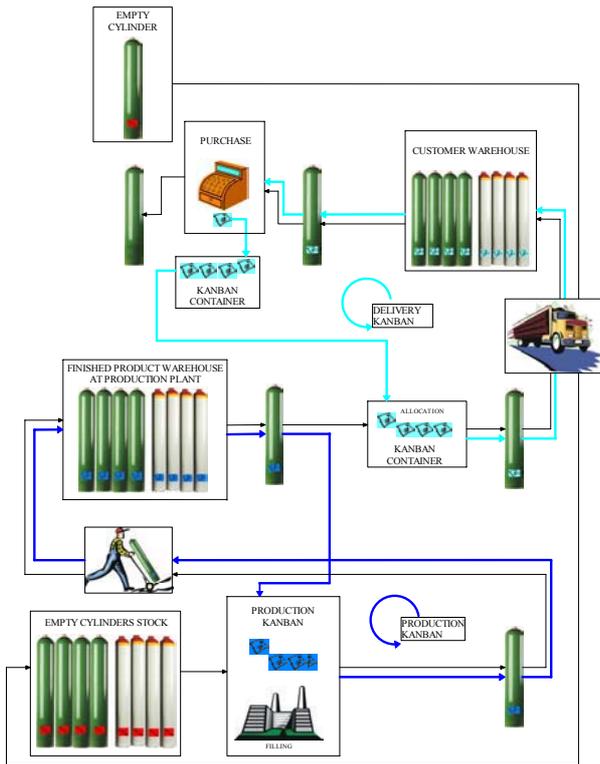


Fig. 4 – Double kanban

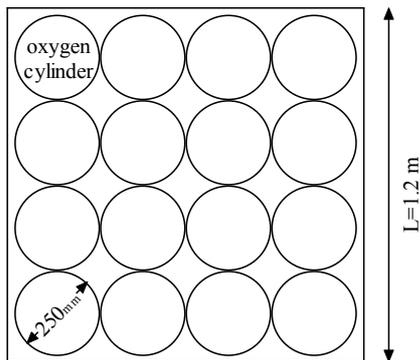


Fig. 5 – Light metal alloy load unit

Reminding the necessity of empties recovering, after delivering, an appropriate algorithm calculates the capacity left to completely fill the trucks from customer to customer. After having completed these operations, the truck is sent to the next customer, following an

appropriate sequence of visit. In fig. 7, all the activities related to the customer are defined in a flow chart.

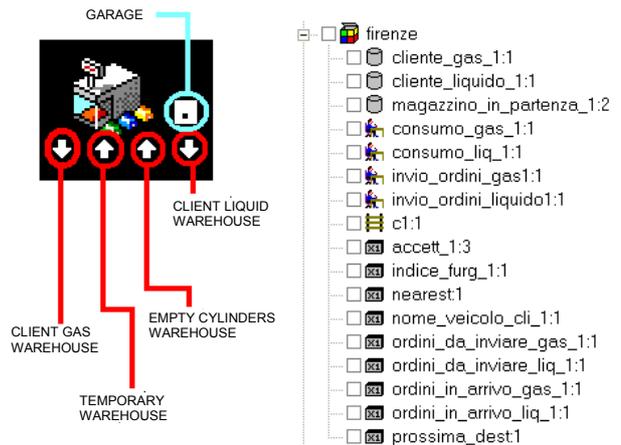


Fig 6 – Customer modelling

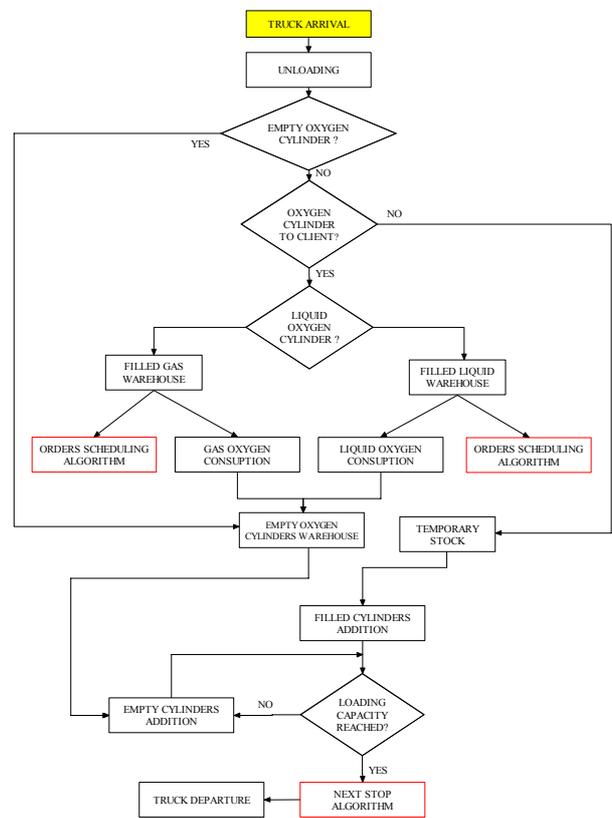


Fig 7 – Customer related processes

## PLANT AND DISTRIBUTION MODEL

The production plant is designed only in the activities that can influence distribution logistic. Production activity of cylinders is oriented to guarantee a certain level of stock. The finished product is assigned to a specific customer in the same sequence as the orders arrive. Then, the gathering of cylinders in load units and after in trucks is effectuated considering, already at this

stage, the final destination of the product, so to avoid many different deliveries to the same customer in short times. The effect is a bypass of FIFO order sequence adopted in the assignment of cylinders to customers to carry at the same time the amount of product of an order and all the short next. The logic frame is presented in fig. 8.

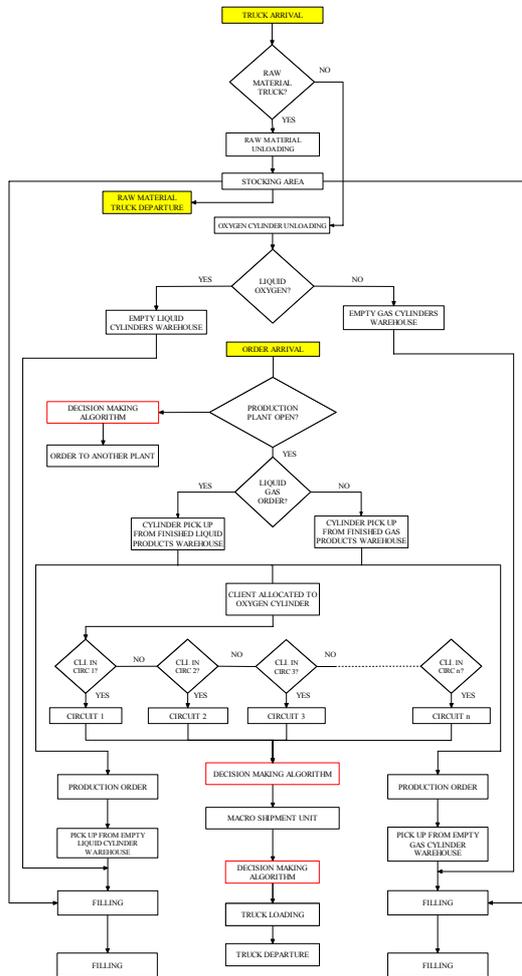


Fig. 8 – Production plant related process

About the grouping of orders based on their final destination, it is necessary to create circuits to link nearest, or easiest to reach, customers. Once chosen the customer to satisfy, according to waiting time of the assigned cylinder, all the product of the same circuit are loaded. After having verified the capacity on the truck, all the product of a different circuit are loaded. Each plant has default assigned circuits on a geographic criterion (fig. 9). On this allocation of customers in neighbour groups, the algorithm of distribution management and scheduling are implemented. Referring to bibliography study on models and algorithm in logistics, having to face an analytic close model with linear bounds, it was used a combination and integration of simulation with analytic techniques, realizing a tool operating on wide range.

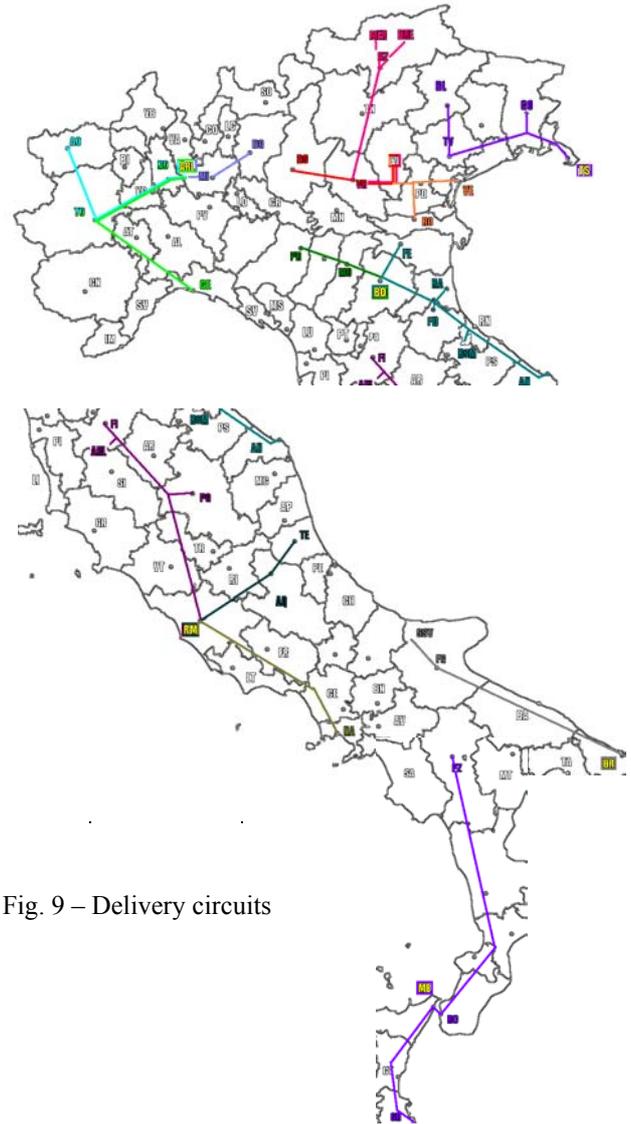


Fig. 9 – Delivery circuits

In particular, the problem regarded a dynamic directing of vehicles, known as Vehicle Routing Problem (VRP), or an optimization of load vectors and sequencing of visits for each vector, determining  $K$  routes, from a central station to a subset of customers in a specific sequence to minimize the global trip cost. VRP can be divided into the Traveller Salesman Problem (TSP), the research of the minimum route for a vector to visit all the customers in different cities, leaving and coming back to its base, and a Clustering Problem (CP), the assignment of customers to the single vectors. These activities have to be carried out considering:

- capacity bound: maximum number of part on the vector;
- time bound: maximum waiting time for orders.

So, it was chosen to implement a VRP algorithm composed by a first clustering step and a second routing step, in a greedy heuristic, where, starting from a knot of reference, the optimal sequence is constructed simply passing on the nearest knot. Therefore, if a circuit can't

complete load capacity of the correspondent vector, another circuit is chosen on delivery time bound, till a time or capacity bound is violated, so obtaining the cluster assigned. This activity is carried out inside the plant by the simulation model on constituting the load for each vector.

For each load unit in a truck, the distance between its destination and all the other destinations of the units already on is calculated. The entering unit is placed in the sequence when its distance is less than the distance that the same destination had. If this doesn't happen, the entering unit is so put at the end of the sequence. When, instead, the current unit is inserted in a sequence already existing, reasonably it creates changes that involve all the successive to the one of insertion; it's so necessary a greedy optimization procedure for the next visits (fig. 10). Furthermore, given the purpose of the study, an algorithm was implemented to allow, in case of closing of a plant, the redistribution of orders to another production site on the beginning of the allocation procedure.

## THE EXPERIMENT

The research analyzed the effects of supply chain configuration changes in delivering activities, in particular the closing of plants in the production and stocking net and the consequent new set up of customer service. The parameters of the simulation (invariant input) are:

- customers demand modelled as negative exponential distribution curve with characteristic value given by historical series analysis;
- maximum waiting time of orders (time in stock before delivery). In defining the value, it's to consider also a reasonable transport time to reach the end user: this highly affects the load of each leaving vector and, if too short, it can cause partially empty trucks. In this simulation experiments, considering the particular urgency of the product and the contractual clauses, it's fixed at 12 hours;

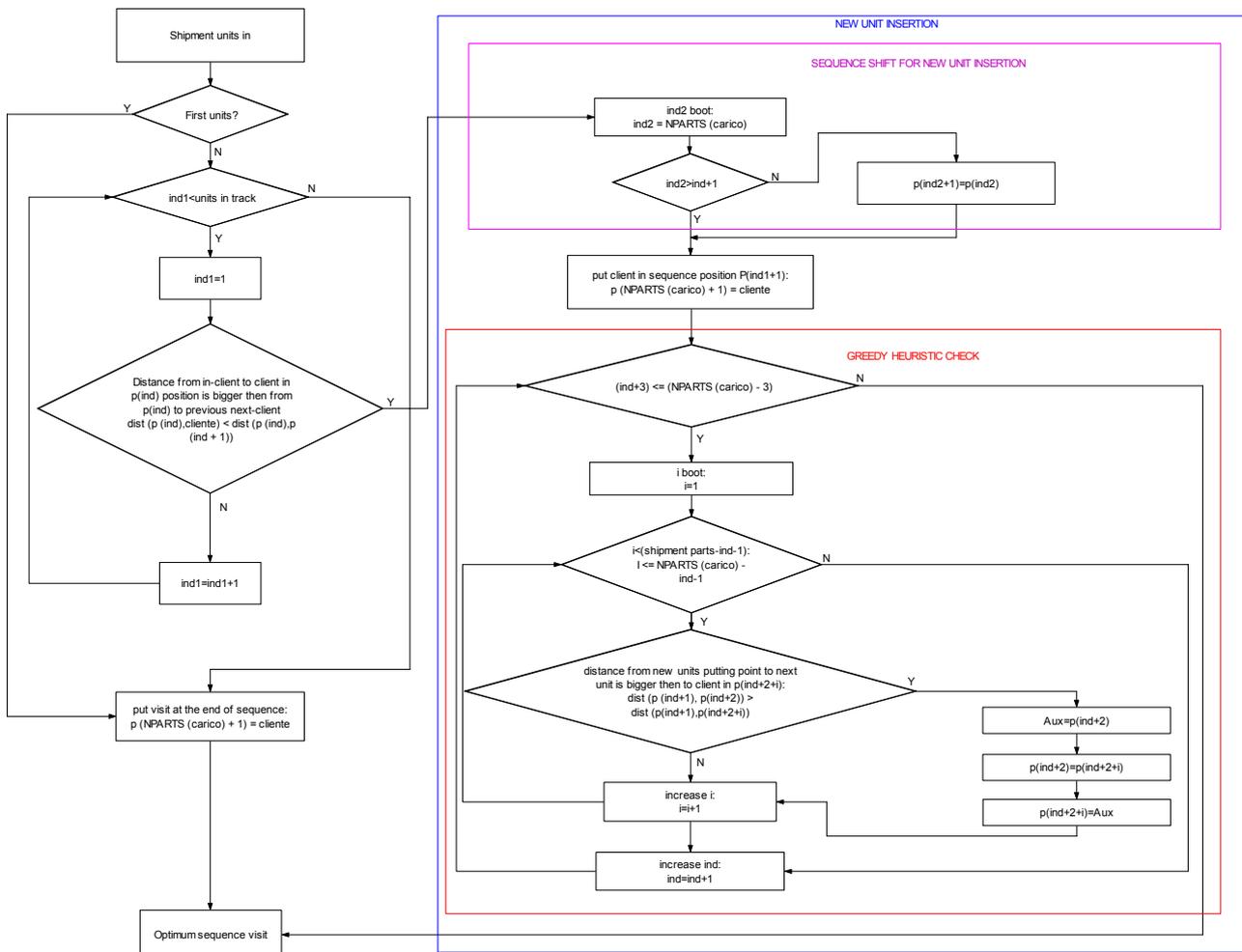


Fig. 10 – Routing algorithm flow chart

- production time and capacity. For the purpose of the study, they are taken in appropriate way to always guarantee the correct amount of product available with no delay.

The output results to analyze are the following:

- average waiting time for orders (average service time) for each customer;
- maximum waiting time (maximum service time) for all the customers;
- number of units transported by vehicles, relative to every production site, to estimate logistic costs;
- number of stock-outs.

The distribution activities is monitored in a campaign of runs of 62 consecutive days, long enough to smooth the transitory effect of the first supplies and give a regime solution.

#### Closing Rome plant

When a production site is closed, the reassignment procedure is automatically activated, shifting the orders to the still active plants, on a geographic criterion. All Rome's customers are diverted to Bologna, the nearest factory. To define the new situation, the service level offered to Naples is monitored, representing the most critical customer for Bologna, as it's the most distant.

Figure 11 sums up the new situation, varying the number of active vectors related to the number of stock-outs generated, to the average waiting time and to the average number of loads on the truck. In table 1 Naples levels of performance, before and after Rome closing, are compared.

$\Delta_{A.W.T.}$  is the percentage variation of average waiting time related to the balanced starting situation.

As it can be seen in the table, average waiting time of Naples, even with an increase of 14 vectors in Bologna (only 5 were active before in Rome), worsen of 67%.

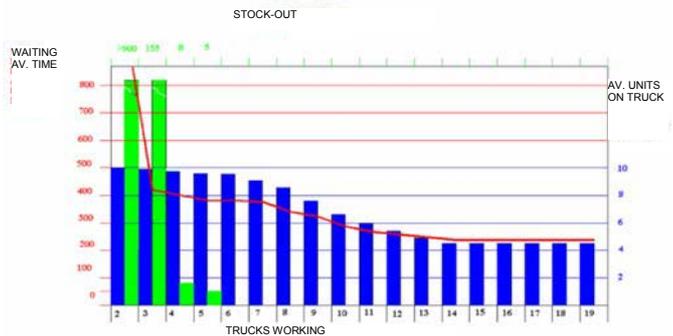


Fig. 11 – Summary of Bologna distribution and Naples service level

Any new vector doesn't allow an increase of level because, always respecting the maximum order wait bound, Naples is too far, so to limit inferiorly the average waiting time of the customer. In fact, there are physical bounds (distances) that prevent from having waiting average time lower than a certain value, as it's impossible to go below 7 hours if it needs 7 hours to reach Naples from Bologna.

With 6 vectors there is already no stock-out but the average waiting time becomes 17h 30m (387 time units), 176% higher than the balanced situation. Nevertheless a service distribution with only 6 vectors is notably more economic, not only for a lower number of vehicles acquired and the related costs (maintenance, consumption, drivers, etc.) but also because they generally travel with 9,52 load units instead of 4,59 in case of 14 vectors, the best service level situation. The maximization of marginal unit, 25%, is obtained passing from 8 to 9 trucks.

Tab. 1 – Comparison of Naples service levels before and after Rome closing

NAPLES								
	All open	Rome closed						
	served by Rome	served by Bologna						
N° trucks	5	2	3	4	5	6	7	8
Av. waiting time	140	///	430	400	390	387	380	370
$\Delta_{A.W.T.}$		///	205%	187%	178%	176%	170%	164%
Av. shipment	5.40	10	9.88	9.74	9.56	9.2	8.2	7.01
N° trucks		9	10	11	12	13	14	15
Av. waiting time		330	320	290	260	240	235	235
$\Delta_{A.W.T.}$		135%	128%	107%	85%	71%	67%	67%
Av. shipment		6.19	5.5	5.02	4.7	4.63	4.59	4.57
N° trucks		16	17	18	19	20		
Av. waiting time		235	235	235	235	236		
$\Delta_{A.W.T.}$		67%	67%	67%	67%	67%		
Av. shipment		4.57	4.57	4.57	4.57			

Considering that the number of vectors of Bologna in the balanced scene is 2, it's to face that without an empowerment of resources the distribution system falls in crisis: each customer has a stock-out and the average waiting time for Naples surpasses 40h. Even acquiring a further truck the situation doesn't improve, with 155 stock-outs in the simulation time. Furthermore, the average waiting time is still at an unacceptable level of 30h.

**Closing Bologna plant**

After having closed Bologna site, the algorithm reassigns Parma customer to Arluno plant and Modena, Ferrara, Ravenna, Forlì, San Marino to Vicenza.

After having balanced the production, San Marino service level is analyzed in the new situation, representing the most critic customer as the most distant (fig. 12). In table 2 San Marino service level is shown, in the two cases of Bologna and Vicenza assignment.

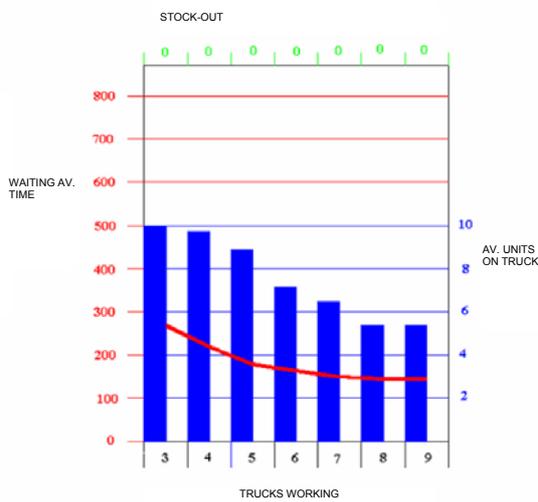


Fig. 12 – Summary of Vicenza distribution and San Marino service level

As it can be noticed, the number of vectors owned by Vicenza in standard situation is sufficient to obtain a service level without stock out.

Increasing the number of vehicles, the improvement of average waiting time goes in a linear way to arrive at

the same level of standard Bologna assignment, with an equipment of 8 vectors that almost double the initial park. Nevertheless this new balanced situation, transport costs are sensibly higher as the average load, in fact, is lower, passing from 8 to 5,6 units.

**Rome and Bologna site closing**

After the closing of Rome and Bologna, they become both customer of the resulting net. The geographic criterion reassigns Teramo and L'Aquila to Brindisi, Firenze, Certaldo, San Marino, Ravenna, Ferrara, Forlì, Modena, and Roma to Vicenza, Parma to Arluno, Napoli to Reggio Calabria.

Rome customer is described as a significant example (fig. 13), representing the most critical customer as the most distant from Vicenza.



Fig. 13 – Summary of Vicenza distribution and Rome service level

At this point, the comparison in table 3 shows two different cases:

- closing of Rome site and subsequent assignment of Rome customer to Bologna;
- closing of both Rome and Bologna and subsequent assignment to Vicenza.

The passage of Rome customer from Bologna to Vicenza causes a general increase in average waiting time, highly varying with the number of trucks considered. For levels of about 15h (330 time units and 8 vectors), the difference between the two scenes is sensibly lower than what happens on higher levels (less than 14h), as seen in figure 14.

Tab. 2 – Comparison of San Marino service levels before and after Bologna closing

SAN MARINO								
	All open	Bologna closed						
	(served by Bo)	(served by Vicenza)						
N° trucks	5	3	4	5	6	7	8	9
Av. waiting time	140	260	230	180	160	150	140	140
$\Delta_{A.W.T}$		85%	65%	28%	14%	7%	0%	0%
Av. shipment	8	10	9.67	9.68	7.35	8.8	6.5	6.5

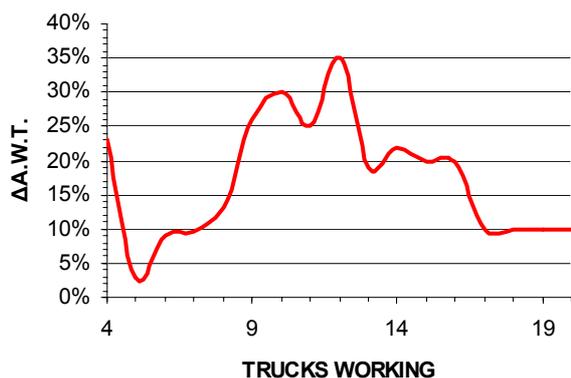


Fig. 14 – Percentage difference on Rome service level of Bologna and Vicenza

After 10 trucks, the value decreases and stabilize at 10%, as the service level depends no more on production site capacity but only on the distance of the two cities.

A global analysis of the results obtained in the simulation runs determines that a site closing always worsens total service level, even more sensibly when the new assignment allocates together distant customers and plants. Closing Rome and guaranteeing a good service level (average waiting time of 14h for the most critic customer) needs an acquisition of at least 8 vectors. The reason of this big increase is that Bologna has to serve 6 clients very far one from the other, someone also from the plant. To confirm what supposed, closing Rome and Bologna creates an increase of 9 customers in Vicenza, but to obtain a service level of 14h for the most critic one it needs only

4 more vectors: the clients assigned to Vicenza are in fact easier to reach and more concentrated by the site. This is more evident analyzing the closing of Bologna that redirects 6 customers to Vicenza, being anyway able to react properly to this new incoming traffic with its vehicles and establishing the standard situation with only 4 more.

### CONCLUSION AND FUTURE RESEARCH

Simulation study showed that, on deliveries point of view, sites with very few customers, if distant, are anyway important to maintain an high service level, more than highly productive factories with many customers but near to other production centres. This means the necessity of having sites opened only for their strategic geographic location even with a not so relevant production.

The decision support system is in its first stage, as it has many further development not already considered:

- after having closed all the production sites and leaving only primary factory, an analysis can be carried out to fix an optimal stock level for each customer, to avoid at best stock outs and minimizing immobilization costs;
- on the same stage, it's possible to define a reorder level for raw materials to avoid production stops and minimize immobilization costs;
- a costs – benefits analysis can be implemented. The natural development of the model allows not only to investigate effects on service level caused by the closing of a production centre, but to completely evaluate its economic impact on company general behaviour;

Tab. 3 – Comparison of Rome service levels before and after Bologna closing

ROME										
Served by Bologna						Served by Vicenza				
3	4	5	6	7	N° trucks	3	4	5	6	7
410	380	340	320	310	Av. Waiting time	///	470	350	350	340
					Δ <sub>A.W.T.</sub>		23%	3%	9%	9.6%
					equal trucks number					
10	9.9	9.81	9.69	9.27	Av. shipment	10	9.88	9.86	9.65	9.34
8	9	10	11	12	N° trucks	8	9	10	11	12
290	260	240	235	215	Av. Waiting time	330	330	310	300	290
					Δ <sub>A.W.T.</sub>	13%	26%	30%	25%	35%
					equal trucks number					
8.4	7.5	6.5	6	5,5	Av. shipment	9.05	8.45	7.87	7.04	6.32
13	14	15	16	17	N° trucks	13	14	15	16	17
210	200	200	200	200	Av. Waiting time	250	245	240	220	220
					Δ <sub>A.W.T.</sub>	19%	22%	20%	10%	10%
					equal trucks number					
4.9	4.5	4.5	4.5	4.5	Av. shipment	5.7	5.25	5.25	4.79	4.52

- tests on different ordering policies for customers are easily designable, increasing their stock level and delivering more than a truck at the same time;
- the simulation tool can be easily refined introducing production programs and activities with the logics related with distributive system. It would be possible to analyze overloads of factories after an increase in their number of customers and the necessary action to prevent (empowerment of production line resources or new investments in capacity). Furthermore, the effects of production stops on customers and the way to react to crisis simulation could be investigated.

It's also remarkable that the modularity of the model allows an easy adaptability to any different supply chain situation. Considering that it's mainly based on a distance matrix loaded at start, any different logistic scene can be implemented.

## REFERENCES

- Archibal G., Karabakal N., (1999), Supply Chain vs. Supply Chain: Using Simulation to Complete Beyond the Four Walls, Iwinter Simulation Conference.
- Bodin L., Golden B., Assad A., (1983), Routing and scheduling of vehicles and Crews: the state of art, Computer Operation Research n°10, pp 63-212.
- Bresnahan J., (1998), The incredible journey - supply chain anatomy, CIO Enterprise Magazine, August 15 1998, p 5-6.
- Banks J. et al (1998), Handbook of Simulation, John Wiley and Sons, New York.
- Haskett J. L., (1973), Macroeconomic Cost of Physical Distribution, ATRF, 1962 and Haskett and others, Business Logistics (2nd ed).
- Hillier F. S., Lieberman G. J., (2001), Introduction to operations research, McGraw-Jill, pp 350-403.
- Lambert D., Stock J., (2001), Strategic Logistic Management, McGraw-hill, New York.
- Law A., Kelton B.,(2000), Simulation modelling and analysis, McGraw-Hill, New York.
- Nozick L. K., Turnquist M. A., (1998), Integrating inventory impacts into a fixed-charge model for locating distribution centers, Transportation Research E, Vol.34, 173-186.
- Nozick L. K., Turnquist M. A., (2001), A two-echelon inventory allocation and distribution center location analysis, Transportation Research E, Vol.37, 425-441.
- Okumura M., Tsukai M, (2003), Distribution Network Configuration Considering Inventory Cost, 43rd Congress of the European Regional Science Association, University of Jyväskylä, Finland.
- Pedgen, Shannon, Sadowsky, (1995), Introduction with simulation using Simann, McGraw-Hill, New York.
- Russel R. S., Taylor III B. W., (2000), Operations Management, Prentice-Hall, pp 371-422.
- Schiber, (1990), Introduction to simulation using GPSS-H, John Wiley and Sons, New York.
- Van Breedam A., (2001), Comparing descent heuristics and metaheuristics for the vehicle routing problem, Computers Operations Research, n°28, pp 289-315.

## AUTHOR BIOGRAPHIES

**FRANCESCO COSTANTINO**, degree in mechanical engineering and master in quality engineering and management, is currently a Ph.D. student in industrial production engineering at the University of Rome "La Sapienza".

His e-mail address is: francesco.costantino@uniroma1.it

**GIULIO DI GRAVIO**, degree in mechanical engineering and master in quality engineering and management, is currently a Ph.D. student in industrial production engineering at the University of Rome "La Sapienza".

His e-mail address is: giulio.digravio@uniroma1.it

**MASSIMO TRONCI**, degree in mechanical engineering and Ph.D. in energetics, is an ordinary professor of industrial plants, operation management and quality management at the University of Rome "La Sapienza", faculty of engineering, full member of EOQ as vice-president of AICQ (Italian Organization for Quality).

His e-mail address is : massimo.tronci@uniroma1.it

# VISUALISING LAYOUT AND OPERATION OF A CONTAINER TERMINAL

Felix A Schmidt,  
Maritime & Supply Chain  
Solutions (Europe) Ltd,  
BALLYCARRY, BT38 9HZ,  
UK  
Email:  
[fa.schmidt@btinternet.com](mailto:fa.schmidt@btinternet.com)

Rahila Yazdani,  
University of Ulster at  
Jordanstown,  
NEWTOWNABBEY, BT37  
0QB, UK  
Email:  
rahilayazdani@hotmail.com

Robert Young,  
University of Ulster at  
Jordanstown,  
NEWTOWNABBEY, BT37  
0QB, UK  
Email:  
r.young@ulster.ac.uk

## KEYWORDS

Container Terminal, Computer Modelling, Planning, Operation.

## ABSTRACT

The proposed paper summarises the work done in WP4 of the EU sponsored RTD project TRAPIST where the needs of small to medium sized ports / terminals with 'multi-purpose' capability had to be addressed. To this end the following had to be considered:

- The absence of extensive IT support and how to deal with it,
- Information flows to/from and within the terminal,
- Cargo flows to/from and within the terminal and in association with this,
- Layout of land and maritime approaches including berths,
- Position of in-and out gates,
- Terminal and yard layout in conjunction with selected cargo handling equipment.

The above resulted in an electronic terminal planning board (TPB) with generic applicability. This TPB was successfully applied to visualise initially the Status Quo and subsequently to explore possible extensions of the Tivoli Container Terminal in the Port of Cork, Republic of Ireland. It was further used to review and improve the layout of the road infrastructure in the Port of Kokkola, Finland.

## INTRODUCTION AND CONTRIBUTION TO EU POLICIES

Implemented and pending European policies are a common denominator for ports. This is particularly applicable to small and medium sized ports, which as a consequence of the European Spatial Development Perspective (DETR,200) and The Common Transport Policy (CTP) (European Commission, 1992) transposed into national law, benefit from the resulting national and regional spatial development and transport plans.

These policies favour polycentric and regional development as clearly identified in the EU sponsored North European Trade Axes Study (NETA, 2002). Small to medium sized ports were found to play a crucial part in this development process by serving as regional logistics nodes and gateways in conjunction with short sea shipping and / or ferry services. The commercial attractiveness of these gateways, through operational efficiency supported by the physical development of associated facilities, their land and maritime access, is crucial to the economic development of the regions.

The European Commission (EC) in its publication "Time to Decide – 2010" (DGTREN,2001), the White Paper of September 2001, reviewed the results of the implementation of the CTP. It revealed the EC's plans of how the CTP can be further and successfully implemented within the framework of 'Sustainable Development' and actions to achieve the all-important 'Balance of the Modes of Transport'. In this respect the development of Freight Corridors, Motorways of the Seas and the removal of bottlenecks were identified as being of particular importance. Combining Short Sea Shipping (SSS) and 'Rail' with 'Road' offers long-distance and pre-/post-carriage respectively. This brings the operation of terminals, the ship-shore interface, and associated bottlenecks into play.

Despite considerable improvements and growth in SSS, port turn-around times of container feeder vessels need to be further reduced to offer a transit time more comparable with those of the combination of road and ferry transport. The EC Fiches "Bottlenecks in Short Sea Shipping", e.g. April / May 1999 (DGTREN,1999), that discuss bottlenecks and possible solutions indicate that port turn-around time can be reduced by ensuring that terminal operations are more efficient and that access to terminals conforms with the access, parking, manoeuvring and berthing requirements of road and rail vehicles as well as ships accessing the port or terminal from land and from sea.

The need to improve 'Terminal Efficiency' has been addressed amongst others during the execution of the 5<sup>th</sup> EU sponsored 5<sup>th</sup> FWP project TRAPIST. Of

particular interest for this paper is the contribution made in Workpackage 4 with the aim of “Developing Means for Gathering and Intelligently use Data from Identified Sources”. These “means” comprise of an “Object Oriented, Relational Database” and an “Electronic Terminal Planning Board”. Together they addresses in a very comprehensive way the bottleneck ‘Port’ described in the EC publication “COM99 317 en final Development of SSS in Europe”, 1999 (EC, 1999), where short comings of port infrastructure and port efficiency were commented upon and ‘port turn-around time’ was singled out as one of these inefficiencies. WP4 proposes “Means” aimed at increasing the efficiency of small to medium sized ports by specifically increasing their commercial attractiveness to regional exporters and importers. This proposition can be achieved by improving the flow of information, cargoes, vehicles and people through the port or terminal. Correct and timely information is the first of two key aspects of this. Solving operational problems quickly and effectively by visualising them, simultaneously enabling the port / terminal operator to easily explore alternate solutions by using common sense based on experience is the second key aspect. Both are paramount to achieve the transfer of the flow of goods away from the at present prevailing mode of ‘Road’ using heavy goods vehicles (HGV), to water-borne transport, particularly to the ‘Motorways of the Sea’ employing short sea shipping as an attractive alternative mode of transport as envisaged by the CPT of the European Union (EU). The benefits of this will affect port communities and port hinterlands throughout the EU from the Mediterranean to the Baltic including those of the New Member States and further to the East.

## OBJECTIVES AND ACHIEVEMENTS

Workpackage 4 (WP4) of TRAPIST had the following major objective:

- To develop a methodology for data gathering and its general representation to visualise and resolve operational problems in ports and terminals.

This major objective was achieved and took the form of:

- An object oriented relational database and, based on this,
- An electronic terminal planning board.

The development of a “Methodology for Data Gathering” resulted in the above indicated object oriented relational database. Microsoft Access was used as a development tool and the proposed “Methodology” took the shape of a database comprising of the minimum information required to address the combined efficiency of the flows of information, cargoes, vehicles, ships and people to and from the hinterland of a port or terminal and, through this logistics node to destination. The methodology

developed is equally applicable to seaports, inland ports, rail terminals, logistics centres or freight yards. In outline, the “Methodology” consists of:

- An object orientated, relational data model in Microsoft Visio format.
- A data dictionary in HTML format, and based upon these two components,
- An object orientated, relational database using the framework of Microsoft Access without as yet specific user applications apart from those of the ‘terminal’.

## THE DATABASE

The data fields in the database encapsulate the information required to plan, organise, facilitate and manage international multimodal freight transport logistics including port and terminal operations. Thus they identify the transport characteristics of cargoes including codified data and international standards, e.g. ISO container number, tariff codes, UN country codes, EU transit references and the Unique Consignment Reference (UCR) number each shipment requires. Data fields contain information that enables the user to delineate the layout of ports and terminals including container stacks and marshalling areas. Information contained in the data fields also identifies and describes, cargo handling equipment, vehicles and routes used within the port or terminal as well as particulars and messages passed as well as manpower requirements pertaining the working practices of the terminal / port of interest. Particular attention was given to the EU sponsored projects of BALTPORTS-IT, SPHERE, TRIM, INFOLOG, BOBCOM, EUROBORDER, D2D, THEMIS, INTRA-SEAS and INTERPORT. Apart from the above listed ones, 24 FP4 and six FP5 projects were examined with emphasis on freight transport related data and data models used. The ensuing information formed the basis for this work and the resulting data model gives a comprehensive, object-oriented view of international, multimodal transport logistics processes as well as port / terminal operations. It offers a database independent description of what should be stored and how to store it. Making the data model comprehensive and generic enables it to act as the basis for the Electronic Planning Board application that will facilitate terminal operations.

The possibility of maintaining compatibility with other “user applications” such as a simulator, e.g. the simulator initiated as part of the SPHERE project and further developed during the BALTPORTS-IT project come to mind.

Figure 1 below offers an appreciation of the data model.

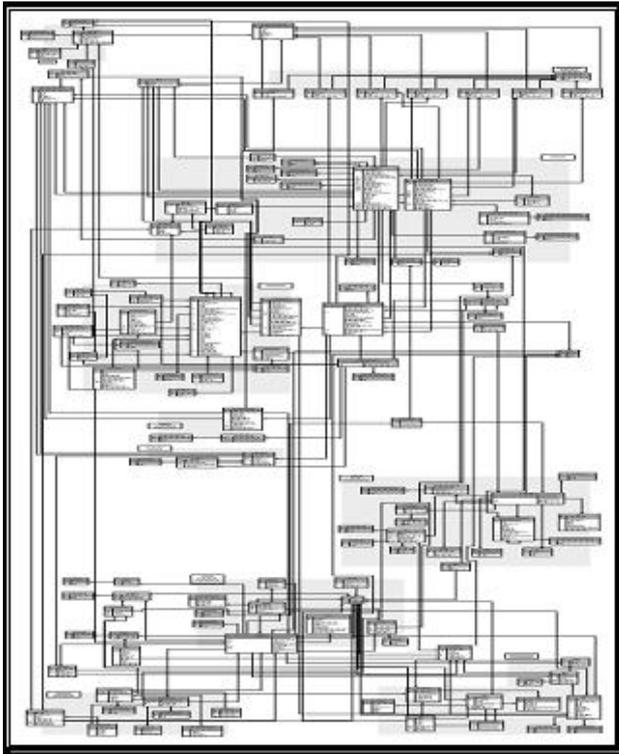


Figure 1: An Impression of the completed "Object Oriented Relational" Data Model

## THE TERMINAL PLANNING BOARD

The objective of WP4 required the development of:

- A computer based methodology in the form of an 'electronic terminal planning board' (TPB) comprising of a set of building blocks depicting:
  - Plan view of the maritime access, the berth, the apron, the container yard, the gate and the land access to the terminal – road, rail and inland waterways.
  - A selection of cargo handling equipment, container stacks, ships and road vehicles.
  - Items of information that trigger activities.
  - Links and nodes representing the movements of, e.g. cargo, vehicles, information and people through the terminal.

In all, a visual means to use common sense as well as offer the possibility of simulation to solve operational, equipment and layout problems of small to medium sized ports and terminals.

The TPB's design comprises of six modules. They were designed, programmed, tested, peer reviewed and validated as standalones. However, when combined they form the complete Terminal Planning Board as shown in Figure 2. Moving from left to right, the normal sequence of events, as they occur on a terminal when engaged in exporting, was followed. The reverse is true for importing.

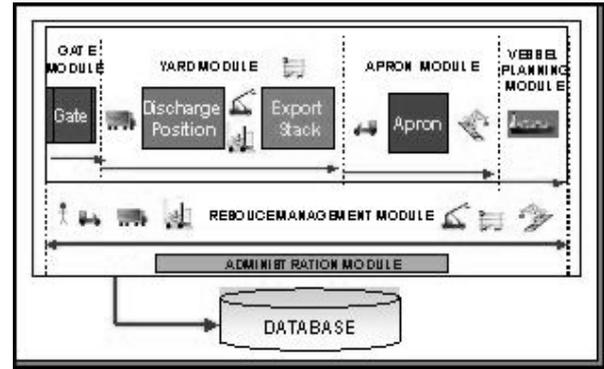


Figure 2: The Modular Program Structure of TPB

All modules depicted in Figure 2 follow the 3-Tier Distributed Approach by implementing the Client/Server communications model in Fat Client Configuration as Figure 3 below outlines.

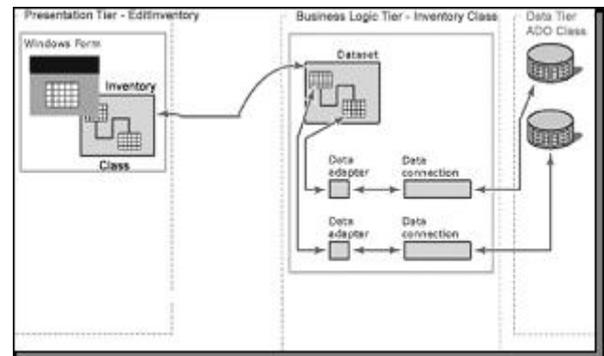


Figure 3: The 3-Tier Design

This distributed architecture allows the Terminal Planning Board application to run on multiple computers whilst the database resides on the main server. The benefits of this approach are:

- Events happen concurrently, thus more can get done.
- By replicating functionality, a more robust service can be provided.
- Users and other actors in different locations can readily be served.
- Not all the functional nodes need to be operational at once.
- Diverse systems may be connected.

At the design stage, the TPB has been deliberately given the added potential capability for it to be used as a means of communication for the port community. This makes the TPB a business application in which the various steps in the business processes can be located at the most efficient places in a network of computers. In the typical transaction using the 3-tier model, user interface processing is done in the PC at the user's location, business processing is done in a remote computer, and database access and processing is done in yet another remote computer providing centralized access.

Figure 2 shows that the backbone of the structure of the Terminal Planning Board is the object oriented relational database. To date, this database comprises of some 1,300 entities residing in some 130 tables. In an application that is supported by a large database, simultaneous access by different users and from different computers as indicated, correctness of data must be ensured.

To eventually be useful in every-day operation, the prototype of the TPB has been designed with three purposes in mind not all of which have as yet been implemented due to resource and time limitations. However, to draw attention to the latent potential of the TPB, these three purposes are:

**The Planning Mode:** In this case information gathered from identified sources relates to terminal layout, cargo handling equipment and vehicles. Data can be entered, stored in and retrieved from the database. This information can be superimposed onto a map or engineering drawing showing a plan view of the terminal to depict a current or planned layout of the terminal of interest. This has been implemented including the required application specific graphical user interfaces (GUI) enabling the TPB to communicate with the database and is fully operational.

**The Operational Mode:** In this case the application of the TPB is biased towards the daily operation of the terminal. Again, this requires the TPB to gather data from identified sources, i.e. from freight forwarders, hauliers and other members of the port community, and store it in the database. Operations relate to gate, yard, storage, apron and discharging and loading as well as tracking and tracing activities. The GUI enabling communication between the gate and the database have been developed and are operational including the allocation of 'storage space' as opposed to 'ground slot space'. The application specific GUI enabling the members of the port community to directly communicate with the TRAPIST database do not yet exist but are mentioned and form part of the latent potential of the TPB.

**The Simulation Mode:** In this case, the simultaneous activities of the gates, the yard, the apron and the vehicles could be simulated using terminal specific distributions in conjunction with pseudo random numbers and the animated icons of, e.g. vehicles, cargo, cargo handling equipment, information and terminal staff in order to generate dynamic information of benefit in the management of the terminal. This use

of the TPB too is part of its latent potential. The link between TPB and the simulation is again the object oriented relational database.

The following may serve as examples of the latent potential once the 'operational mode' of the TPB is fully implemented:

- The central controller of the port or terminal is accessing the database to query empty stack positions.
- The in-gate keeper and the central controller are simultaneously accessing the database regarding access, drop-off positions and storage location in the export stack for containers to be delivered to the port /terminal for export under, e.g. the 24 hour rule.
- The stevedore or the central controller and the cargo handling equipment operator are simultaneously accessing the database to place, receive or confirm an order to deliver an import container to a specific loading position for road tractor trailer units or to deliver an export container to a specific storage position in the export stack or on the apron in readiness for loading.
- The freight forwarder accesses the database to transfer information concerning export cargo and / or to trace the progress of export or import cargo.

The TPB uses the extended modelling language (XML) language to pass data between the modules of the TPB, the database, and the computers of individual and concurrent users. XML is a text-based protocol. It can describe any type of data, even images. Using XML format, data can be moved easily and reliably between modules and layers in three-tier applications, between different operating systems, to and from databases and using the Internet, between different users of the port community. The application of mobile computing using web messaging in conjunction with XML make it possible for terminals and other actors of the port community to communicate easily and effectively exchanging data as events occur, i.e. in real time.

Figure 4 below portrays the TPB and offers some insight into its functionality. It shows the menu bar to control the operation of the TPB at the top of the screen. The tool bar or 'toolbox' is placed on the left hand side, and the status bar including management information are included at the bottom of the screen.

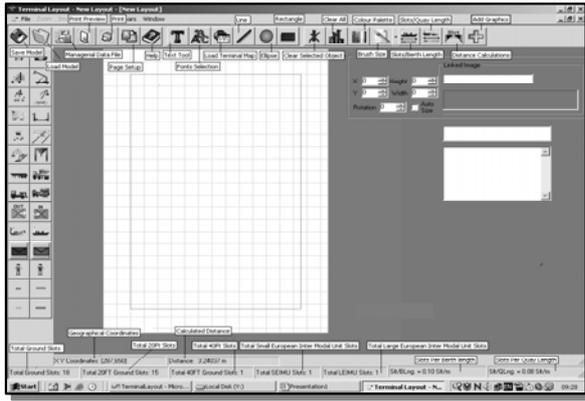


Figure 4: The Terminal Planning Board

### TESTING OF THE TERMINAL PLANNING BOARD

The objective of this RTD work implied testing the capability of the TPB by at first representing the “Status Quo” of the Tivoli Container Terminal in the Port of Cork, Republic of Ireland, and of the Port of Kokkola in Finland, as they existed at the start of the TRAPIST project in 2002. The status quo had to be conveyed by means of the physical flows of cargoes, the equipment used and the messages passed. This had to be superimposed on the plan view of the Tivoli Terminal as well as the map of the Port of Kokkola.

Apart from access, gates, ground slots, cargo handling equipment and vehicles the presentation indicates a network of paths forming links and nodes, over which cargo, equipment and information flow. This information is stored in the TRAPIST database and in conjunction with simulation (WP5) it will show traffic density, path service level, origin-destination service level and the distribution of flow over paths and thus possible bottlenecks. The objectives of Task 4.3 were achieved but involved:

- To prepare the plan view of the terminal, including maritime access, berths, apron, container yard, gate and land access, under consideration for display on the screen.
- To use the drawing tool and icons to depict paths and positions of cargo flows, equipment and associated movements, storage areas and information.

Figure 5 below shows the results related to the “Status Quo” of the Tivoli Container Terminal. It indicates the layout of TEU ground slots spaced for “2+1” straddle carrier operations. It shows the in- and out-gates, road access to the check-in area, the quay including the two berths, the ship-to-shore gantry cranes, the apron, the road access to the apron, straddle carriers and articulated vehicles delivering and / or collecting containers.

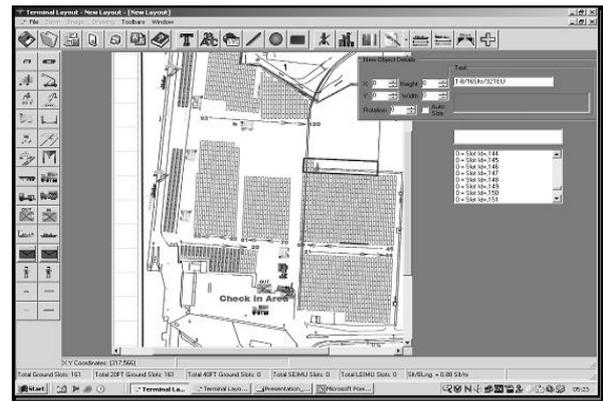


Figure 5: The “Status Quo” of the Tivoli Container Terminal in Cork – TEU only

The bottom four icons in the tool bar on the left hand side of Figure 5 represent ground slots. The top two mimic the space required by a 20 ft (TEU) on the ground of the terminal and by a 40 ft (FEU) container respectively. The bottom two icons block out the ground space needed by the European Intermodal Loading Units (EILU), proposed by the Commission. These EILU are based on the dimensions of swap bodies. They maximise the space for the stowage of pallets, something ISO containers do not do, and, as a result, these EILU have different space requirements to ISO containers on whose dimensions ground slots are traditionally based. To achieve the terminal layout shown in Figure 5, each ground slot had to be individually selected from the tool bar, entered into the database, positioned on the screen, dragged and dropped into position and aligned by rotation. The TPB uses ‘left’ mouse clicks to select, drag and drop an icon, ‘right’ mouse clicks to rotate the icon, and wheel rotation to zoom in and zoom out.

The objective of WP4 further implied that the characteristics of each link, cargo- and equipment item, ship and vehicle in a path would be identified and documented, if necessary through a time and motion study and analysed with respect to compatibility of cargo, terminal equipment, vessel, the physical layout of / along the path, and manpower requirements, i.e. the number, skill and training requirements of staff. To this end the stochastic sequences of activities, facilities, cargo handling equipment, information and manpower constituting the port / terminal processes were reviewed and documented for a container terminal, a dry bulk cargo terminal and for a break bulk cargo terminal.

The TPB discussed above appears to concentrate on container terminals. This appearance is deceiving and equal attention has been given to Ro-Ro-, dry bulk- and break-bulk terminals. The extensively documented processes were peer reviewed by the industrial partners. A fabulous wealth of information and experience was made available by them and could

be included in the documentation. Observing the flow of container handling vehicles on the container terminal in Cork led to the conclusion that there was no one-way system in operation on this terminal. The movements of straddle carriers in the container yard, on the apron and on the unloading and loading areas for articulated road vehicles near the gates were somewhat random. Cargo, cargo handling equipment, terminal staff and terminal layout were found to be in harmony and working well. A subsequent time and motion study on this terminal confirmed this.

The characteristics of links, cargoes (containers), equipment items, ships and land vehicles are stored in the object oriented relational database. The user enters or selects them. The TPB is generic. Its application as presented in this paper though was specific. This necessitated that each entity was considered individually. Object orientation demanded that each entity of the TPB is 'live', can be saved, edited and reloaded for subsequent work. This implies that each, e.g. line representing a path or a segment of a path, cargo item, piece of equipment or vehicle has to be stored together with its characteristics. The icons in the tool bar are not only pictures but objects. They are associated with the characteristics of the real life object they depict. Rather than use a fixed set of data for this, a facility for the user to enter this data, as befits the terminal under consideration, has been provided. The database has been designed to take account of this. The graphical user interfaces to enter characteristics of cargo handling equipment are demonstrated in Figure 6 and 7 below with respect to a straddle carrier and a 20 ft container.

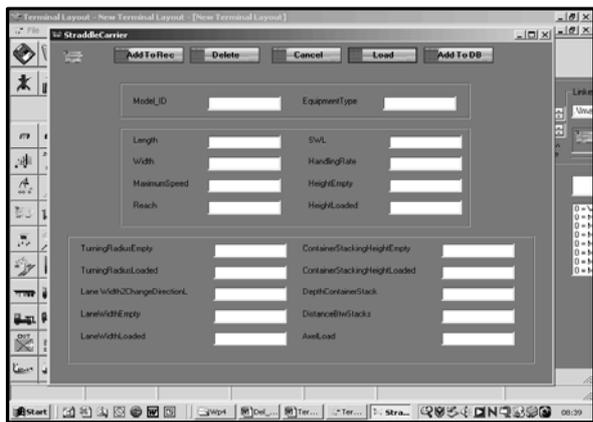


Figure 6: The GUI for Terminal specific Equipment

The Information required for Terminal Layout for 40 ft Containers (2 TEU Ground Slots) using Straddle Carriers is listed below and may serve as example.

- SWL 40 tonne
- Handling rate 20 / h
- Length 9.2 m
- Width 4.9 m
- Height empty 11.8 m
- Height loaded 11.8 m
- Container stacking height loaded 3
- Container stacking height empty 3
- Distance between stacks 2 m
- Depth of container stack (1 row) 2.5 m
- Turning radius loaded 10.5 m
- Turning radius empty 9.5 m
- Lane width to change direction
- – loaded 10.5 m
- Lane width – loaded 7.0 m
- Lane width – empty 7.0 m
- Axle load 24 tonne
- Maximum travelling speed laden 24 km/h

The same approach of combining TPB and database has been used to determine, enter and store the characteristics of terminal entities required for operational as well as layout planning. Figure 7 below illustrates this for containers.

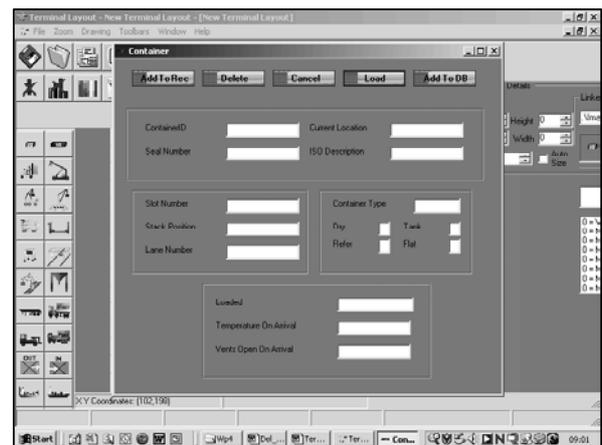


Figure 7: The Graphical User Interface for a 20-Foot Container

Slot number, stack position and lane number specify where on the terminal the container is stored. The information helps with tracking and tracing of the box. The same GUI as shown in Figure 7 above and the same comments made, apply to the 40 foot container. The arrangement of ground slots on the terminal is generally made in terms of "Twenty Foot Equivalent Units" (TEU) with two of these making up a ground slot required for a 40 ft box.

The information of all of the icons, together with that related to lines and nodes, once stored in the object oriented relational database constitute the data required for simulation.

After establishing and verifying the display and correctness of the “Status Quo” by means of peer reviews and terminal visits, and engaging in similar activities with respect to terminal processes and characteristics of, e.g. cargoes, cargo handling equipment and vehicles, the objective of WP4 demanded that: “The building blocks or entities of the TPB could be rearranged or changed within the confines of the plan view of the terminal and that options regarding the selection of cargo handling equipment, terminal vehicles, manpower requirements, terminal layout as well as information flows, could be explored with the aim to find not only a more efficient way of operating but also a better combination of terminal layout and operation”. Figure 8 below gives an indication of this work.

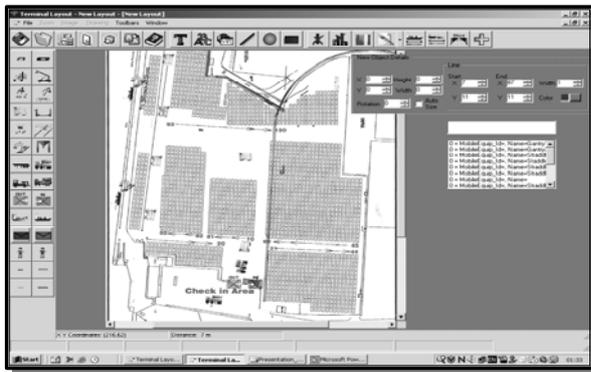


Figure 8: Phase I of the Development at the Tivoli Container Terminal, Cork

The Tivoli Container Terminal underwent considerable development and expansion during the two years of TRAPIST. This was done in two phases. Figures 8 above and Figure 9 below show these developments. They were produced in the same way as Figure 5 and demonstrate the capability of the TPB to take account of changes in terminal layout.

In all cases the TEU ground slots were aligned East-West and about parallel to the quay. This resulted in some access and congestion problems for straddle carriers in the North of the terminal. The solution was found by realigning the ground slots of this area North-South.

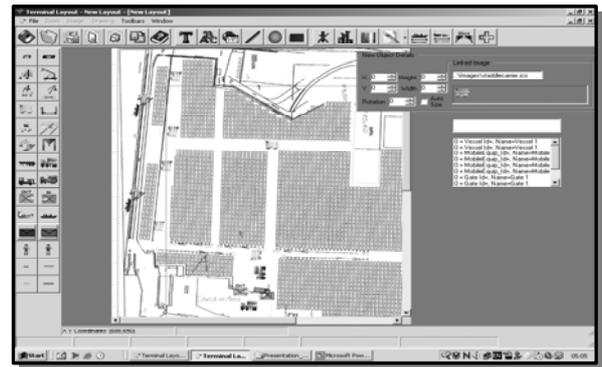


Figure 9: Phase II of the Development at the Tivoli Container Terminal, Cork

Figure 10 depicts this common sense solution to the layout problem of the latest development phase of ‘Tivoli’ without negatively affecting the terminal’s capacity.

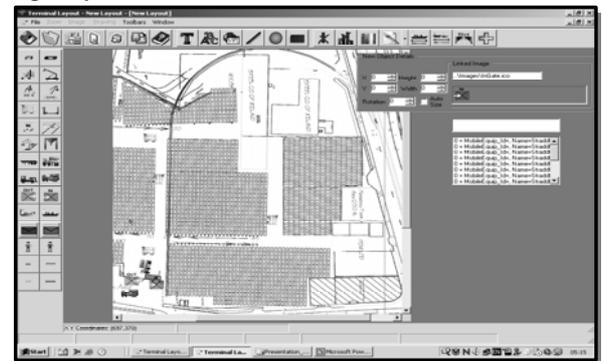


Figure 10: Alternate Layout of Ground Slots for the 2<sup>nd</sup> Development Phase of ‘Tivoli’

With respect to the Port of Kokkola, the problem was related to a new road to be built to connect the ‘General Port’ with the ‘Deep Port’. The old and existing road connection was too long, too narrow and had too many bends and was difficult to negotiate for the large and heavy vehicles transporting zinc ingots from the warehouses of the factory situated in the vicinity of the Deep Port to the General Port for loading into vessels bound for Amsterdam. Figure 11 below shows the layout of this port and the connecting new road to be built, thereby halving the travelling distance.

The capability of the TPB to develop alternate layouts within the confines of an existing terminal layout was successfully tested and industrially applied by selecting two considerably different ports. The objective of WP4 of TRAPIST has been well achieved.

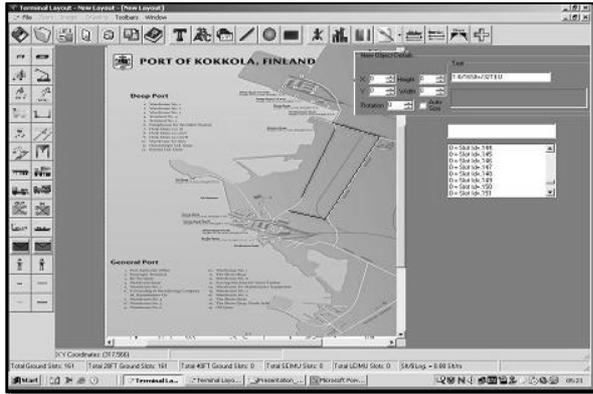


Figure 11: New Road (in red) connecting the General Port with the Deep Port

## BENEFITS

These accrue to terminal and ship operators and to a similar extent to hauliers delivering and / or collecting containers to / from the terminal. Regarding exporters, importers and freight forwarders, an efficient port / terminal is a commercially attractive proposition and generates more business. Improving the efficiency of a container terminal, or of any other terminal for that matter, by increasing the cargo throughput, reduces the turn-around time of vessels and other vehicles as well as the unproductive times on the terminal. A benefit-cost analysis was carried out as part of Workpackage 6 of TRAPIST. It was concerned with “Optimal Loading / Discharging of Liner / Feeder Vessels” and based on information prepared by MARAN, NL. Benefits were estimated as 515 Euro for every hour gained for a 750 TEU vessel. With respect to a small to medium sized terminal and assuming operating expenses per year to be 6,180,000 Euro, assuming further a 52 week year, six days / week and 10 working hours per day, terminal / port benefit of 1980 Euro for every hour gained through increased terminal / port efficiency were estimated.

## CONCLUSIONS

This paper puts the EU sponsored RTD work done in WP4 of TRAPIST into a European perspective. It examines critically the overall objectives and achievements of the WP. These are:

- An object oriented relational database and associated GUI, and
- A Terminal Planning Board.

It has been shown that the prototype system referred to as TPB in conjunction with the object oriented relational database is capable of displaying existing and alternative terminal layouts. It has been demonstrated that it can be used to depict the status

quo and from this starting point develop alternate terminal layouts. The ‘Tivoli’ Container Terminal of the Port of Cork has been used for this purpose. Subsequently, the need expressed by the Port Authority of the Port of Kokkola for a new road connecting the general Port with the Deep Port was used to prove the applicability of the TPB to the planning of this task.

The combination of TPB and database enables:

- Re-engineering of terminal layout, selection of cargo handling equipment and working methods to reduce manpower, travelling distance and or -time by using different routes, equipment and arrangements of slot spaces to increase terminal capacity, terminal throughput and terminal efficiency.
- Making use of the ‘common sense’ approach by providing the terminal operator with a tool enabling him to translate his operational experience into the layout of slot spaces, positioning of gates, terminal access, and the selection of cargo handling equipment. Entities of the same equipment type but with different operating characteristics can be stored in the database and used.
- Populating the database with the terminal entities and their characteristics yields data for the simulation system, addresses access issues both for land and maritime access, superimposing and displaying the links over the map of the terminal enables network analysis, and last but not least, the operational characteristics entered into the database aid benchmarking.

In summary, the TPB affords:

- Potential planning opportunities
- Obtaining of real time information through the operational aspects of the TPB in association with gate procedures
- Operationally feasible results
- Helps maximising the use of existing facilities
- Supports the efficient layout of new facilities
- Is user friendly to industry
- Does not need extensive and specialised support of IT personnel.
- Forms a direct link to selected simulation programs.

## REFERENCES

Augoustatos, D.; Papageorgiou, K.; and Schmidt, F.A.; 1997, “*SPHERE D3: Report on Modelling of SMP Operation*, Contract No. WA-95-SC.224, October.

BMT-TK; “*D2D Development of tools and IT databases for Intermodal Freight Transport Chains*.”

BOPCOM; 1996, “Baltic/Basic Open Port Communication”.

COM(97) 243; 1997,  
<http://europa.eu.int/scadplus/leg/en/lvb/124179.htm>

DETR, 2000, “European Spatial Planning and Urban-Rural Relationships: The UK Dimension”, June

DGTREN; 1999, ‘Bottlenecks in Door-to-Door Short Sea Shipping’, Fiche N015, DGTREN, 1999, chapter 9.1, April.

ENYCA; 1996, “EUROBORDER Deliverable D4.1: Establishment of a functional port terminal model”, Contract No. WA-95-SC.153, April.

European Commission, 1998, COM (1998) 716.

European Commission, 1999, “COM99 317 en final Development of SSS in Europe”.

European Commission; 1992, Communication on the Future Development of the Common Transport Policy (COM(92)0494)

European Commission; 2001, EU White Paper: “Time to Decide”, September.

European Commission; 2003, COM(2003)155; 2003, ‘Proposal for a Directive on Intermodal Loading Units’

INFOLOG, 1999, “Intermodal Information Link for Improved Logistics”, EU, Project Reference : WA-97-SC.2173.

INTERREG II, 2002, North European Trade Axes Development (NETA).

INTRA-SEAS; 1999,  
[www.cordis.lu/transport/src/intrasea.htm](http://www.cordis.lu/transport/src/intrasea.htm)

Schmidt, F. A. TRAPIST WP03, 2003, “ Optimising Information and Physical Flows in a Port/Terminal”.

Schmidt, F.A.; 2004, “Development of “Routines” to Evaluate the Combination of “Terminal and Ship” including the Conceptual Designs for Ships facilitating Loading and Discharging of Unitised Cargoes at under-used Dry Bulk Terminals”, TRAPIST, WP8, D8.5.

Schmidt, F.A.; 2004, “New Concepts and Designs for Ships facilitating loading and discharging unitised Cargoes at under-used Dry Bulk Terminals”, TRAPIST, WP8, D8.2.

Schmidt, F.A.; and Yazdani, R.; 2003, “The Terminal Planning Board User Guide”.

Schmidt, F.A.; Yazdani, R.; and Young, R.; 2003, “Development of Routines for Gathering Data from Identified Sources”, TRAPIST, WP4, and D4.1.

TJ; 2001, “TRIM Transport Reference Information Model”, June.

US Customs; 2003, “Container Security Initiative”, Trade Facilitation Conference, March 21<sup>st</sup>, Irish Exporters Association, Dublin.  
[www.tfk-hamburg.com/en/projects/](http://www.tfk-hamburg.com/en/projects/)

Yazdani, R.; and Schmidt F. A.; 2003, *Port Processes Simulator “BALTPORTS-IT D4.1: User Guide*, Contract No. ISD-2001-33030, February.

## AUTHOR BIOGRAPHIES

**Dr F A Schmidt (Captain), BSc (NAUT), PhD (Naval. Arch.), Master Mariner (FG), MNI, MRINA, FIFP.**

Main interests are international intermodal freight transport logistics, port development and operation, ship design, ship operation and maritime transport. He worked as shipmaster and in preceding ranks, as ship manager and in academia. He lead several workpackages in the EU and UK Government sponsored projects of SPHERE, BALTPORTS-IT, TRAPIST, NETA, ITeLS and the ‘Integrated Design Project’ as part of the UK Efficient Ship project and now ELOGMAR-M. He acted as consultant to, e.g US Navy Sealift Command, the International Labour Office, the New Northern Ireland Assembly, and the Northern Ireland Branch of the Confederation of British Industry, the Irish Business and Employers Confederation, the Irish Exporters Association and the Irish Maritime Development Office. He founded M&SS in February 2003 and currently serves as managing director of this company.

**Miss Rahila Yazdani, MSc (Physics), MCS, AMBCS.**

Main interests are software engineering and systems development with particular emphasis on simulation and visualisation. She considerably furthered the Port Process Simulator initiated in SPHERE during the BALTPORTS-IT project and developed the systems and software required for the Terminal Planning Board during the TRAPIST project. She works in industry as a systems analyst and software engineer and completes her PhD studies part-time. She can be contacted using [rahilayazdani@hotmail.com](mailto:rahilayazdani@hotmail.com).

# MODEL-BASED ESSENTIAL LOGISTICS PRINCIPLES FOR CREATING A WEB-PORTAL OF TRANSPORT SERVICES' CONSUMERS

Eberhard Bluemel  
Fraunhofer Institute FhG/IFF  
Sandtorstrasse 22, 39106 Magdeburg,  
Germany  
E-mail:  
[Eberhard.Bluemel@iff.fraunhofer.de](mailto:Eberhard.Bluemel@iff.fraunhofer.de)

Svetlana Vinichenko  
Logitrans Consult Ltd.  
Kellamae 18-34, 74113 Maardu,  
Estonia  
E-mail:  
[svetlana@logitrans.ee](mailto:svetlana@logitrans.ee)

Leonid Novickis  
Riga Technical university  
Kalku str.1, LV-1658 Riga,  
Latvia  
E-mail:  
[idc@balva.lv](mailto:idc@balva.lv)

## KEYWORDS

Transport logistics, Modelling, Web-based solutions.

## ABSTRACT

Logistics platform for development of Web-portal for transport services' consumers is described in this paper: the maritime freight route and target groups of actors operating along it are identified, logistics model as a set of business charts, communication diagrams and flowcharts is described.

## INTRODUCTION

Functioning of the market economy depends on the effective information provision of the transport services. It is really important to reach synchronization of business processes, cargo and data flows and integrity among different activities accompanying cargo transportation along the selected freight route. In 2004 the eLOGMAR-M project (Web-based and Mobile Solutions for Collaborative Work Environment with Logistics and Maritime Applications), funded by the Commission of the European Communities within the Sixth Framework Programme (DG INFSO), has been strated. One of the goals of eLOGMAR-M project is to create a Web-portal for information providing to transport services' consumers. The major idea from logistics point of view is to estimate a start-to-finish rate of cargo transportation and to select the most suitable supply chain.

## THE MARITIME FREIGHT ROUTE AND TARGET GROUPS

The maritime freight route "Baltic Sea feeder ports - Western Europe hub port (Hamburg) - Mediterranean port (Thessaloniki) - Chinese ports" is selected as the subject of investigation and demonstration. The rapidly developing trade between Europe and Asia, the polarisation of producers in Asia and of consumers in

Europe needs the improvement of supporting services along this transportation routes. Containerships present one half of the turnover measured in gross tonnage along the route "East Asia ↔ North-Western Europe", with China being the largest producer of container traffic originating in Asian countries. Multilevel logistics and transport business process will be analysed at three levels (Figure 1): First level – containers delivery from one deep sea port (hub) to another by deep sea shipping lines, Second level – containers delivery from hub port to smaller feeder ports by feeder shipping lines, Third level – containers delivery from feeder ports to customers by trains.

The Baltic Sea Region is selected as the sample of feeder shipping and Port of Hamburg as container hub for this region. IT- and Communication support of intermodal transportation is illustrated by logistics supply chain "Baltic feeder ports ↔ trains ↔ Customers".

Special attention will be also devoted to the study of maritime and logistics processes in the Baltic Sea Region as they form an integral part of the above mentioned freight route. This region has specific requirements: after the candidate countries (Lithuania, Latvia and Estonia) joined both the EU and NATO, they play the role of a new border between EU and CIS.

Actors from two major target groups are involved in cargo transportation process:

- 1) **Transportation group (K1):** deep sea and feeder shipping lines, shipowners, terminal operators, block train operators, forwarding companies, multimodal transportation operators, freight brokers;
- 2) **Transportation group (K2):** cargo owners, forwarding companies, traders.

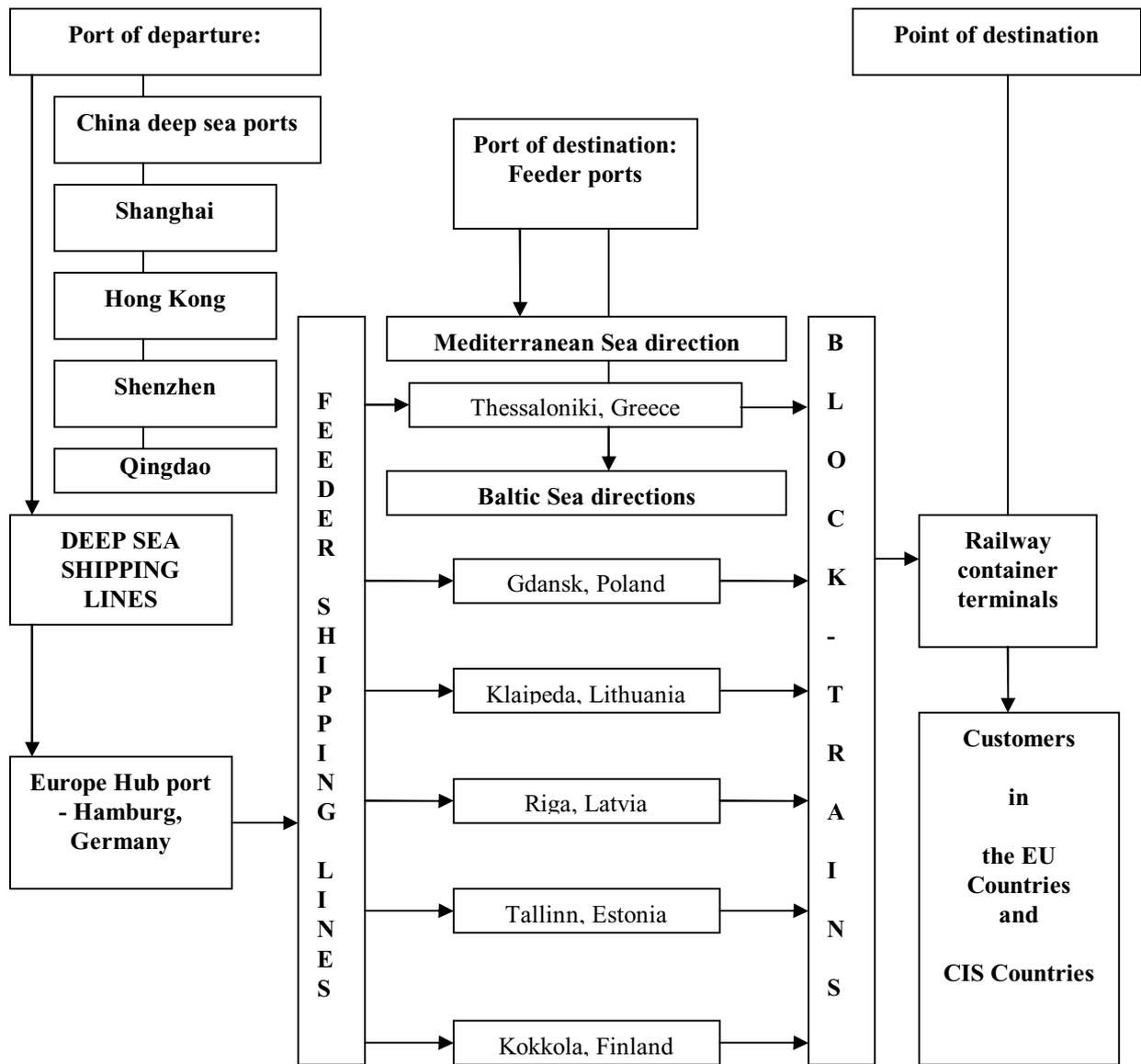


Figure 1: Multilevel Logistics and Transport Business Process.

### ANALYSIS AND MODELLING OF BUSINESS PROCESSES

A set of business charts and communication diagrams from BSP method (Bluemel et al. 2003) and LIS Technology (Ginters et. al. 2002) is used for presentation of business processes of collaboration between partners from two major groups (transportation and cargo groups), which are involved

in logistics processes of freight transportation along the selected route. The BSP method is used to produce the set of business charts (“Processes – Actions” and “Actions – Executors”). Table 1 presents three main processes and appropriate actions of their detailed definition. Table 2 presents business actions sharing among classes of executors (two major groups: cargo group and transportation group).

Table 1: Business Chart “Processes – Actions”

Processes	Actions
1. Preparation of Information regarding services of actors from transportation group	1.1. Preparation of information regarding sailing schedules (deep sea shipping lines)
	1.2. Preparation of information regarding sailing schedules (feeder lines)
	1.3. Preparation of information regarding characteristics of port terminals
	1.4. Preparation of information regarding intermodal (block train) services
	1.5. Preparation of information regarding characteristics of rail terminals
2. Promotion of services and information distribution among actors from cargo group	2.1. Printing and distribution of leaflets and handbooks
	2.2. Presentation at conferences and exhibitions
	2.3. Meetings arrangement with potential customers
	2.4. Providing access to information via Internet
3. Looking for suitable actors from transportation group for cargo delivery in accordance with terms and conditions of a contract	3.1. Study and analysis of information presented by actors from transportation group in different ways (handbooks, Web-sites etc.)
	3.2. Preparation and distribution of inquiries regarding conditions of cargo transportation
	3.3. Study and analysis of replies to inquiries received from transportation group
	3.4. Making a decision concerning the selection of partners from transportation group

Table 2: Business Chart “Actions – Executors”

Executors	Actors from K1	Actors from K2
1.1. Preparation of information regarding sailing schedules (deep sea shipping lines)		
1.2. Preparation of information regarding sailing schedules (feeder lines)		
1.3. Preparation of information regarding characteristics of port terminals		
1.4. Preparation of information regarding intermodal (block train) services		
1.5. Preparation of information regarding characteristics of rail terminals		
2.1. Printing and distribution of leaflets and handbooks		
2.2. Presentation at conferences and exhibitions		
2.3. Meetings arrangement with potential customers		
2.4. Providing access to information via Internet		
3.1. Study and analysis of information presented by actors from transportation group in different ways (handbooks, Web-sites etc.)		
3.2. Preparation and distribution of inquiries regarding conditions of cargo transportation		
3.3. Study and analysis of replies to inquiries received from transportation group		
3.4. Making a decision concerning the selection of partners from transportation group		

Note: K1 – Transportation group; K2 – Cargo group.

Communication diagrams present a model of information objects and links between them: three levels model is presented in Figures 2, 3 and 4.

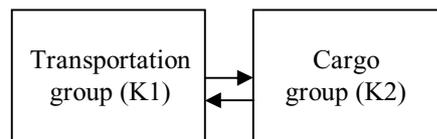


Figure 2: First Level.

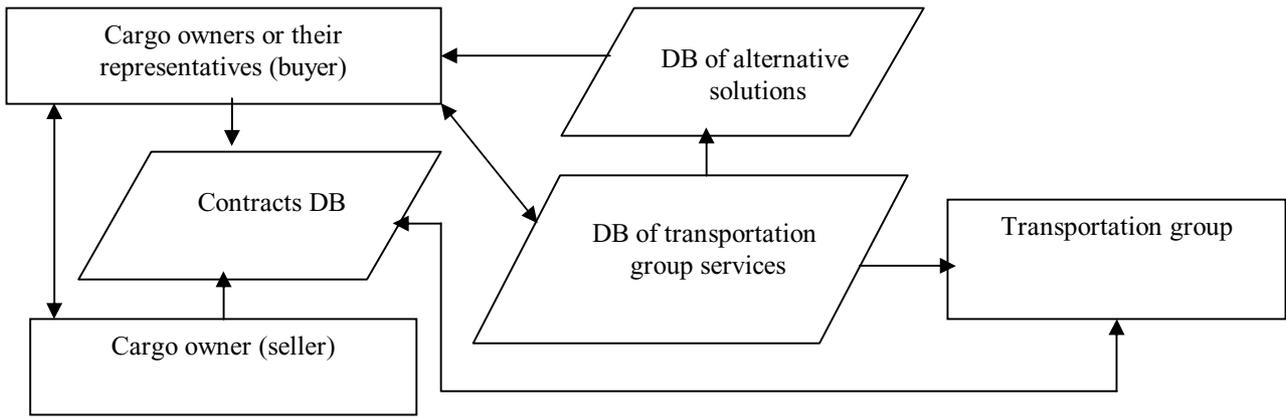


Figure 3: Second Level.

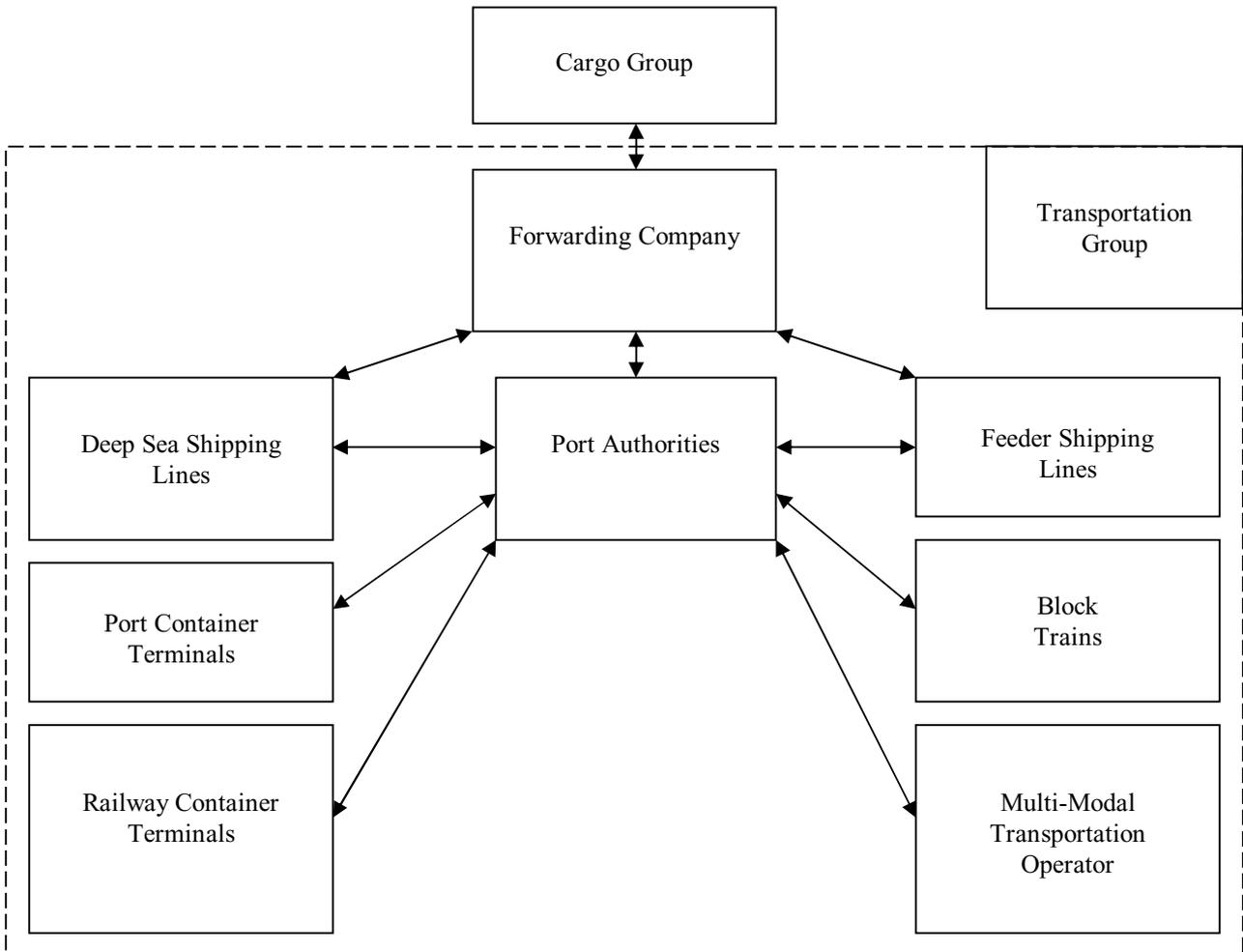


Figure 4: Third Level.

The following business processes will be supported by Web-portal:

- 1) Cargo group – preparation of contract of Sale/Purchase;
- 2) Cargo group – looking for suitable actors from cargo transportation group in accordance with terms and conditions of a contract of Sale/Purchase;
- 3) Transportation group – promotion of services by the way of distribution of information via Internet;

- 4) Transportation group – preparation of initial information regarding services and its maintenance.

#### COMPONENTS OF WEB-PORTAL IN TRANSPORT LOGISTICS

Each actor, who is involved in business processes of cargo transportation along the selected freight route, has a freedom to maintain the content of its part taking

into consideration the specifics of the transport services. However, the portal will have a core of components and each of the actors has to contribute to its maintenance. This core of components contains:

- General information (overview of IT and Communication Solutions with transport logistics and maritime applications, overview of EU regulations, international standards and national laws, overview of opportunities and computer-based courses for training);
- Information regarding solutions in transport logistics (initial information and algorithm of decision making). Initial information contains: sailing schedules of ocean and feeder lines, schedules of block-train services, characteristics of port and railway terminals, terms of freight delivery and transportation etc.).

A Web-portal is aimed at supporting of decision making in transport logistics, using business processes models, general and initial information. Both target groups will benefit from Web-portal applications:

- Actors from transportation group by promotion of their services and increasing cargoes flows;
- Actors from cargo group by calculating and estimation a start-to-finish transportation tariffs and choosing better cargo carrier, who meets the required criteria.

**FLOWCHART OF DECISION MAKING**

Flowchart of decision making is presented in Figure 5.

As it was mentioned above, two groups of actors are involved in solving logistics tasks: **Group K1** – transportation group; **Group K2** –cargo group.

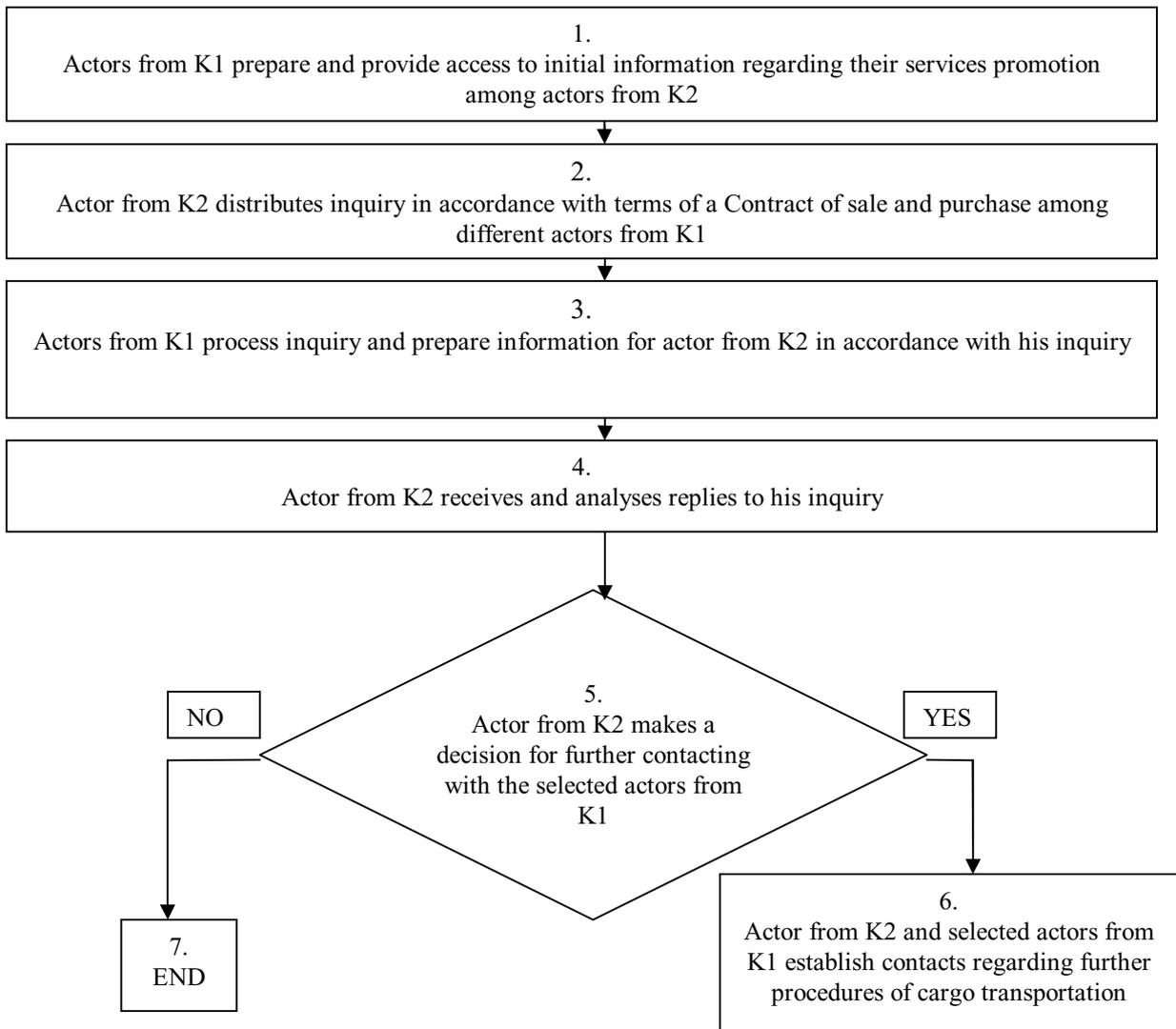


Figure 5: Flowchart Showing Processes of Decision Making.

**Block 1:** For promotion of their services actors from transportation group must present information regarding their activities.

**Block 2:** Actors from cargo group prepare their inquiries in accordance with terms of a contract and distribute them among actors from transportation group using results of marketing analysis.

**Block 3:** Actors from transportation group study inquiries from cargo owners and prepare necessary information, which is based on transportation tariffs and delivery times.

**Block 4 & Block 5:** After receiving reply representative of cargo group studies it, compare with alternative proposals and make decision for further contacting with selected actors from transportation group.

**Block 6:** Different communication and organisational means are used for further establishing and keeping contacts between actors from two major groups to support cargo transportation process.

## CONCLUSIONS

1. Application of business charts and communication diagrams for modelling and presentation of transport logistics processes along the selected freight route is illustrated.
2. The major logistics principles of creating a Web-portal in transport logistics are described.
3. The brief description of the structure of Web-portal is presented.
4. Some of results could be further also used in creating a dictionary in logistics information systems within LOGIS-Mobile project "Competence Framework for Mobile On-Site Accelerated Vocational Training in Logistics

Information Systems" (Leonardo da Vinci Programme).

## ACKNOWLEDGEMENT

The presented activity is supported by the eLOGMAR-M project funded under the IST Six Framework Programme of the European Commission.

## REFERENCES

- Bluemel, E. et al. 2003. "Applications of Simulation and IT-solutions in the Baltic Port Areas of the Associated Candidate Countries". JUMI, Riga.
- Ginters, E. et al. 2002. "Logistics Information Systems. Part 1". JUMI, Riga.

## AUTHOR BIOGRAPHIES

**EBERHARD BLUEMEL** is Director of the Division of Virtual Development and Training (VDT) at the Fraunhofer IFF in Magdeburg. He holds a Ph.D. in Natural Sciences from Otto-von-Guericke University in Magdeburg. His research fields include Virtual Reality, Computer-Based Training, Simulation.

**LEONID NOVICKIS** is Head of the Division of Applied Systems Software at the Riga Technical University. He holds a Ph.D. from Riga Technical University (1980) and Dr. Habil. Sc.ing. from Latvian Academy of Science (1990). His research interests include software engineering, computer science, design and development of applied software systems.

**SVETLANA VINICHENKO** is Director of "Logitrans Consult Ltd." (Estonia). She graduated from Odessa Marine Transport Institute. Her research interests include transport logistics and business process modelling.

# MODELS FOR SUPPORT MARITIME LOGISTICS: A CASE STUDY FOR IMPROVING TERMINAL PLANNING

Chiara Briano, Enrico Briano, Agostino G. Bruzzone

Liophant Simulation  
Savona Campus - Genoa University  
Ex-Caserma Bligny - via Molinero  
17100 Savona, Italy

Roberto Revetria  
Centro Italiano di Eccellenza per la Logistica Integrata  
Via Balbi, 22/7  
16100 Genova GE, Italy

## KEYWORDS

Simulation, Berth Management, Container Yards, Optimization, Linear Programming

## ABSTRACT

Container terminals world wide are trying to expand capacity and increase performance at a minimum of investments. Often the container terminal operations are changing to meet increased customer demands as well as to adapt to new technologies. The increasing costs of container terminal development do justify the use of computer simulation to assist in planning and policy making. The use of specific mathematical models for supporting yard operations optimization is turning to be the key of the next generation Decision Support Systems, the paper outline the integration between a flexible simulator which represents the marine-side operations of a container terminal with a Linear Programming model for improving berth assignment management policies and yard stacking management policies.

## INTRODUCTION

In the last years maritime transport shown an outstanding growth, causing to some harbors to reach their maximum capacity limits and so simulation gain, consequently, interest as primary methodology to proper address the upcoming years challenges..

Another important aspect is the radical changing in the Supply Chain where Europe and the Western Countries are going to change their position from mostly exporting to importing area due to the growing in outsourcing towards Far East countries like China and India.

This change involves also the logistics structure of the harbors moving from, basically, handling and storage facilities towards an Integrated Multimodal Terminals in which the goods are sorted, routed and, eventually, transformed to finished goods. The challenge of increase the port traffic avoiding to paralyze the harbor itself and its accessing ways is turning to be the critical point of the new development strategies for the

upcoming years.

In the past modeling & simulation was used as base methodology for improving port operations, yard planning and resource control in such way several experiences can be capitalized and new approaches can be designed based on it .

Several simulation projects were developed, in fact, to optimize the scheduling policies of the terminal in order to reduce the waiting times and improve the overall performances. In such field techniques used ranging from discrete event models in order to study the advantages of an automated container terminal ( Delft University in Netherlands), to the use of a specific model for simulating sea conditions and their effect on the maritime logistics (University of Ulster) up to Petri Networks (University of Barcelona, Spain).

Some recent studies made by the Dong-A University of Korea demonstrated also the advantages of an ERP-like approach in container handling for reducing the system development time, eliminating redundancies, maintaining integration and enabling flexibility, standardization of workflow.

Starting from the authors experiences in design and implement Decision Support System (DSS) devoted to optimize logistics of maritime transportation the authors propose an innovative approach based on the integration among Discrete Event Simulation and Linear Programming to improve the effectiveness of the yard schedule by validating traffic forecasts with credible simulation scenario. Genoa Port, particularly, starting from an high percentage of empty containers (i.e. about 35% in 2003 versus the 20% of Rotterdam Port) can be regarded as the proper case study for the proposed methodology.

The idea and the goal of this paper is to optimize the sequence of loading and unloading operations focusing on the wharfs' occupation considering all the factors, controllable and not controllable, involved in the berthing process in order to reduce inefficiency and improve the potentials of the terminal.

The factors involved in this approach are the stochastic process of the incoming ships, sea conditions, and the consequent extra costs due to additional wharf occupation for improper dock schedule. Constrained

resources (i.e. crane drivers, loaders, berth operators, cranes, stackers, transtainers, forklifts, tugboats and container yard) were considered in the proposed study.

## **SIMULATION FOR MANAGING TERMINALS AND PORTS**

Simulation in maritime environments has been developed by the scientists for a long time, in fact both the authors themselves and other researchers had studied technologies devoted to improve performances inside an harbour and on the infrastructures around it (railroads, motorways, etc.).

The authors took care of this aspects since 1996, when the first project regarding the use of Virtual Reality devoted to train Service Operators inside Genoa Port and then to improve safety, called "Safety First", was implemented by Liophant Simulation, after that, they continued to use simulation in order to improve port operations; in fact in 2002, during the TransBaltica 2002 Conference, POSEIDON simulator was implemented in order to study the behavior of an Oil Terminal in Genoa at the arriving of several types of tugs pilots and boats and to manage the docks and its resources in order to reach the goal of the best performance.

Other important works, developed principally in the years 2002-2003, at the 18th International Port Conference of Alexandria in Egypt in 2002, in fact, was presented an innovative work by the Dong-A University in South Korea devoted to manage Container Terminal Operating Systems using an ERP Approach, clustering for example the workflow of the container terminals, analyzing the business process to generate the best workflow and using the planning facility, coupled with data flow from client entities such as shipping companies.

In 2002 was developed the TOMAS project, a tool for Object-Oriented Modeling and Simulation created two years before by the researchers of the Delft University in Netherlands, applied in an automated Container Terminal, that guarantees the advantages of the stand-alone complex model divided into small sub models easy to understand and an improved transparency and maintainability of the simulation model.

During the International Conference HMS & MAS 2002, held in Bergeggi, Italy, the University of Barcelona, Spain, presented a modelling methodology based on Coloured Petri Nets (CPN) formalism to design a software tool useful in port process simulation in order to make easier the design and the management of port terminals.

The next year (2003) in Alexandria, Egypt was presented a system of PDGPS (Precision Differential Global Position System), applied in the Gantry Cranes in Alexandria Port, in order to increase and improve the Yard Productivity to match today's Ship To Shore Crane Productivity tracking the Containers and using Auto Steering tools for cranes.

Another application, called SIRIO; was presented as a DSS (Decision Support System) in harbor operations using Web Based Distributed Simulation in a Java, based architecture able to share quantitative models

among Internet and Intranet for helping decision makers using Autonomous Agents (AA).

The researchers of the Limburg University Central in Belgium, instead, developed an heuristic algorithm in order to find the best sequence of placing vessels inside a lock for harbours set in a river or in a canal, which are sometimes subjected to tides and other bad conditions.

Always in 2003 were presented works about the production of a software helping to manage port operations, one of them is the "Port Process Simulator" (PPS) developed by the Ulster University and already installed in three ports of the Baltic Area, which simplify all the harbouring activities.

In the same conference, HMS 2003 in Riga, Latvia, a real application of this software was presented for the Gdansk port, in Poland, and a possible scenario of further developments was shown, after a phase of VV&A (Verification, Validation & Analysis), by the simulator fed by the real data.

So simulation in harbours, both maritime and fluvial, it is one of the most active fields of the research in modeling and simulation.

## **THE IMPLEMENTED SIMULATION MODEL**

The idea of this work is to find a strong solution even not necessarily the best performing for managing the ships arrivals timetable inside an Italian port container terminal.

The estimated time of arrivals of every single vessel is scheduled inside a DB (Database) calculating the "Delta" of the hypothetical arrival time versus the current time.

The real time of arriving is calculated taking into account several coefficients related to different events and using a Monte Carlo methodology; the most important factors are the quality of the data, the sea condition coefficient, that is bigger when the condition of the sea are worse and inflates the "Delta" between prediction time and real time and the distance between prediction and events themselves. The sum of these factors determines a coefficient that returns the delay of the ship arrival versus the prediction made inside the database.

Inside berthing operations it is obligatory to distinct the status of the jobs: some of them have a fixed position, which is not allowed to delay, while others are "free" and are classified with a priority index based on the customers' criticality (using for example an ABC analysis) and/or on the Due Date.

This work is also devoted to simulate the resources' utilization in a stochastic environment based on a priority index that determines the operating sequence, except for the operations already started, which is fixed and deterministic; inside there different scenarios are simulated using a system of stochastic rules to be evaluated in terms of strength, which is the goal of the entire scheduling process.

The model used for simulation is a discrete event one and it aims to be an useful tool for supporting planning of a great reality inside Italian maritime ports with container terminals: La Spezia Harbor; the model is quite simple and for its implementation it was preferred

an higher speed of computation rather than a better precision.

Conceptually the idea is to divide all the operations in tasks, which are the elementary work units and so they are the leafs of an hypothetic tree of the operations; every task requires the work specialization and the availability of the resources allowed to do this task; every task is inside a well defined sequence with a well defined order of priority and it has a known (in stochastic terms) length, which is at least 1 hour or 1 day (8 hours).

The sequence of a group of tasks defined before is called "job", and it is the next level of the operations' tree, the tasks of a job are ordered with a strict sequence of priority and they are often grouped by workers' specializations (preparation, loading, unloading, moving...)

Inside this project are also defined the workstations, which are the descriptions of the available resources in order to run a certain number of simulation replications. In order to sequence the activities at their best it is also provided to use an optimization algorithm well known in the field of the Operative Research, the "2-Opt Algorithm".

## COMBINING SIMULATION AND MATHEMATICAL PROGRAMMING

One of the most important aspects in the maritime terminal management is related to the yard planning and to the internal traffic in order to reduce leadtime of incoming and outgoing goods. This point is even truer in container terminal where space and movement procedures plays an important role. One of the most interesting techniques used to proper solve this problem is based on Modeling & Simulation (M&S) where the entire terminal is modeled in order to find bottleneck and to test the effect of the upcoming and outgoing ships. This methodology is extremely expensive and time consuming since requires an intensive modeling effort that has to be continued in the life of the terminal in order to ensure proper representation of the reality. In other words simulation is sometime used only to solve some particular problems (i.e. yard management, docking/undocking procedures) and only in a very few case employed as a general methodology for the proper management of the entire terminal. This paper outline an approach in which simulation still play an important role remaining, at the same time, the principal test bed of the terminal management strategies an tactics rather than became the sole driver.

Since simulation is, basically, a what if technique isn't able alone to propose innovative solutions to the managers but helps to better evaluate their decisions. Only by combining it with Response Surface Methodology (RSM) a quest for a better solution can begin and, in this sense, simulation can play a crucial role.

In real life application, in fact, simplified models are used to rough cut first tentative solution that have to be tested under uncertain constraints in order to be declared effective. Classical mathematical solutions, such as Linear Programming and/or Theory of Constrains can

be regarded as base approach for specific allocation problems such as a Yard Planning. Still one point remain on the table: how robust is the solution? In this way simulation is used to test solutions for their highest robustness by using DOE (Design of Experiment) and RSM policies and practices can now be set to their maximum robustness by investigating the role of their "parameter" to the appropriate objective function.

The proposed methodology starts with a Mathematical Model for supporting Berth Planning able to identify the correct place (space) for berthing a ship and the correct instant (time) to carried out this operation, this part of the problem resolution take, as input, the optimal position of the berth for each ship, that is, for our purpose, the nearest docking place where the containers have to be taken or dropped. In this way any choice made in the direction of planning the container yard has is effect on the Berth Planning that is optimized by using an appropriate model.

In the berth planning, we try to minimize the penalty cost resulting from delayed departures of vessels and the additional handling cost resulting from deviation of the berthing position from the best location on the berth. Carriers usually inform the expected arrival time and the requested departure time of vessels to the terminal operator. Based on the information, a terminal operator tries to satisfy the requested departure time of each vessel. However, when the arrival rate of vessels is high or unexpected arrivals occur, it may not be possible to service all of the vessels before the service completion time that they requested. Thus, departures of some vessels may be delayed over the requested due time. Note that the terminal operator has different priorities for different types of vessels. The priorities can be converted into weights (c) on the penalty cost of vessels in the objective function.

By considering the following notation is possible to express the Berth Planning problem in the form of a Linear Programming model, in particular let be:

$N$  = The total number of vessels.

$L$  = The length of the berth.

$p_i$  = The best berthing location of vessel  $i$ -th.

This location is represented by the  $x$ -coordinate of the leftmost end of the vessel and determined considering the distribution of containers already arrived or a designated location for a specific vessel. The reference point for  $x$ -coordinate is the leftmost boundary of the berth

$x_i$  = The berthing position of vessel  $i$ -th (a decision variable).

$y_i$  = The berthing time of vessel  $i$ -th (a decision variable).

$a_i$  = The estimated arrival time of vessel  $i$ -th.

$d_i$  = The requested departure time of vessel  $i$ -th.

$b_i$  = The requested time for the ship operation for vessel  $i$ -th.

This value includes the requested allowance between departure of a vessel and berthing of another vessel.

$c_{1i}$  = The additional travel cost per unit distance for delivering containers of vessel  $i$ -th resulting from deviation of berthing location from the best position .

$c_{2i}$  = The penalty cost per unit time of vessel  $i$ -th

resulting from a delayed departure over the requested due time.

$l_i$  = The length of vessel i-th.

This value includes the requested gap between adjacent vessels.

$z_{ij}^x = 1$ : if vessel i-th is located in the left-hand side of vessel j-th  
 $z_{ij}^x = 0$ : otherwise.

$z_{ij}^y = 1$ : if vessel i-th is located lower side of vessel j-th  
 $z_{ij}^y = 0$ : otherwise.

The objective function of the berth-planning problem can be written as follows:

$$\min \sum_{i=1}^N \{c_{1i} |x_i - p_i| + c_{2i} (y_i - b_i - d_i)^+\} \quad (1)$$

where:

$$x^+ = \max\{0, x\} \quad (2)$$

The first term of the objective function comes from the deviation of the berthing position from the best location and the second term is related to the penalty cost from the delay of the departure of vessels behind the requested departure time.

The model for the Berth Planning can be expressed in figure 1 where two different ships (i-th and j-th) are presented according to the above-mentioned notation.

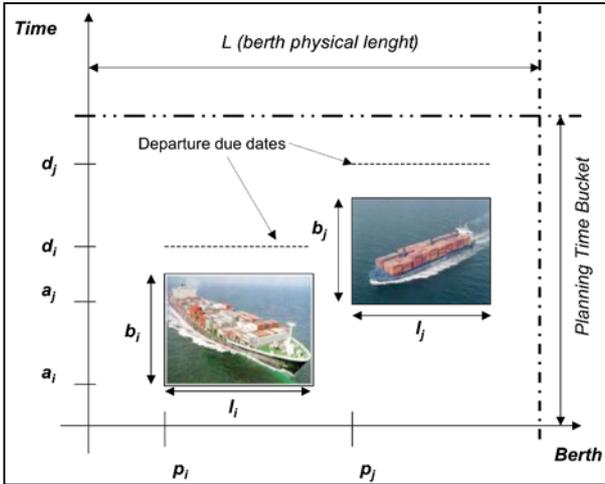


Figure 1: Model for Berth Planning

In order to implement the planning model some transformations in the variables and in the Objective Function have to be taken, in particular 4 new variables have to be defined as follows:

$$\geq_i^+ = |x_i \geq p_i| : x_i \geq p_i \geq 0 \quad (3)$$

$$\geq_i^{\approx} = |x_i \geq p_i| : x_i \geq p_i \geq 0 \quad (4)$$

$$i^+ = (y_i + b_i - d_i) : y_i + b_i - d_i \geq 0 \quad (5)$$

$$i^- = (y_i + b_i - d_i) : y_i + b_i - d_i < 0 \quad (6)$$

Following this notation the Objective Function can be defined as follows:

$$\min \sum_{i=1}^N \{c_{1i} (\geq_i^+ + \geq_i^-) + c_{2i} i^+\} \quad (7)$$

subject to the following constraints:

$$x_i + p_i = \sum_{j=1}^N z_{ij}^x l_j \quad (8)$$

$$y_i + b_i = \sum_{j=1}^N z_{ij}^y b_j \quad (9)$$

Other constraints must be added in order to guarantee that the position of the rightmost end of vessel i will be restricted by the length of the berth (10) and to ensure that two adjacent vessels will never be in conflict with each other with respect to the berthing time (11) and the berthing position (12). In these last constraints M is a big (i.e. 100000) value that void the relationship when the respective  $z_{ij}$  is different from 1. Constraint (13) excludes the case in which case the rectangles representing schedules for vessel i and j overlap with each other. Constraint (14) implies that a vessel cannot berth before she arrives.

$$x_i + l_i \leq L \quad \forall i \quad (10)$$

$$y_i + b_i = y_j + M(1 - z_{ij}^y) : i \neq j : i \neq j \quad (11)$$

$$x_i + l_i = x_j + M(1 - z_{ij}^x) : i \neq j : i \neq j \quad (12)$$

$$z_{ij}^x + z_{ji}^x + z_{ij}^y + z_{ji}^y \leq 2 : i \neq j : i \neq j \quad (13)$$

$$y_i \leq a_i \quad \forall i \quad (14)$$

To this constraint is necessary to add the common non negative ones and the integer constrains on the binary variables  $z_{ij}$ .

$$\forall_i^+, \forall_i^{\approx}, \forall_i^+, \forall_i^{\approx}, x_i \geq 0 : \forall i \quad (15)$$

$$z_{ij}^x, z_{ij}^y \in \{0, 1\} : i \neq j : i \neq j \quad (16)$$

Since the problem is known to be NP-hard practical solutions can be obtained only within 7 arriving ships and 72 hours of planning time buckets that is suitable for on-line, real time planning where the uncertainty of the arriving forecast is still reasonable.

For problems exceeding this sizes several heuristics are available in literature.

The proposed methodology is now based on the integration among arriving forecast process (time series analysis and regression), berth planning and yard planning. This last point is generally managed by using a simulator to support the choice of the container best location. According to the nature of the goods transported the entire container yard is divided in various areas and, inside of this area into various physical locations. Since each container can be part of a stack a handling coefficient has to be considered for each area and each location:  $HC_{ahk}(t)$  Handling Cost coefficient is generally calculated as squared value of the stack size within the upper limit of the stack itself. In this way container from a cargo presents a base handling cost given by (17), where a is the area, t is the time, h is the row position, k is the column position on the container yard and  $g_{shk}$  has the value of 1 for each container from a cargo placed, in the yard, in position

h-th k-th from s-th ship.

$$BHC_s(t) = \sum_{a=1}^{Areas} \sum_{h=1}^{Row} \sum_{k=1}^{Cols} (HC_{ahk}(t) \sum_{shk}) \quad (17)$$

Since the terminal situation is continuously changing the base handling cost for c-th cargo and s-th ship is changing according to the time. Total cost supported by the terminal C for managing containers that belong to the time bucket can be calculated as follow where the loading/unloading instant can be calculated for each i-th ship as time  $y_i$ :

$$C = \int_{i=1}^N \left\{ c_{1i} |x_i| p_i + c_{2i} (y_i | b_i | d_i)^+ \right\} + \int_{a=1}^{Areas} \int_{h=1}^{Row} \int_{k=1}^{Cols} [HC_{ahk}(y_i) | g_{ihk} ] \quad (18)$$

The simulator is, then, used to better estimate the second term of (19) where the time depended situation of the yard is varying within the simulation.

The Linear Programming model was implemented using LINGO™ and embedded into a Java based simulator that models the container yard management.

Data were efficiently stored in an ODBC RDBMS and successfully connected to the embedded LINGO™ solver. Thanks to the possibility of defining generally a Linear Programming problem into the embedded solver. The authors successfully implemented a tool able to assist managers in every day terminal operations.

Recent development have pointed up the opportunity of integrate a railroad simulator model that is under implementation in the CIELI Laboratory in order to support the input/output operations of the containers arriving and leaving the terminal from land.

## CONCLUSIONS

Modeling and simulation in the field of maritime container logistics operations has proven to be an effective technique able to solve complex problems and support managers in taking their decision.

Combination of both mathematical modelling and discrete event simulation can be regarded as a powerful approach for increase the robustness of the decision making process by evaluating different scenarios.

The proposed implementation combined all the presented techniques into software tools continuously updated from harbour ERP effectively serving as terminal DSS.

A real life application in the La Spezia terminal has now finishing its implementation phase and is actively demonstrates the benefits of the proposed methodology.

## REFERENCES

Bacardit M., Canudas J., Piera M.A., Vila J., Guasch A, Amettler S., Verges D., "A Generic Reusable Model of a Harbour System Described at a Macro Level Detail" Proceedings of HMS 2002, October, 3-5, 2002, Bergeggi, Italy.

- Bruzzone A.G., Mosca R., Briano C., Brandolini M., "Simulation & Web Technologies in Harbour Applications", Proceedings of TransBaltica 2002, June 14-15, Riga, Latvia, pp.147-160
- Bruzzone A.G., Revetria R., "Web Based Distributed Simulation for Supporting Maritime Terminal Operations", Proceedings of the 19th International Port Conference "Essentials Strategic Issues for Port & Transport Development", January, 26-28,2003 Alexandria, Egypt.
- Bruzzone, A.G., Giribone, P., and Revetria, R. Operative requirements and advances for the new generation simulators in multimodal container terminals. in Proceedings of the 1999 Winter Simulation Conference. 1999: Society for Computer Simulation International.
- Choi H.R., Park N.K., Kim H.S. Lee S.W., Park B.J., Kim J.J., Lee K.S. "An ERP approach for Container Terminal Operating Systems", Proceedings of the 18Th International Port Conference on Increasing Private Sector Involvement in Ports & Transport: An Opportunity or A Threat?, January 27-29, 2002 Alexandria, Egypt
- Duinkerken M.B., Ottjes J.A., Lodewijks G., "The Application of Distributed Simulation in TOMAS: Redesigning a Complex transportation Model", Proceedings of WSC '02, December 8-12,2002 San Diego, CA, USA, pp. 1207-1213
- Dzielinski A., Amborski K., Kowalczyk P., Sukienik J., "Simulation - Based Approach to the Analysis of Logistics Processes at Gdansk Container Terminal" Proceedings of HMS 2003, September, 18-20, 2003, Riga, Latvia
- Fuji S., Ogita A., Kidani Y., Kaihara T., "Synchronization Mechanisms for Integration of Distributed Manufacturing Simulation Systems", Simulation 72:3., 1999, pp.187-197
- Fujimoto R.M., "Parallel and Distributed Simulation Systems", Proceedings of WSC '01, Arlington, VA, USA
- Griese H., El-Nahass M., "Operational Benefits of a PDGPS Based Container Position Tracking System", Proceedings of the 19th International Port Conference "Essentials Strategic Issues for Port & Transport Development", January, 26-28,2003 Alexandria, Egypt.
- Janssens G.K., Theunissen C., "A Less - Flexibility: First Heuristic for the Placement of Inland Vessels in a Lock" Proceedings of HMS 2003, September, 18-20, 2003, Riga, Latvia
- Jeffery K., "Recent Developments in Information Technology for Container Terminals", Cargo Systems, London, 1999
- K.C. Moon "A Mathematical Model and a Heuristic Algorithm for Berth Planning," Logistics Systems Lab., Industrial Engineering, Pusan National University, Unpublished Doctoral Thesis, 2000.
- Lui E., Poh H.Y., "Computerized Container Terminal Management", UNCTAD Monographs on Port Management, UNCTAD/ SHIP/ 494 (10), New York, 1993
- Vecke H.P.M.; Ottjes J.A., "TOMAS: Tools for Object-Oriented Modelling and Simulation", Proceedings of the Business and Industry Simulation Symposium, Washington D.C., 2000, pp.76-81
- Yazdani R., Schmidt F.A.; Wilkie G., "The Development of a Generic Simulation System for Ports", Proceedings of HMS 2003, September, 18-20, 2003, Riga, Latvia

# THE ALGORITHM OF NEGOTIATION OF MULTI AGENTS FOR PLANNING IN GEOGRAPHICALLY DISTRIBUTED LOGISTIC SUPPLY CHAINS

Anatoly Levchenkov\*, Mikhail Gorobetz\*\*

Transport and Machinery Faculty\*  
Computer Science and Information Technology Faculty\*\*  
Riga Technical University  
1, Kalku St.,  
LV-1658 Riga, Latvia  
[levas@latnet.lv](mailto:levas@latnet.lv)\*  
[mgorobetz@gmail.com](mailto:mgorobetz@gmail.com)\*\*

## KEYWORDS

Multi-agent, schedule, negotiation, group decision making, logistics tasks

## ABSTRACT

The purpose is to realize the negotiation algorithm for multi-agents[2] in stochastic condition using Monte-Carlo methods [3].

The negotiation is intended to support planning in geographically distributed supply chains and to create such schedule[1] that reduces usage of resources and coordinates actions of the logistics process participants.

It is proposed to test negotiation algorithm to create production schedule for the model of geographically distributed supply chain, where processors are plants, distribution centres, retailers and node points of the route, but jobs are transport units. The main condition is indefinite and stochastic character of some supply chain parameters such as delivery time and demand.

All the data for solution of the task have to be incorporated in a database, and appropriate programming languages and servers will be used for realization of algorithms using group decision support system.

The main advantage of using the negotiation in supply chain is organizing a centralized control system for all system, which follows large reduction of costs concerned with idle time and unsatisfied demand.

## INTRODUCTION

This paper is based on our previous work [5]. This paper introduces a modification of negotiation algorithm in order to implement it in multi-agent supply chain systems.

Traditionally, marketing, distribution, planning, manufacturing, and the purchasing of organizations along the supply chain operate independently. Many manufacturing operations are designed to maximize throughput and lower costs with little consideration for the impact on inventory levels and distribution capabilities. Purchasing contracts are often negotiated with very little information beyond historical buying patterns. The result of these factors is that there is not a single, integrated plan for the organization. Clearly, there is a need for a mechanism through which these different functions can be integrated together. Supply chain

management is a strategy through which such integration can be achieved. [4]

Usage multi-agents of high level architecture and negotiation in supply chain allow centralizing control of inventory, demand and orders.

There are two types of geographically distributed supply chain [4]:

- Supply chain has wide distribution network
- Big distances and long delivery time between locations

There are three main goals of each supply chain participant:

- Satisfy final demand
- Reduce costs
- Increase profit

Conflicts can be solved by negotiation are

- Time – impossible to accomplish multiple jobs simultaneously
- Transport - transport is not available
- Demand - not enough inventories or not enough time to complete order

## MULTI-AGENT SUPPLY CHAIN STRUCTURE

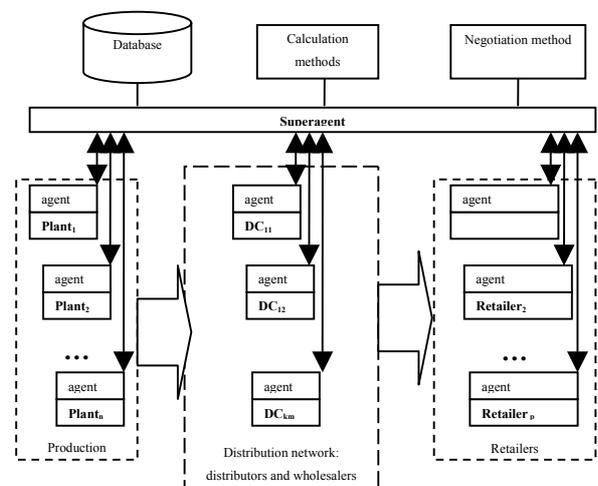


Figure 1. Structure.

There can be two kinds of objects in logistics supply process[1]. The processors are immovable objects – plants, distributors, shops etc. and the jobs that are

objects served by processors - for example, trucks. Each object in a supply chain has own multi-agent. Also there is a super agent – a multi-agent, which realizes negotiation and makes the schedule by using group decision support system.

A set of processors and a set of jobs are given as input data to solve the task. Each job consists of a number of operations and each operation is carried out by one processor. The duration of each operation and directive terms are randomly generated. General algorithm is proposed to solve the scheduling task to co-operate the work of multi-agents.

As it is seen from structure in fig.1 all information, calculations and negotiation process is concentrated in superagent.

The superagent organizes group decision making process. Other multi-agents are responsible only for their own objects, share operational information and accept or do not accept decisions and negotiation according to possible concessions.

The scheduling task is represented as a cooperative, non-coalition game, where all players are multi-agents and each is responsible only for its own object. They are interested in receiving maximal profit with minimal costs, not disturbing each other. The result of negotiation algorithm is the schedule that depends on stochastic processes.

All conflicts in supply chain are solved through negotiation process. Superagent plays a role of the arbitration judge, which nominates what players have to do proceeding from results of negotiation.

## MATHEMATICAL MODEL FOR SUPERAGENT

### Notation:

$\mathfrak{R}^+$  - nonnegative real number

*Description* – standardized verbal description for specified case.

### Basic information in superagent's database

- Cost description  
*Description.Costs* = {fixed order cost, running cost per hour, running cost per km, holding cost per unit, price Plant DC, price DC Retailer, price Retailer Client, Price Plant Retailer}
- Transportation routes (from-to) – map of distribution network:
  - Graph –  $G(N,E)$ , where
    - nodes –  $N = \{n_1, n_2, \dots, n_m\}; n \in V$
    - edges – subset from Cartesian product of nodes  
 $E = \{e_1, e_2, \dots, e_n\};$   
 $e_j = \langle n_a, n_b \rangle, j = \overline{1, n}, a, b = \overline{1, m}, a \neq b, E \subseteq N \times N$
    - supply chain levels -  $Lvls = \overline{0, LMax}, LMax \in \mathbb{N}$
- A set of possible types of transport unit for all system:  $TTypes = \{ttype_1, ttype_2, \dots, ttype_t\}, ttype_d \in Text$ 
  - speed -  $speed[ttype_d] \in \mathfrak{R}^+, d = \overline{1, t}$

- costs -  
 $Costs[ttype_t] = \langle Ctype_1, Cvalue_1 \rangle, \langle Ctype_2, Cvalue_2 \rangle, \dots, Ctype_s \in Description.Costs, Cvalue_s \in \mathfrak{R}^+, d = \overline{1, t}$
- cargo maximum -  $CapMax[ttype_d] \in \mathbb{N}, d = \overline{1, t}$

- Information about all edges of graph:
  - distance -  $dist[e_j] j = \overline{1, n}$
  - possible transport type on current edge -  
 $TTedge[e_j] = \{ttype_1, ttype_2, \dots\}, ttype_d \in TTypes$
  - probable delivery deviation for each transport type in percentage -  $dev\%_j[ttype_d] \in \mathfrak{R}^+$

### Necessary input information in supply chain for negotiation from agents:

- Information about all nodes in graph:
  - type of node -  $ttype[n_i] \in Description, i = \overline{1, m}$
  - level in supply chain (0 – production, max – retailer) -  $level[n_i] \in Lvls$
  - supplier -  $Sup[n_i] \in N, i = \overline{1, m}$
  - capacity -  $cap[n_i] \in \mathbb{N}, i = \overline{1, m}$
  - costs –  $Costs[n_i] = \langle Ctype_1, Cvalue_1 \rangle, \langle Ctype_2, Cvalue_2 \rangle, \dots, Ctype_s \in Description.Costs, Cvalue_s \in \mathfrak{R}^+$
  - Inventory
    - current -  $InvCur[n_i], i = \overline{1, m}$
    - ordered -  $InvOrd_l[n_i], i = \overline{1, m}, l \in \mathbb{N}$
    - directive terms -  
 $dterm_l[n_i] = \langle beg, end \rangle,$   
 $i = \overline{1, m}, l \in \mathbb{N}, beg, end \in \mathfrak{R}^+ (date\ or\ time)$
  - order preparing time -  $OprepTime[n_i] \in \mathfrak{R}^+$
  - compromises and additional charges –  
 $C[n_i] = \langle type_1, values_1 \rangle, \langle type_2, values_2 \rangle, \dots, type_p \in Description,$   
 $values_p = \langle minvalue_p, costs_{p1} \rangle, \langle maxvalue_p, costs_{p1} \rangle, \dots, p \in \mathbb{N}, r = \overline{1, v}, minvalue_p, maxvalue_p \in \mathfrak{R}, costs_p \in \mathfrak{R}^+$
  - forecasted demand -  
 $demand[n_i] = \langle time_1, amount_1 \rangle, \langle time_2, amount_2 \rangle, \dots, \langle time_k, amount_k \rangle, \dots, i = \overline{1, m}$
  - schedule  
 $sched[n_i] = \langle time_1, amount_1 \rangle, \langle time_2, amount_2 \rangle, \dots, \langle time_k, amount_k \rangle, \dots$   
 $time \in \mathfrak{R}, amount \in \mathbb{N}, k \in \mathbb{N}, i = \overline{1, m}$
- Additional information from production nodes:
  - maximal production amount -  
 $prodMax[n_i] \in \mathbb{N}, i = \overline{1, m}$
  - current production amount -  
 $prodCur[n_i] \in \mathbb{N}, i = \overline{1, m}$
- Information about all movable objects (transport):
  - a set of transport units -  $TR = \{tr_1, tr_2, \dots, tr_v\}$ 
    - unit type -  $ttype[tr_r] \in TTypes, r = \overline{1, v}$
    - assigned to -  $tassign[tr_r] = n_i, r = \overline{1, v}, n_i \in N$
    - origin node -  $origin[tr_r] \in N, r = \overline{1, v}$

- destination node -  $\underline{dest}[tr_r] \in N, \quad r = \overline{1, v}$
- current location at the moment of negotiation -  $\underline{loc}[tr_r] = \langle place, dist_1, dist_2 \rangle,$   
 $place \in E \cup N, \quad dist_1, dist_2 \in \mathfrak{R}^+$
- current cargo -  $\underline{Cur}[tr_r] \in \mathfrak{S}, \quad r = \overline{1, v}$
- directive term -  $\underline{dterm}[tr_r] = \langle beg, end \rangle,$   
 $r = \overline{1, v}, \quad beg, end \in \mathfrak{R}^+ \text{ (date or time)}$
- due date/time -  $\underline{dtime}[tr_r] \in \mathfrak{R}^+$
- compromises and additional charges -  
 $\underline{C}[tr_r] = \langle type_1, values_1 \rangle, \langle type_2, values_2 \rangle, \dots,$   
 $type_p \in Description,$   
 $values_p = \langle \minvalue_p, costs_{p1} \rangle, \langle \maxvalue_p, costs_{p1} \rangle, \dots,$   
 $p \in \mathfrak{S}, \quad r = \overline{1, v}, \quad \minvalue_p, \maxvalue_p \in \mathfrak{R}, \quad costs_p \in \mathfrak{R}^+$
- schedule -  
 $\underline{sched}[tr_r] = \langle time_1, amount_1 \rangle, \langle time_2, amount_2 \rangle,$   
 $\dots, \langle time_k, amount_k \rangle, \dots,$   
 $time \in \mathfrak{R}, \quad amount \in \mathfrak{S}, \quad k \in \mathfrak{S}, \quad r = \overline{1, v}$

• Information about delivery routes in graph:

- delivery route -  
 $\underline{Routes}(Sup[n_i], n_i) = \{route_1, \dots, route_y\}$   
 $route_k(Sup[n_i], n_i) = \{edge_1, \dots, edge_z\},$   
 $k = \overline{1, y} \quad z \in \overline{1, n} \quad edge_j \in E,$   
 $edge_1 = \langle Sup[n_i], n_a \rangle, \quad edge_z = \langle n_a, n_i \rangle$
- distance -  $\underline{dist}[route_k] = \sum_{j=1}^z \underline{dist}[edge_j],$   
 $\forall route_k \in \underline{Routes}(Sup[n_i], n_i)$
- possible transport types for each route  
 $\underline{TTroute}_k(Sup[n_i], n_i) = \bigcap_{j=1}^z \underline{TTEdge}(edge_j),$   
 $k = \overline{1, y}, \quad i = \overline{1, m}$

Necessary calculations for supply chain model

- running duration and costs for each edge transport:
  - running mean value -  
 $\underline{dmean}_j[ttype_d] = \underline{dist}[e_j] / \underline{speed}[ttype_d]$   
 $j = \overline{1, n} \quad \forall ttype_d \in \underline{TTEdge}[e_j]$
  - running standard deviation -  
 $\underline{ddev}_j[ttype_d] = \underline{dmean}_j[ttype_d] * \underline{dev}\%_j[ttype_d]$   
 $j = \overline{1, n} \quad \forall ttype_d \in \underline{TTEdge}[e_j]$
  - maximal running time  
 $\underline{MaxDTime}_d[e_j] = \underline{dmean}_j[ttype_d] + \underline{ddev}_j[ttype_d]$   
 $j = \overline{1, n} \quad \forall ttype_d \in \underline{TTEdge}[e_j]$
  - maximal running costs -  
 $\underline{MaxRunCosts}_d[e_j] = \underline{MaxDTime}_j[ttype_d] \times \underline{RunCostsPerHour}[ttype_d]$   
 $j = \overline{1, n} \quad \forall ttype_d \in \underline{TTEdge}[e_j]$
- running duration and costs for each route transport:
  - maximal running time  
 $\underline{MaxDTime}_d[route_k] = \sum_{j=1}^z \underline{MaxDTime}_d[e_j]$   
 $\forall route_k \in \underline{Routes}(Sup[n_i], n_i), \quad \forall ttype_d \in \underline{TTroute}_k(Sup[n_i], n_i)$   
 $k = \overline{1, z} \quad i = \overline{1, m}$
  - maximal running costs -  
 $\underline{MaxRunCosts}_d[route_k] = \underline{MaxDTime}_d[route_k] \times$   
 $\times \underline{running\ cost\ per\ hour}[ttype_d]$   
 $\forall ttype_d \in \underline{TTroute}_k(Sup[n_i], n_i) \quad k = \overline{1, z} \quad i = \overline{1, m}$

**PRODUCTION SCHEDULING ALGORITHM**

A production schedule is a result of multi agent collaboration which specify production amount for the specific date.

Step 0. Level selection for analysis

- select level of supply chain  $L = LMax$

Step 1. Demand analysis using Monte-Carlo methods

- 1.1. superagent gets subset of retailers  
 $N' \subset N, \quad \text{where } level[n_i] = L$
- 1.2. superagent gets demand of each retailer  
 $demand[n_i], \quad \forall n_i \in N'$
- 1.3. calculating average demand per day for each retailer:  
 $\mu[n_i] = \frac{\sum_{a=1}^k demand[n_i].amount_a}{k}$
- 1.4. calculating standard deviation of demand for each retailer:  
 $\sigma[n_i] = \sqrt{\frac{\sum_{a=1}^{k-1} (demand[n_i].amount_a - demand[n_i].amount_{a+1})^2}{k}}$
- 1.5. calculating safety stock for each retailer:  
 $Sstock[n_i] = \sigma[n_i] * z, \quad z = 1,65 \text{ (for service level 95\%)}$

Step 2. Optimal order quantity planning for each retailer

- 2.1. superagent gets supplier for each retailer:  
 $Sup[n_i], \quad \forall n_i \in N'$
- 2.2. get possible transport types for each supply:  
 $\underline{TSup}[n_i] = \{TTroute_1(Sup[n_i], n_i), \dots, TTroute_k(Sup[n_i], n_i)\},$   
 $k = \overline{1, y} \quad \forall n_i \in RET$
- 2.3. calculate EOQ for each retailer and transport:  
 $\underline{EOQ}_d[n_i] = \sqrt{\frac{2 \cdot (\mu[n_i] + \sigma[n_i]) \cdot \text{fixed order cost}[ttype_d]}{\text{holding cost per unit}}}$   
 $ttype_d \in \underline{TSup}[n_i], \quad n_i \in N'$
- 2.4. EOQ correction according to transport maximal capacity:  
 $\underline{EOQ}_d[n_i] = \begin{cases} \underline{EOQ}_d[n_i], & \text{if } \underline{EOQ}_d[n_i] < \underline{CapMax}[ttype_d] \\ \underline{CapMax}[ttype_d], & \text{if } \underline{EOQ}_d[n_i] > \underline{CapMax}[ttype_d] \end{cases}$   
 $ttype_d \in \underline{TSup}[n_i], \quad n_i \in N'$
- 2.5. Average interval between order calculation:  $\underline{Int}_d[n_i] = \underline{EOQ}_d[n_i] / \mu[n_i]$

Step 3. Order time and transport type calculation

- 3.0. Start with current date:  $date = 0$
- 3.1. Getting current inventory and capacity.  
 $\underline{Inv}[date].start = \underline{InvCur}[n_i] + \underline{Order}_i[date].amount,$   
 $\underline{cap}[n_i] \quad n_i \in N'$   
If  $\underline{Inv}_i[date].start > \underline{cap}[n_i]$  then initiate negotiation
- 3.2. Calculating inventory status  
 $\underline{Inv}_i[date].end = \underline{Inv}_i[date].start + \underline{Order}_i[date].amount -$   
 $- \underline{demand}[n_i].amount_k$   
 $\underline{Inv}_i[date].start = \underline{Inv}_i[date-1].end + \underline{Order}_i[date].amount$   
 $\forall n_i \in N'$
- 3.3. Calculating order delivery due date critical delivery date:  
if  $\underline{Inv}_i[date] - \underline{Sstock}[n_i] < 0$  then  $\underline{ODate}_{nr} = date$   
else  $date = date + 1, \text{ repeat 3.2.}$   
if  $\underline{Inv}_i[date] < 0$  then  $\underline{OCritDate}_{nr} = date$   
else  $date = date + 1, \text{ repeat 3.2.}$
- 3.4. Getting maximal running time and running costs

$MaxDTime_d[route_k], MaxRunCosts_d[route_k],$

$\forall route_k \in Routes(Sup[n_i], n_i), \forall n_i \in N'$

3.5. Evaluate transport type by order cost:

$TC_d[n_i] = \text{fixed order cost}[ttype_d] + 2MaxRunCosts_d[route_k]$

$eval_d[n_i] = TC_d[n_i] / Int_d[n_i]$

$\forall route_k \in Routes(Sup[n_i], n_i), \forall n_i \in N'$

3.6. Select the best transport type

$DTType(Sup[n_i], n_i) = d, \text{ if } eval_d[n_i] = \min(\forall eval_d[n_i])$

$ttype_d \in TSup[n_i], \forall n_i \in N'$

3.7. Calculating order amount:

$Order_i[date].amount = EOQ_{DTType(Sup[n_i], n_i)}[n_i] + neg.amount[n_i].corr$

3.8. Create schedule records (amount, order preparing, departure, delivery, return)

$record[n_i] = \langle amount, prep, depart, due, return \rangle$

$amount = Order_i[date].amount + negotiation[n_i].demand.corr,$

$due = ODate_{nr}[n_i] + negotiation[n_i].sched.corr$

$depart = due - MaxDTime_d(Sup[n_i], n_i)$

$prep = depart - OprepTime[n_i]$

$return = due + MaxDTime_d(Sup[n_i], n_i)$

3.9. Check records.

If  $record[n_i].prep \geq 0$  then 3.9.

else - **initiate negotiation procedure**

3.10. Check availability for transport unit

$tr[n_i] \in TR$

3.11. If transport is available then assign records to schedules

$sched_{nr}(Sup[n_i]) = \langle Job_{nr}, n_i, tr[n_i], record[n_i].prep,$

$record[n_i].departure, record[n_i].amount \rangle$

$sched_{nr}[tr] = \langle Job_{nr}, Sup[n_i], n_i, record[n_i].depart, record[n_i].due,$

$OCritDate_{nr}[n_i], record[n_i].return, record[n_i].amount \rangle$

else - **initiate negotiation procedure**

3.12. Return 3.1.

3.13.  $L=L-1$ . Return Step 1.

## NEGOTIATION PROCEDURE

Step 1. Define conflict types in a problem:

$conflicts \subseteq \{\text{time, demand, transport}\}$  time,

Step 2. Offer possible strategies to solve a problem according to conflicts

$strategies[conflict_i] \subseteq \{\text{schedule correction, another transport unit selection, another transport type selection}\}$

$conflict_i \in \text{conflict}$

Step 3. Imitate implementation of

$\exists strategy_j \in strategies$  using solution methods [5].

Step 4. Evaluate charges of using strategy.

Step 5. Choose strategy with minimal costs

## NUMERICAL EXAMPLE

For numerical example, abstract supply chain is taken. It is simplified just to show principles of negotiation between multi agents. So, there is one type of product,

one plant and one level distribution network with 2 DC and 4 retailers.

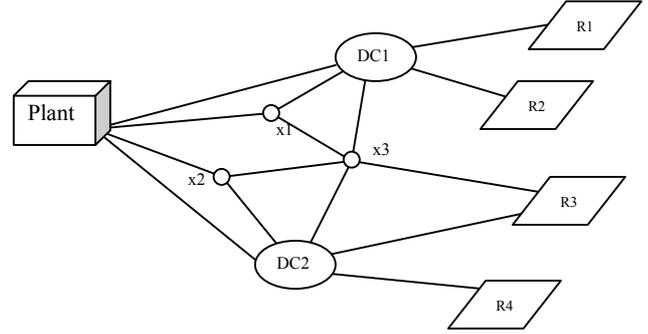


Figure 2. Numerical example supply network

### INPUT DATA

A set of nodes N (table 1), a set of possible transport types TTypes (table 2), a set of edges E and prenegotiation calculation (table 3), a set of Routes (table 4), transport units TR (table 5) are given.

Table 1: Nodes

Node	Type	Level
Plant	Plant	0
DC1	DC	1
DC2	DC	1
R1	Retailer	2
R2	Retailer	2
R3	Retailer	2
R4	Retailer	2
x1	Crossroad	NULL
x2	Crossroad	NULL
x3	Crossroad	NULL

Table 2: Transport types

Type	Speed (km/h)	FC	Running cost (CU/h)	Order preparing	Capacity
Ship	30	2300	6	24	500
Truck	50	800	2	24	50
Train	60	3000	9	24	500
Aircraft	400	7200	24	24	150

Table 3: Edges

Edge	From	To	Distance	Transport type	Probable delivery deviation	dmean (h)	ddev (h)	MaxDime (h)	MaxRun Cost (CU)
e1	Plant	x1	2500	Truck	0,1	50	5	55	110
e1	Plant	x1	2500	Train	0,1	42	5	47	423
e2	Plant	DC1	3500	Ship	0,1	117	12	129	774
e3	Plant	x2	3000	Train	0,1	50	5	55	495
e4	Plant	DC2	3900	Aircraft	0,1	10	1	11	264
e5	x1	DC1	2000	Truck	0,1	40	4	44	88
e5	x1	DC1	2000	Train	0,1	34	4	38	342
e6	x1	x3	1800	Train	0,1	30	3	33	297
e7	x2	x3	2000	Train	0,1	34	4	38	342
e8	x2	DC2	1800	Train	0,1	30	3	33	297
e9	DC1	R1	1600	Truck	0,1	32	4	36	72
e9	DC1	R1	1600	Train	0,1	27	3	30	270
e10	DC1	R2	1500	Truck	0,1	30	3	33	66
e10	DC1	R2	1500	Train	0,1	25	3	28	252
e11	DC1	x3	700	Truck	0,1	14	2	16	32
e12	DC2	x3	800	Truck	0,1	16	2	18	36
e13	DC2	R3	1300	Truck	0,1	26	3	29	58
e14	DC2	R4	1200	Truck	0,1	24	3	27	54
e15	x3	R3	2000	Train	0,1	34	4	38	342

Table 4: Routes

From	To	Route	Edges	Distance	Transport type	MaxDtime (h)	MaxRun Cost (CU)
Plant	DC1	route1	e1,e5	4500	Truck	99	198
					Train	83	747
					Ship	129	774
Plant	DC2	route1	e3,e8	4800	Train	88	792
					Aircraft	11	264
DC1	R1	route1	e9	1600	Truck	36	72
					Train	30	270
DC1	R2	route1	e10	1500	Truck	33	66
					Train	28	252
DC2	R3	route1	e13	1300	Truck	29	58
					Truck	27	54
DC1	DC2	route1	e11,e12	1500	Truck	33	66
					Train	126	1134
Plant	R3	route1	e1,e6,e15	6800	Train	126	1134
					Train	129	1161

Table 5: Transport units

Unit	Type	Current location		
		Edge/Node	Dist1	Dist2
Plane1	Aircraft	Plant	0	0
Ship1	Ship	Plant	0	0
Truck1	Truck	Plant	0	0
Truck2	Truck	Plant	0	0
Truck3	Truck	DC1	0	0
Truck4	Truck	DC1	0	0
Truck5	Truck	DC2	0	0
Truck6	Truck	DC2	0	0
Truck7	Truck	DC2	0	0
Train1	Train	Plant	0	0
Train2	Train	Plant	0	0
Train3	Train	DC1	0	0

GETTING PRODUCTION SCHEDULE USING MULTI-AGENTS AND NEGOTIATION

Step 0. L = 2

Step 1.

1.1.  $N' = \{R1, R2, R3, R4, R5, R6\}$

1.2. Getting demand from each retailer.

Day	1	2	3	4	5	6	7	8
Begin h	24	48	72	96	120	144	168	192
End h	48	72	96	120	144	168	192	216
R1	15	12	10	8	19	14	16	11
R2	13	14	10	11	17	19	22	12
R3	10	12	11	8	15	13	14	11
R4	8	7	6	5	9	4	3	6
Day	9	10	11	12	13	14	15	16
Begin h	216	240	264	288	312	336	360	384
End h	240	264	288	312	336	360	384	408
R1	21	18	15	10	17	18	16	15
R2	18	16	11	14	15	25	23	13
R3	12	13	10	9	10	11	9	10
R4	7	10	9	8	7	11	13	8

1.3., 1.4., 1.5. Mean value, deviation of demand and safety stock calculation (table 7)

Customer	$\mu$	$\sigma$	Safety stock
R1	14,69	3,65	7
R2	15,81	4,53	8
R3	11,13	1,93	4
R4	7,56	2,56	5

Step 2.

2.1., 2.2. Getting suppliers and transport types

Customer	Supplier	Transport
R1	DC1	train, truck
R2	DC1	train, truck
R3	DC2	truck
R4	DC2	truck

2.3., 2.4., 2.5. EOQ calculation, correction and frequency

Customer	Supplier	Transport	Max capacity	EOQ	EOQ corr.	interval
R1	DC1	truck	50	55	50	3
		train	500	105	105	7
R2	DC1	truck	50	58	50	3
		train	500	111	111	7
R3	DC2	truck	50	46	46	4
R4	DC2	truck	50	41	41	5

Step 3.

3.0. Day = 1.

Iteration 1

3.1. Getting inventory

Customer	Current	Maximal	Check
R1	45	120	OK
R2	50	150	OK
R3	30	90	OK
R4	25	75	OK

3.2., 3.3. Calculating inventory status due date and critical due date:

	Day	1	2	3
	Start h	24	48	72
	End h	48	72	96
R1	Start	45	30	18
	End	30	18	8
	Order date	FALSE	FALSE	FALSE
	Critical date	FALSE	FALSE	FALSE
R2	Start	50	37	23
	End	37	23	13
	Order date	FALSE	FALSE	FALSE
	Critical date	FALSE	FALSE	FALSE
R3	Start	30	20	8
	End	20	8	-3
	Order date	FALSE	FALSE	72
	Critical date	FALSE	FALSE	72
R4	Start	25	17	10
	End	17	10	4
	Order date	FALSE	FALSE	72
	Critical date	FALSE	FALSE	FALSE

3.4., 3.5, 3.6. Getting maximal running time and running costs, evaluate transport type and selecting the best for each retailer:

From	To	Transport type	Fixed order cost (CU)	Max Dtime (h)	Max Run Cost (CU)	TC (CU)	interval	Eval	Select
DC2	R3	Truck	800	29	58	916	4	229	OK
DC2	R4	Truck	800	27	54	908	5	182	OK

3.7. Order amounts:

From	To	Negotiated correction	Order amount
DC2	R3	NULL	46
DC2	R4	NULL	41

3.8., 3.9. Creating and checking schedule records

Job	To	Amount	Order preparing	departure	due	return	negotiation correction	Record check
Job11	R3	46	19	43	72	101	0	OK
Job12	R4	41	21	45	72	99	0	OK

3.10. Select available transport:

Supplier	Name	Type	Available	Return	Select
DC2	Truck5	Truck	TRUE	NULL	OK
	Truck6	Truck	TRUE	NULL	OK
	Truck7	Truck	TRUE	NULL	

3.11. Assigning record to schedules:

Table 6: DC2 schedule

Job	Dest	Unit	Prep	Depart	Amount
Job1	R3	Truck5	19(0)	43(1)	46
Job2	R4	Truck5	21(0)	45(1)	41

Table 7: Truck5 schedule

Job	Orig	Dest	Amount	Depart	Due	Return
Job1	DC2	R3	46	43(1)	72(3)	101(4)

Table 8: Truck6 schedule:

Job	Orig	Dest	Amount	Depart	Due	Return
Job2	DC2	R4	41	45(1)	72(3)	99(4)

3.12. Return 3.1.

**Iteration 2**

3.1. Inventory (day 3)

Customer	Current	Maximal	Check
R1	45	120	OK
R2	50	150	OK
R3	54	90	OK
R4	51	75	OK

3.2. etc...

3.11. Result of iteration

Table 9: DC1 schedule:

Job	Dest	Unit	Prep	Depart	Amount
Job1	R1	Truck3	36(1)	60(2)	50
Job2	R2	Truck4	39(1)	63(2)	50

Table 10: Truck3 schedule:

Job	Orig	Dest	Amount	Depart	Due	Return
Job1	DC1	R1	50	60(2)	96(4)	132(5)

Table 11: Truck4 schedule:

Job	Orig	Dest	Amount	Depart	Due	Return
Job2	DC1	R4	50	63(2)	96(4)	129(5)

**Iterations 3 - 7**

Repeat previous steps without conflicts. Result schedules are following.

Table 12: DC1 schedule

Job	Dest	Unit	Prep	Depart	Amount
Job11	R1	Truck3	36	60	50
Job12	R2	Truck4	39	63	50
Job13	R1	Truck4	108	132	50
Job14	R2	Truck3	111	135	50
Job15	R1	Truck3	180	204	50
Job16	R2	Truck4	183	207	50
Job17	R1	Truck4	276	300	50
Job18	R2	Truck3	279	303	50

Table 13: DC2 schedule

Job	Dest	Unit	Prep	Depart	Amount
Job21	R3	Truck5	19	43	46
Job22	R4	Truck6	21	45	41
Job23	R3	Truck6	115	139	46
Job24	R3	Truck5	187	211	46
Job25	R4	Truck6	189	213	41
Job26	R3	Truck6	307	331	46
Job27	R4	Truck5	309	333	41

Table 14: Truck3 schedule

job	orig	dest	amount	departure	due	return
job11	DC1	R1	50	60	96	132
job14	DC1	R2	50	135	168	201
job15	DC1	R1	50	204	240	276
job18	DC1	R2	50	303	336	369

Table 15: Truck4 schedule

job	orig	dest	amount	departure	due	return
job12	DC1	R2	50	63	96	129
job13	DC1	R1	50	132	168	204
job16	DC1	R2	50	207	240	273
job17	DC1	R1	50	300	336	372

Table 16: Truck5 schedule

job	orig	dest	amount	departure	due	return
job21	DC2	R3	46	43	72	101
job24	DC2	R3	46	211	240	269
job27	DC2	R4	41	333	360	387

Table 17: Truck6 schedule

job	orig	dest	amount	departure	due	return
job22	DC2	R4	41	45	72	99
job23	DC2	R3	46	139	168	197
job25	DC2	R4	41	213	240	267
job26	DC2	R3	46	331	360	389

**Iteration 7**

There are fragments from iteration

3.2., 3.3. Finding next due date:

	Day	16
	Start h	384
	End h	408
R1	...	
R2	Start	10
	End	-3
	Order date	384
	Critical date	384
R3	...	
R4	...	

3.4., 3.5, 3.6. Transport type evaluation

From	To	Transport type	Fixed order cost (CU)	Max Dtime (h)	Max Run Cost (CU)	TC (CU)	interval	Eval	Select
DC1	R2	Truck	800	33	72	944	3	290	OK
DC1	R2	Train	3000	28	270	3540	7	467	

3.7., 3.8, 3.9. Defining amount of order, creating record and check.

Job	To	Amount	Order preparing	departure	due	return	negotiation correction	Record check
Job19	R2	50	327	351	384	417	0	OK

3.10. Select available transport:

Supplier	Name	Type	Available	Return	Select
DC1	Truck3	Truck	FALSE	369(15)	
	Truck4	Truck	FALSE	372(15)	

3.11. There is no transport available – it is necessary to initiate negotiation process.

NEGOTIATION PROCEDURE

1. conflict detection
    - Not available transport of the best type
    - Time conflict
  2. possible solutions for this conflicts
    - Check availability for all possible transport type
    - Existing schedule correction
  3. evaluate solutions
- 3.1. Another transport strategy
- Checking all transport units of DC1

Supplier	Name	Type	Available	Return	Select
DC1	Truck3	Truck	FALSE	369(15)	
	Truck4	Truck	FALSE	372(15)	
	Train3	Train	TRUE	NULL	

- Evaluate costs of using this strategy

From	To	Transport type	Fixed order cost (CU)	Max Dtime (h)	Max Run Cost (CU)	TC (CU)
DC1	R2	Train	3000	28	270	3540

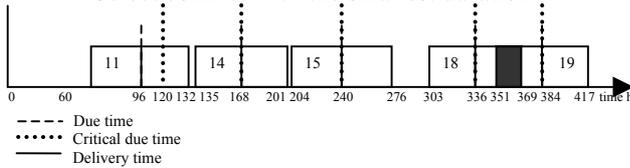
- Strategy 3.1. charges: **TC = 3540**

3.2. Schedule correction strategy – using negotiation algorithm for time conflict solution [5].

- Creating negotiation set

Conflict	Time crossing	Duration
Truck3, job18, job19	351-369	18
Truck4, job17, job19	351-372	21

- Select conflict with the smallest duration



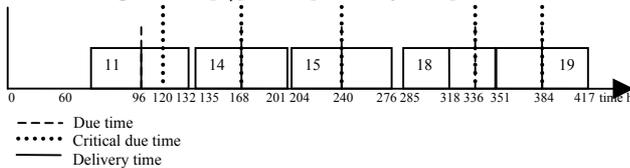
- Free space for job18, job19 t1 and t2

Truck	Interval	Free time
job19	384 – 384	0
job18	303 – 274	29

- Interval for moving – 369 - 351=18  
Free time = 29

- Result:

$$\text{negotiation}[n_i].\text{sched}[\text{truck 3, job18}].\text{corr} = -18$$



- Strategy 3.2. charges: **additional holding cost = time\*unit holding cost /2 = 18\*100/2 = 900**

Negotiation procedure result – strategy 3.2:

$$\text{negotiation}[n_i].\text{sched}[\text{truck 3, job18}].\text{corr} = -18$$

Result changed schedules are for DC1 (Table 18) and for Truck3 (table 19).

Table 18: DC1 schedule

Job	Dest	Unit	Prep	Depart	Amount
Job11	R1	Truck3	36	60	50
Job12	R2	Truck4	39	63	50
Job13	R1	Truck4	108	132	50
Job14	R2	Truck3	111	135	50
Job15	R1	Truck3	180	204	50
Job16	R2	Truck4	183	207	50
Job17	R1	Truck4	276	300	50
Job18	R2	Truck3	261	285	50
Job19	R2	Truck3	327	351	50

Table 19: Truck3 schedule

job	orig	dest	amount	departure	due	return
job11	DC1	R1	50	60	96	132
job14	DC1	R2	50	135	168	201
job15	DC1	R1	50	204	240	276
job18	DC1	R2	50	285	318	351
job19	DC1	R2	50	351	384	417

Calculated schedules for DC1 and DC2 are demand for next level of supply chain. This way, algorithm will repeat until the result – convenient schedule for production, distribution network participants and all transport units.

## CONCLUSIONS AND FURTHER RESEARCH

Numerical example shows functionality of the negotiation algorithm. In proposed mathematical model all types of all parameters are given. That is why developed algorithm can be easy realized in any programming language and DBMS.

Our aim for future is an expert system for supply chain management based on multi-agent negotiation algorithm intended to fast tasks solutions in conditions approached to real. For near future it is planned to focus attention on negotiation strategies and evaluation of these strategies.

## REFERENCES

- Conway R. Theory of Scheduling. Dover Publications. 2003.
- Ferber J. Multi-agent Systems: An Introduction to Distributed Artificial Intelligence. Addison Wesley. 1999
- Jeremy F. Shapiro. Modeling the Supply Chain. Wadsworth group, Thomson Learning inc., 2001
- F. Frank Chen. Process Modeling, Performance Analysis and Configuration Simulation in Integrated Supply Chain Network Design, Blacksburg, Virginia, 2001
- Anatoly Levchenkov, Mikhail Gorobetz. The Algorithm of Negotiation for Software Agents for the Open Conveyor Schedule in Logistics Tasks. In: HMS2003, Riga, Latvia, 142-148.

## BIOGRAPHY

**ANATOLY LEVCHENKOV** is Associate Professor of the Logistics and Telematics at Riga Technical University from 2001. Since 1997 he has been Associate Professor at the Transport Railways Institute and from 1992 to 1997 was Head of the Riga Technical University's Department of Decision Support system. He studied Electrical Engineering in Automation and Telemechanics from 1964 to 1969 and from 1969 to 1992 he was a reader and associate professor at the Department of Automated Control and Decision Support Systems at Riga Technical University.

**MIKHAIL GOROBETZ** is graduated Bachelor of Computer Science and Computer Management at Riga Technical University Computer Science and Information Technology Faculty Information Technology Institute in 2003. He started studies at Riga Technical University in 2000. At present he studies for Master of Computer Science grade at Riga Technical University. He participated in Riga Technical University Annual Student Scientific Conference 2003.

# LOGISTIC EXPERT SYSTEMS AND ARTIFICIAL INTELLIGENT IN ELECTRIC POWER

Nadezhda Kunicina\*, Anatoly Levchenkovs\*\*, Leonids Ribickis\*\*\*,

*Faculty of Transport and Mechanical Engineering, \*, \*\**

*Faculty of Electrical and Power Engineering \*\*\**

*Riga Technical University, Riga, Latvia*

*1, Kalku Street*

*Riga, LV - 1658,*

*Latvia*

[kunicina@latnet.lv](mailto:kunicina@latnet.lv)\*

[levas@latnet.lv](mailto:levas@latnet.lv)\*\*\*

[leonids.ribickis@rtu.lv](mailto:leonids.ribickis@rtu.lv)\*\*\*

<http://www.rtu.lv/>

## KEYWORDS

Logistics, expert systems, artificial intelligence, electric power.

## ABSTRACT

The paper considers artificial intelligent application for the solution of the tasks of electric power supply. It analyses logistics systems, decision making, expert systems, and types of multi-agents applications in electric power supply systems.

This paper reviews some questions, which are connected with electric power supply company expert group decision making in logistic processes field, using artificial intelligent methods and tools. In this paper is paying special attention for the expert group consultant activities in the client serving.

The expert group consultant functions in the serving of database.

The expert group consultant activities and functions in the capsule the information by technical condition of processes and example – of the cooperation of expert group in global network environment with technical processes in company. There is an analysis of the results of mentioned system by dispersion analysis data.

The main part of given project is use of dispersion analysis in the organization of power supply process optimization.

## INTRODUCTION

This paper is devoted to the observation of the methods, it analyses a complex structure of the models for electric power supply in the global network. The expert group interaction between themselves and complex of power supply process for the staff models software agents characteristics are considered. The formalized target setting of the model investigation is demonstrated. It is demonstrated in the work that for the solving of the task of software agent modeling the models of management goals, management process and multicriterial evaluation are necessary to be elaborated. For the development and

investigation of the software agents models of electric power supply a complex of nine mutually co-ordinated models is investigated in the paper:

- A functional model of feedstock supply system is shown - Sum;
- Functional model of the transport system for resources delivery to the electrical power producers - Stm;
- Model of electrical power supply to the consumers - Sem;
- Model of electrical energy consumers behavior in the conditions of liberal electrical power market - Spm;
- Model of software agents which interaction with the model of feedstock supply systems solves the task of optimal feedstock supply to power systems - Au;
- Model of software agents which interaction the model of transport system solves the task of optimal feedstock supply to power systems - At ;
- The model of software agents which solves the task of electric power supply to the consumers - Ae;
- The model of consumers software agents which takes into account priority of the consumers, reaching the goals of the real expert groups - Ap.

The model of Supra software agent which provides the reaching of the goal nominated by the expert group with the help of software agents in the processes of electric power supply is analyzed in the work.

## TARGET SETTING

Expert group making decision, about the task of optimal feedstock supply to power systems.

In the evaluation of the given project effectivity a group of three experts takes part. The experts evaluate feedstock supply routes according to the following criteria:

- Price of the route
- Maximum speed of the vehicle
- Technical safety of the transportations

## METHODOLOGY AND TECHNIQUE

Optimal feedstock supply for power station in the Baltic region (fig.1) software for the agent models development and investigation are analysed in the interrelation of nine models, power systems, software agent models, which together with transport solve the problem of optimal raw materials delivery to the power systems, reaching the goals of real expert groups, Supra expert groups

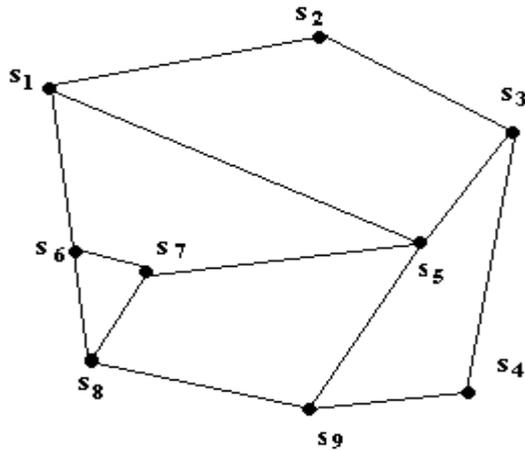


Figure 1. att. Elektroenerģijas lielākie ražotāji Baltijā.

Which take into account the goals of the expert group, providing the realization of consumers aims in an uninterrupted regime. The interrelation of the expert group is in Fig.2.

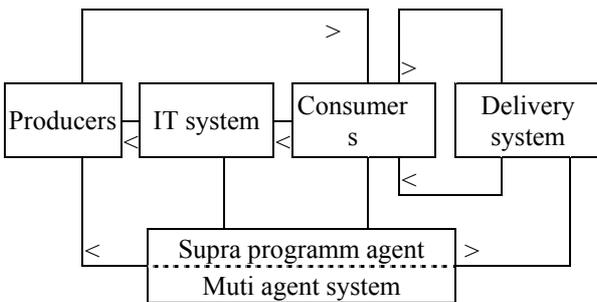


Figure 2 Interrelation of the system.

## PROCEDURE OF EXPERT GROUP MANAGEMENT GOALS

1.part. The expert group management goal definition; 1.step. Asking for management goal  $Z_i$  definition; 2.step. Input goal  $Z_i$  definition (text not longer than gs symbols); 2.part. Objects set  $B$  representation; 3.step. Asking for set  $B$ ; 4.step. Input  $Z_i \in Z$ ; 5.step. Test  $Z_i \in Z_0$ ; If yes go to 8.step; If no go to 6.steo; 6.step. Inform  $Z_i$  in not accepted; 7.step. Define  $i=i-1$ . 8.step. If set  $B$  is input, then go to 9. step; If no then define  $i=i+1$  and go to 4.step; 3.part Representation of set  $W$ ; 9.step. Asking for set  $W$ ; 10.step. Input  $w_i \in W$ ; 11.step. Test  $w_i \in W_0$ ; If yes goto 14.step; If no go to 12.step;12.step. Inform  $w_i$

is not found; 13.step. Define  $j=j-1$ ; 14.step. If  $W$  is input, go to 15.step; If no, define  $j=j+1$  and go to 10.step; 15.step. The end of the algorithm operation.

## TASK SOLVING

There are seven variants of routs suggested F1-F7 (1.tab). Evaluating each route variant an expert makes a decision, the results of the decision making could be demonstrated in the type of the following tables, where horizontals are the routes variants and verticals are the each expert's evaluation of each variant, sign "--" means that this expert did not evaluate this variant. E' means an average result.

1.tab. seven feedstock supply routes variants

	F1	F2	F3	F4	F5	F6	F7
E1	1,0	-	-	-	-	0,125	0,5
E2	0,125	-	0,5	0,5	1,0	-	-
E3	-	0,25	-	-	0,125	1,0	0,5
E'	0,563	0,25	0,5	0,5	0,563	0,563	0,5

Presentative of the expert group defines by Pseido Grandi function for how many points this route is worse than an ideal one with all the carriages and loading technologies, the ideal transporting conditions could be shown in the following way:

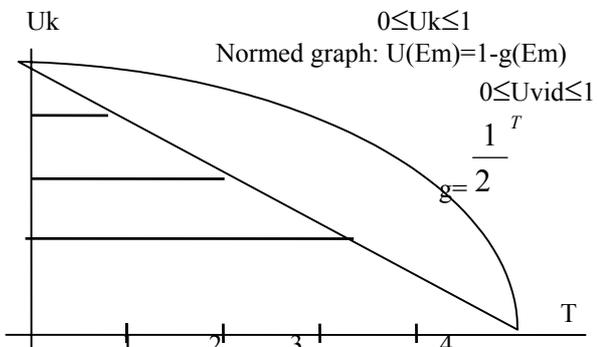


Figure 3. Evaluation of alternative by Pseido Grandi function

At the beginning of the coordinates there is an ideal route and point 1 means that the given alternative is one grade worse than the ideal one, according to this technology a numerical evaluation of each alternative is done. The given data of the expert evaluation (verticals of the graph) have been obtained for the best variants 1,0 and then for the next best station variant 1,00/2, etc. The graph of the solution is the following:

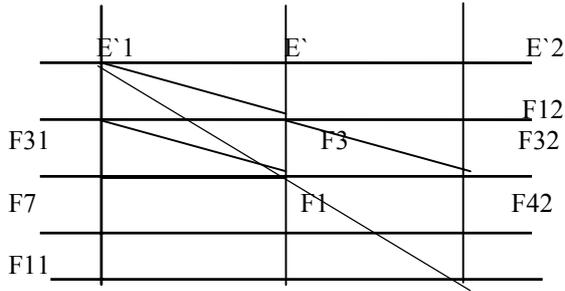


Figure 2 task decision

The given variants of the evaluation follow the evaluation of the route effectivity according to the conventional blocks of methods. At the moment of the decision making experts or responsible persons for the organization of the transportation process take part.

Tādējādi piemērotākais risinājums būs maršruta F3 variants.

The group of experts can give a recommendation to use route F3. the given analysis is needed for evaluation of the route, if the decision of the experts is not univalent. The methodology can be used for the analysis of the other questions.

### APPLICATION OF THE DISPERSION ANALYSIS

Let's analyse factor A k at the levels of A<sub>1</sub>, A<sub>2</sub>, ..., A<sub>k</sub>. for simplification we will assume the number of the observations at all the levels of factor A is n.

The result of the observations at the j level is marked with x<sub>1j</sub>, x<sub>2j</sub>, ..., x<sub>nj</sub>.

The results of all the observations we will sum up in table 2.

Table 2

Number of the observat ion Nr	Level of factor A						Σ
	A <sub>1</sub>	A <sub>2</sub>	...	A <sub>j</sub>	...	A <sub>k</sub>	
2	x <sub>11</sub>	x <sub>12</sub>	...	x <sub>1j</sub>	...	x <sub>1k</sub>	X <sub>1</sub>
...	...	...	...	...	...	...	...
i	x <sub>i1</sub>	x <sub>i2</sub>	...	x <sub>ij</sub>	...	x <sub>ik</sub>	X <sub>i</sub>
...	...	...	...	...	...	...	...
n	x <sub>n1</sub>	x <sub>n2</sub>	...	x <sub>nj</sub>	...	x <sub>nk</sub>	X <sub>n</sub>
Σ	X <sub>1</sub>	X <sub>2</sub>	...	X <sub>j</sub>	...	X <sub>k</sub>	

In this table

$$X_j = \sum_{i=1}^n x_{ij}, \quad X'_i = \sum_{j=1}^k x_{ij} \quad (1)$$

$$x_{ij} = x_0 + \Delta A_j + \Delta X \quad (2)$$

where x<sub>0</sub> – the real value of the X, ΔA<sub>j</sub> – amendment, defined by influence of factor A at the j - level, ΔX – error defined by uncontrolled factor influence. Then we will consider that ΔX distribution is normal and is not depend on ΔA<sub>j</sub>.

We will mark j – that level observation average value  $\bar{x}_j$ , but all the observation with  $\bar{x}$ .

Now we can define all the observed statistical dispersion

$$S^2 = \frac{1}{kn-1} \sum_{i=1}^n \sum_{j=1}^k (x_{ij} - \bar{x})^2 = \frac{1}{kn-1} \left[ \sum_{i=1}^n \sum_{j=1}^k x_{ij}^2 - \frac{1}{kn} \left( \sum_{i=1}^n \sum_{j=1}^k x_{ij} \right)^2 \right] \quad (3)$$

This dispersion defines uncontrolled variable factors and factor A.

As at the level A<sub>j</sub> (j=1,2,...,k) of each factor A there are n observations or parallel observations, it is not difficult to evaluate the influence of the factors on the observation results, as verable value X statistic dispersion S<sup>2</sup><sub>j</sub> of the factor A at the j level defines only uncontrolled variables (dispersion of factor A is constant). Statistic dispersion S<sup>2</sup><sub>j</sub> canbe calculated with formula

$$S_{j}^2 = \frac{1}{n-1} \sum_{i=1}^n (x_{ij} - \bar{x}_j)^2 \quad (4)$$

$$\text{where } \bar{x}_j = \frac{X_j}{n}$$

Let's assume that between the dispersions S<sup>2</sup><sub>j</sub>(j=1,2,...,k) there is no important difference. In this case we can use dispersion S<sup>2</sup><sub>j</sub> to obtain more precise general dispersion D<sub>0</sub>(X)=σ<sup>2</sup><sub>0</sub> evaluation S<sup>2</sup><sub>0</sub>, defined with folmla

$$S_0^2 = \frac{1}{k} \sum_{j=1}^k S_j^2 \frac{1}{k(n-1)} \left[ \sum_{i=1}^n \sum_{j=1}^k x_{ij}^2 - \frac{1}{n} \sum_{j=1}^k \left( \sum_{i=1}^n x_{ij} \right)^2 \right] \quad (5)$$

Knowing the statistic dispersions S<sup>2</sup> un S<sup>2</sup><sub>0</sub>, factor A could approximately define A dispersion D<sub>A</sub>(X)=σ<sup>2</sup><sub>A</sub>. Taking into account that dispersion S<sup>2</sup><sub>0</sub> defines the uncontrolled variable factors, but S<sup>2</sup> defines the uncontrolled variable factors and factor A we can write an approximate expression

$$\sigma_A^2 \approx S^2 - S_0^2. \quad (6.)$$

Evaluation (6.) is very approximate. Factor A could more precisely defined taking into account that separate observation x<sub>ij</sub> dispersion is n times more than dispersion of the average value  $\bar{x}_j$ . In other words average value of the separate level  $\bar{x}_j$ , which dispersion is n times less that separate observation dispersion, ia more sensible to the influence of factor A. Taking into account all mentioned above:

$$\sigma_A^2 \approx \frac{1}{k-1} \sum_{j=1}^k (\bar{x}_j - \bar{x})^2 - \frac{S_0^2}{n} \quad (7)$$

Both parts of (5.9.) we will equate to n then we can obtain

$$S_A^2 \approx n\sigma_A^2 + S_0^2 \quad (8.)$$

For the statistic dispersion S<sup>2</sup><sub>A</sub> is k-1 range of discretion. Influence of factor A is considerable only if statistic dispersion S<sup>2</sup><sub>A</sub> differs from S<sup>2</sup><sub>0</sub>.

The question of the considerable or non-considerable dispersion difference  $S_A^2, S_0^2$  can be clear with Fisher criterion. According to the criterion dispersions  $S_A^2, S_0^2$  differ considerably if

$$\frac{S_A^2}{S_0^2} > F_{1-\alpha}(f_A, f_0), \quad (9.)$$

### NUMERICAL EXAMPLE

Effectivity of transport application can be evaluated with dispersion analysis. Types of the transport are in accordance with four cargo types. We will consider whether the type of the cargo considerably influences the using factor of the transport.

Table 3. Using factors of the transport types

Type of the transport	Type of the cargo			
veids	K1	K2	K3	K4
T1		0.35	0.80	0.80
T2	0.65	0.40	0.76	0.95
T3	0.72	0.55	0.79	0.84
T4	0.68	0.47	0.81	0.89
T5	0.69	0.53	0.75	0.93
T6	0.70	0.45	0.80	0.90

**Solution:** factors A – K1., K2, K3, K4i, - at the levels of factor A.  $x_{ij}$  using factor of the transport types. For calculation of all observations square sum  $Q_1, Q_2, Q_3$  first of all we will consider the necessary data for the these calculations in table 4.

Table 4. Level of factor A levels, valuation of the experts.

i	$x_{i1}$	$x_{i1}^2$	$x_{i2}$	$x_{i2}^2$	$x_{i3}$	$x_{i3}^2$	$x_{i4}$	$x_{i4}^2$
1	0.70	0.490	0.35	0.123	0.80	0.640	0.80	0.640
2	0.65	0.426	0.40	0.160	0.76	0.578	0.95	0.903
3	0.72	0.518	0.55	0.303	0.79	0.624	0.84	0.706
4	0.68	0.462	0.47	0.221	0.81	0.656	0.89	0.792
5	0.69	0.476	0.53	0.281	0.75	0.563	0.93	0.865
6	0.70	0.490	0.45	0.205	0.80	0.640	0.90	0.810
$\Sigma$	4.14	2.859	2.75	1.289	4.71	3.700	5.31	4.715

The following calculation are the next: we will apply the following symbols: n - is i maximum value (i =6), k - is j maximum value (k = 4).

$$Q_1 = \sum_{i=1}^6 \sum_{j=1}^4 x_{ij}^2 = 2.8594 + 1.2893 + 3.7003 + 4.7151 = 12.56$$

$$Q_2 = \frac{1}{6} \sum_{j=1}^4 x_j^2 = 12.51$$

$$Q_3 = \frac{1}{24} \left[ \sum_{i=1}^6 \sum_{j=1}^4 x_{ij} \right]^2 = \frac{1}{24} (4.14 + 2.75 + 4.71 + 5.31)^2 = 11.91$$

Statistic dispersion :

$$S_0^2 = \frac{Q_1 - Q_2}{k(n-1)} = \frac{12.56 - 12.51}{4(6-1)} = 0.0025$$

$$S_A^2 = \frac{Q_2 - Q_3}{k-1} = \frac{12.51 - 11.91}{4-1} = 0.2$$

The level of importance  $\alpha = 0.05$ . An expected probability is  $\beta = 1 - \alpha = 0.95$ . Ratio

$$\frac{S_A^2}{S_0^2} = \frac{0.2}{0.0025} = 80. \text{ In the given case Fisher}$$

distribution  $F_{1-\alpha}(3;20) = 3.10$ . As  $\frac{S_A^2}{S_0^2} > F_{1-\alpha}(3;20)$ , it

can be concluded that type of the cargo considerably influence the using factor of the transport types.

Calculation of the dispersion evaluation of factor A:

$$\sigma_A^2 \approx \frac{S_A^2 - S_0^2}{n} = \frac{0.2 - 0.0025}{6} = 0.033$$

We can say that with probability 0.95 the cargo considerably influence the infill factor of the transport types.

According to the results of the experiment we can say that there is a considerable influence of the cargo type as well as difference of the given means of transport.

### CONCLUSIONS AND FURTHER RESEARCH AND RECOMMENDATIONS.

Scientific novelty of the research covers the following aspects of the work: The task of the development of the software agents modelling methods for the logistic systems of electric power supply in the situation of global networks has been formulated. Operation methods and procedures of multicriterial software agents in risky conditions have been developed. Procedures of evaluation for Supra agents formalized goals models are developed: for consumers profile choice; for making of feedstock supplying schedules; for the expert commissions forming and their interaction with Supra agents for logistic expert systems and artificial intelligent in electric power.

## REFERENCES

- Борисов, А. Н., Левченков, А. С., Методы интерактивной оценки решений, Зинатне, Рига, 1982, 139 стр.
- Дале В. А., Кришан З. П., Паэгле О.Г., “Динамические методы анализа развития сетей энергосистемы” Зинатне, Рига, 1979, 256 стр.
- Grundspenkis, J., Isajeva, L., Qualitative analysis of port organisational structure characteristics, HMS 2003, Riga, 2003, 7p.
- Jakson, M.,A., Principles of Program Design. London: Academic Press, 1975., 300p.
- Jonker, C., Treur, J., Agent – based simulation of animal behaviour: Amsterdam, 1998, 26, 7p.
- Khosla, R., Dillon T., Engineering intelligent hybrid multi-agent systems, Kluwer Academic Publishers, Boston/Dordrecht/London, 1997, 410p.
- Kim, K., Roush, F., Them theory; Ellis Hordwood limited, England, 1987., 241p.
- Kunicina, N., “Globālo tīklu transporta loģistikas grupas konsultanta automatizētās darba vietas projektēšana” inženierprojekts, RTU, Riga, Latvia, 2000 75p.
- Kunicina, N., Levchenkovs, A., Ribickis, L., “Algorithm for software agents to power supply modeling in Baltic region” EPE – PEMC, 11 th International power electronics and motion control conference, 2004., 8 p.
- Krishans, Z., Oleinikova, I., Mutule, A., New Technologies for Long - Term Management of Power Systems Nr.5 2003.
- Ларичев О. И., “ Объективные модели и субъективные решения” Москва, Наука, 1987., 140 стр.
- “Latvenergo” VAS gada pārskats 2002. g. 66p.
- Linkevics O., Balodis M., Bockarjova M., Sauhats A., Chuvychin V., Svalova I., Will Distributed Energy Recources Make a Revolution in the Power Sectoe of Baltic States. EPE – PEMC, 11 th International power electronics and motion control conference, 2004., 9.lpp.
- Лукинский, В. С., Модели и методы теории логистики – СПб.: Питер, 2003., 176 стр.
- Leontjevs L., Plaudis A., “Inženierekonomisko aprēķinu matemātiskās metodes” Zvaigzne, Rīga, 1976., 452 lpp.
- Gutmanis, A., Kabashkin, I., Merkuryev, Y., Reseach and education in freight transport logistocs in Latvia in HMS 2003 Riga, Latvia, 256 – 259p.
- Meystel, A., Intelligent systems: architecture, design, and control / Alexander M. Mystel, James S. Albus. New York, Wilwy – Interorice, 2002., 696p.
- Praca, I., Ramos, C., Valte, Z., Modelling and simulation of electricity markets game theory to improve decision support // In p. 419 - 421.
- Praude, V., Belcikovs, J., Loģistika. Rīga, Vaidelote, 2003., 541 p.
- Ribickis L., Galkin I., Direct Frequency Converters for Induction Motor Drives //ISIE '99 – Proceedings of the

- IEEE International Symposium on Industrial Electronics, 12-16 July, 1999, Bled, Slovenia, 1999, Vol.1, p.26-30.
- Saaty, T. L, Multicriteria Decision Making: The Analytic Hierarchy Process, RWS Publications, 4922 Ellsworth Avenue, Pittsburgh, PA, 1990.
- Sprancmanis N., Transporta pakalpojumu ekonomika un organizācija, RTU, Rīga, Latvija, 2001, 283p.
- Witting, T., ARCHON: an architecture for multi agent systems; Ellis Hordwood limited, England, 1992., 135p .
- Williams, E., Harp, V., Improving logistical procedure within a hospital inpatient pharmacy // In The International Workshop on Harbor, Maritime amd Multimodal Logistics Modelling & Simulation (HMS 2003) proceedings, 2003., p. 137 - 141.

## AUTHOR BIOGRAPHIES



**NADEZDA KUNICINA** In 2004 – Doctor of Science in Electrical Engineering “Development of modelling methods of software agents in electric power supply and transport logistic systems”.

From 2005.01.02 RTU as professors Assistant, Logistics Group Decision Support Systems., Riga Technical University, 1, Kalku Street, LV –1658 Riga, Latvia, e-mail: kunicina@latnet.lv.



**ANATOLY LEVCHENKOV**, 1978. Dr.sc.ing.degree in Technical Cybernetics, Riga Polytechnical Institute. Theme of the doctoral thesis: "Investigation and Development of Repeated Decision

Making Models in Organisational Management Systems".Research interests: group decision support systems, management decisions, transportation problems, scheduling, intelligent logistics transport systems. Associate Professor, Logistics Group Decision Support Systems, Riga Technical University, 1, Kalku Street, Riga, LV-1658, Riga, Latvia, e-mail: [levas@latnet.lv](mailto:levas@latnet.lv)



**LEONIDS RIBICKIS**, Riga Technical University, 1994, Member, Latvian Council of Science, Vice-Rector for Research Riga Technical University, Professor,

Director, Institute of Industrial Electronics and Electrical Drives, esearch interests: Power Electronics, Electrical Drives, Electrical Machines, Process Control Systems. Riga Technical University, 1, Kalku Street, LV –1658 Riga, Latvia, e-mail: leonids.ribickis@rtu.lv.

# MULTIAGENT SYSTEM MODELING IN LOGISTICS TASKS FOR EMERGENCY SITUATIONS

**Anatoly Levchenkov**

Transport and Machinery Faculty  
Riga Technical University  
1, Kalku St.,  
LV-1658 Riga, Latvia  
E-mail: [levas@latnet.lv](mailto:levas@latnet.lv)

**Violeta Medne**

Computer Science and Information Technology Faculty  
Riga Technical University  
1, Kalku St.,  
LV-1658 Riga, Latvia  
E-mail: [vm28@inbox.lv](mailto:vm28@inbox.lv)

## INTRODUCTION

As an emergency situation we can take the situation which was not planned or expected from the schedule or route and which demonstrates replacing problems. The problems could occur in the following situations: closed street, closed bridge, absence of electricity, faults with traffic lights, obstacles on the road etc.

For the solution of these problems Transport Management system is to be developed (TMS).

The mathematic model is a graph where the curves are streets and peaks are crossings of the roads. For replanning the schedule Deijkstra algorithm is used for the shortest way searching from a particular point till the point of destination.

Monte-Carlo method is applied for the for modeling the system of environment development.

For the assessment of model's correspondence to the real situation Flash tool is applied.

## TRANSPORT MANAGEMENT SYSTEM

Transport Management system is demonstrated in Fig.1. TMS shows the situation on the roads, taking facts and information, which comes from different sources and notifies on unexpected events on the road.

If this information is obtained TMS proves it, defines the environment situation ,

Sensors are installed at the problematic places of the streets. As the sensors laser

its development automatically asks for additional information, defines the importance of the situation. If the condition is important it automatically switches to the Crisis Management system (CMS). It uses the information, applies the decision making system (DMS), which works with the prognosing of the development course and/or duration and makes a decision on replanning of schedule and/or defining additional transport and/or safety brigadge call. Information System (IS) contains actual IS and Geographic IS (GIS). It consists of all types of information and objects, all transport schedules and routes as well as information of safety service and other. On the basis of the knowledge the information is processed from different sources for the clear and comprehensively idea of the present situation.

TMS obtains data using agents. Co-operation with other agents goes through interface agents.

Information and Communication System (ICS) agents operate with information collection from sensors and other systems, information storage, knowledge reflection, which allows to the systems analyse the knowledge and data transporting through the communication structure.

Subject agents have enough knowledge in the special area for example weather.

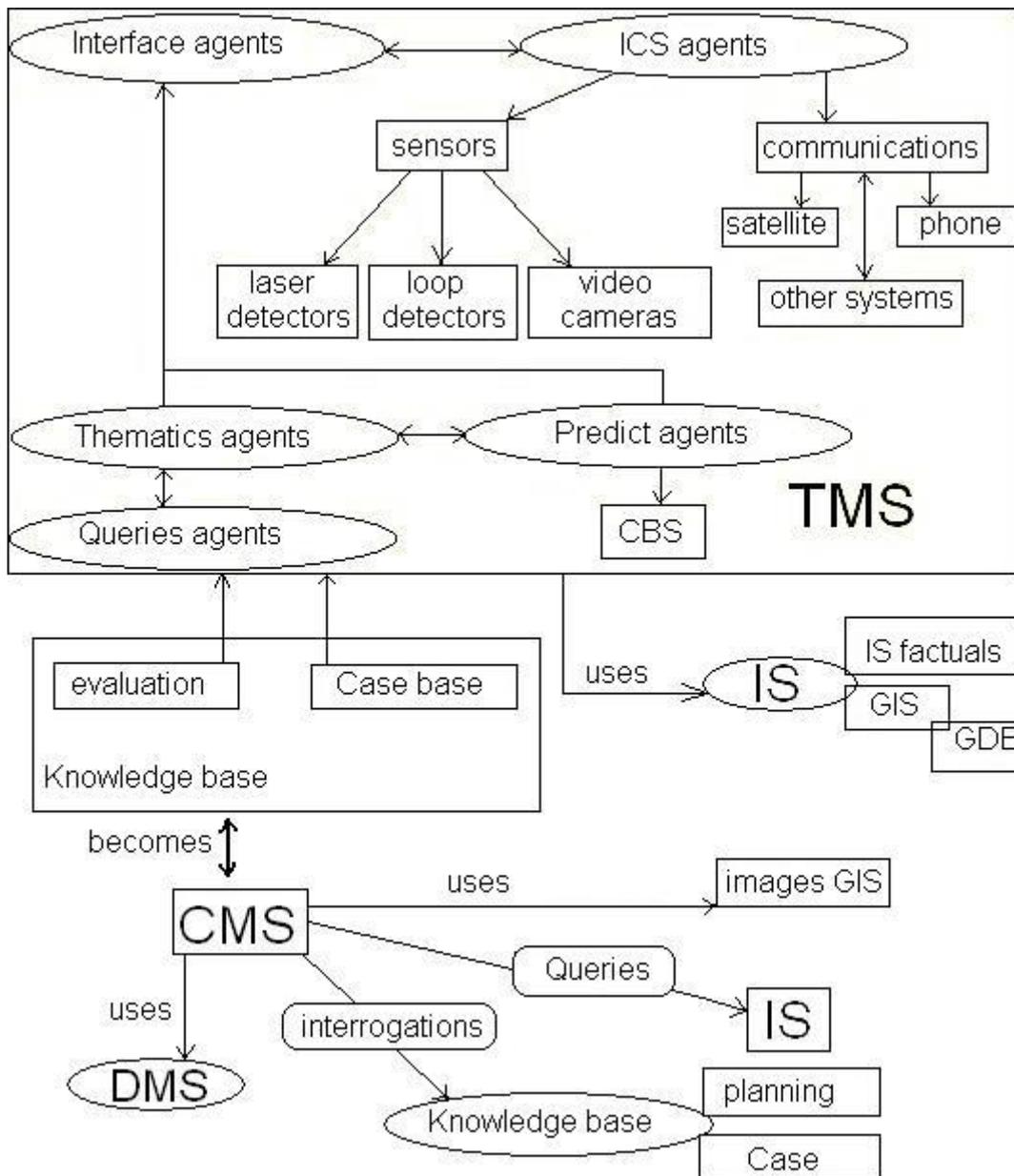
For obtaining data from IS question agents are applied.

Prognosing agents forecast the following situations using information from subject agents, IS and Case based systems (CBS).

detectors are used , which are appllied with ICS agent processing at the real time data,

identify transport, size and speed, define unexpected cases, time and transport density and send the information to TMS, videocameras, with which ICS agents

distinguish unexpected cases and send visual information, loop detectors with the help of which admission time is calculated.



1. figure. Transport Management System

### MATHEMATIC MODEL

Mathematic model is developed for Riga region which contains buses, trolleybuses and trams.

This Riga region is shown in Fig.2. With the help of this map A correspondent graph was developed where the streets are curves and peaks are crossings of the roads. The graph is in Fig.3.

The peaks are marked with letters A\* for the streets which correspond to some routes

and letter B\* for the streets, which could be used for the additional routes. In the graph with oriented pointer only those roads are shown which are one-way streets, the others can be both-directions.

If at any cross (peak) an emergency situation is developed, then this peak is not appropriate and the curves (streets) which go to this peak are not applied.

At the crossroads sensors are installed. Laser detectors are installed at all the peaks of the graph A\*, additional road (loop)

detectors are installed also  $B^*$ , but videocameras are at all the important crosses on the basic roads – they are the graph peaks: AG, AH, AI, AJ, AK and AL.

The goal is during minimum time through the crossroad where an emergency situation is defined. Goal function:

$$M = \min [t_{Ai}, t_{An}],$$

where  $M$  – route,  
 $t$  – time,  
 $A_i$  – starting peak,  
 $A_n$  – final peak.

Sensors of additional roads measure the speed of the transport and with the using of ICS calculate admission time :

$$T = S / V,$$

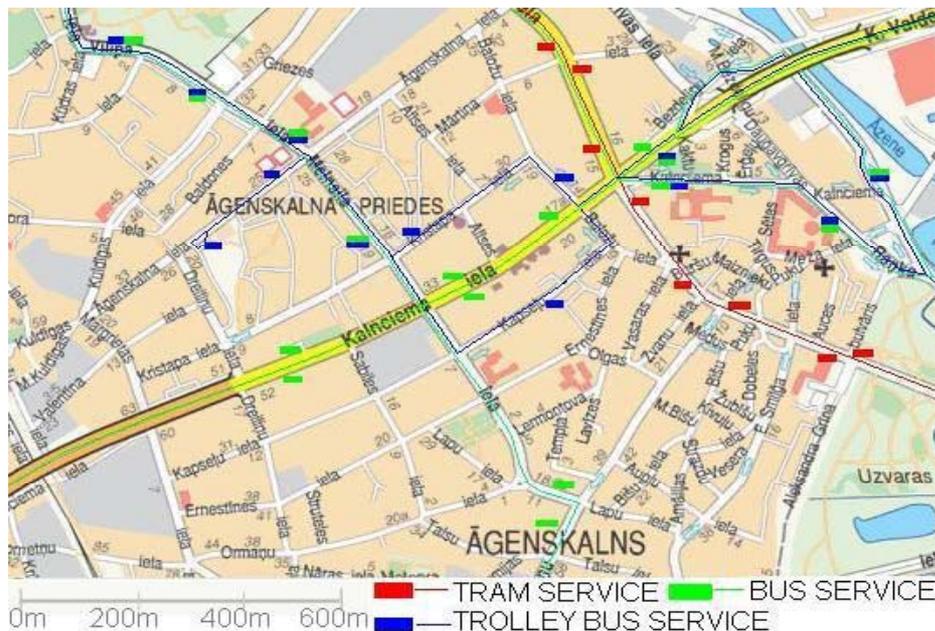
where  $T$  – admission time,  
 $S$  – length of the curve,  
 $V$  – speed of the vehicle.

The obtained  $T$  are used as variable of Deiktra algorithm. With this algorithm the shortest way is calculated:

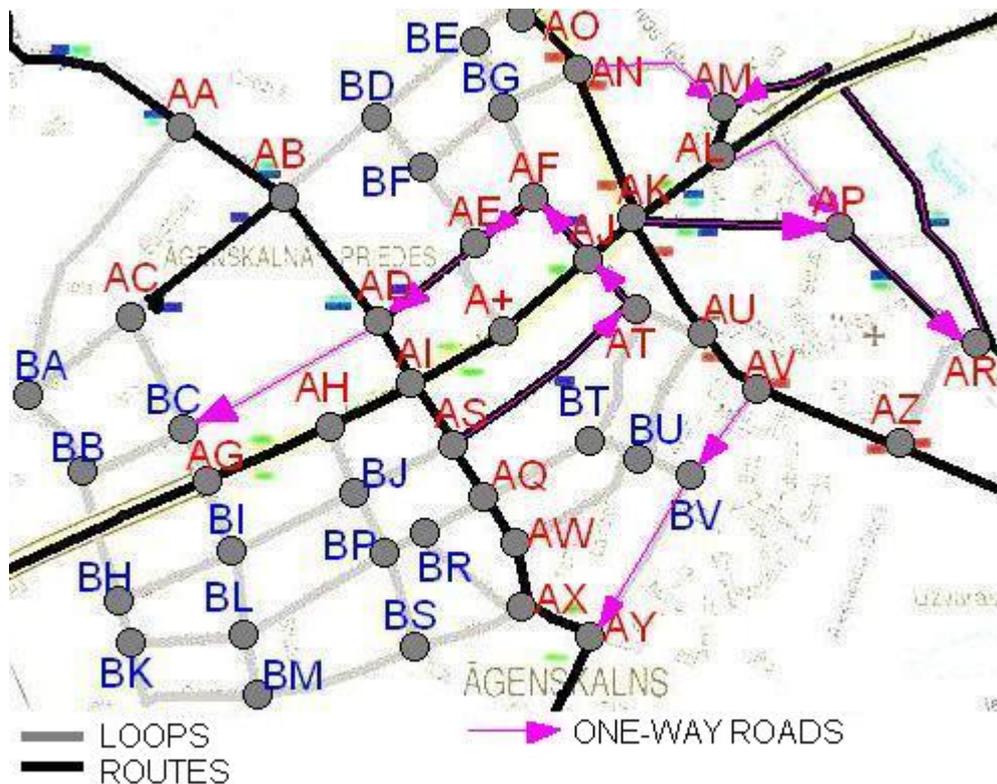
$$T(A_j) = \min [T(A_j), T(p) + w(p, T(A_j))],$$

where  $T(A_j)$  – variable for the considered peak,  
 $p$  – variable peak which can be considered or not considered at the shortest way.

$w$  – weight between the peaks.  
 if  $T(p) + w(p, A_j) < T(A_j)$   
 then  $T(A_j) := T(p) + w(p, A_j)$



2. fig. Riga map



3. fig. Graph of the roads

### BASIC ALGORITHMS OF THE OPERATION

1. Defining of the environment situation place;
2. Defining of the environment situation type;
3. Difficulty level of the situation;
4. Prognosing of the development course and/or duration;
5. Decision on replanning of schedule and/or defining additional transport and/or safety brigadze call.;
6. Defining of the additional roads admission time;
7. Replanning of schedule with the help of Deiktra algorithm;
8. Returning to point 4.

### DIGITAL EXAMPLE

We develop the environment situation using Monte Carlo method. For example, the emergency situation happened at 11:14. The type of situation as accident, very hard, the place is crossroad – AJ.

Police and ambulance are called. The forecasted duration by the release of the cross is 99 minutes. The motion could be

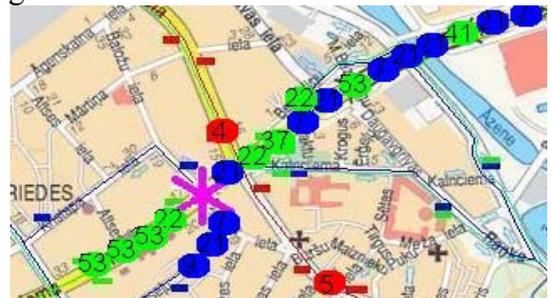
seen on Web using tool Macromedia Flash MX.

Fig.4 demonstrates the situation at the moment of the accident – the place of the vehicles at that moment.



4. fig. Accident

If TMS is not applied that in an hour situation will achieve traffic jam situation , fig.5.





- J. Grundspenķis. Lekciju konspekts mācību priekšmetā – Diskrētās Struktūras Datorzinātnēs. RTU,2001
- S. Halle, B. Chaib-draa, J. Laumonier. Car Platoons Simulated as a Multiagent System.  
<http://www.damas.ift.ulaval.ca/projects/auto21>
- J.Zutt, C. Witteveen. Multi-Agent Transport Planning.  
<http://www.ai.rug.nl/conf/bnaic2004>
- Application:Agent-Based Real-Time Traffic Detection and Management System.  
<http://iel.ucdavis.edu/projects/mobilec>
- P. Davidsson, L. Henesey, L. Ramstedt, J. Tornquist, F. Wernstedt. Agent-Based Approaches to Transport Logistics.  
<http://www.ipd.bth.se/lhe/ABATL>
- H. Boukachour, A. Cardon, C. Duvallet. Multiagent System to Prevent Technological Risks. <http://www-lih.univ-leavre.fr/~duvallet>

#### **AUTHOR BIBLIOGRAPHIES**

**Anatoly Levchencov** is Associate Professor of the Logistics and Telematics at Riga Technical University from 2001. Since

1997 he has been Associate Professor at the Transport Railways Institute and from 1992 to 1997 was Head of the Riga Technical University's Department of Decision Support system. He studied Electrical Engineering in Automation and Telemechanics from 1964 to 1969 and from 1969 to 1992 he was a reader and associate profesor at the Department of Automated Control and Decision Support Systems at Riga Technical University.

**Violeta Medne** is graduated Bachelor of Computer Science and Computer Management at Riga Technical University Computer Science and Information Technology Faculty Information Technology Institute in 2003. She started studies at Riga Technical University in 1999. At present she studies for Master of Computer Science grade at Riga Technical University. She participated in Riga Technical University Annual Student Scientific Conference 2005.

# COMBINING ANALYTICAL AND SIMULATION APPROACHES TO QUANTIFICATION OF THE BULLWHIP EFFECT

Julija Petuhova and Yuri Merkurjev  
Department of Modelling and Simulation  
Riga Technical University  
1 Kalku Street, LV-1658 Riga, Latvia  
E-mail: julija@itl.rtu.lv, merkur@itl.rtu.lv

## KEYWORDS

Simulation, Inventory control, Bullwhip effect.

## ABSTRACT

Nowadays, effective and competitive company operation can be achieved through incorporating the concept of supply chain operation into company management. Inventory control, as a critical part of the supply chain management, becomes the second most frequent application area for simulation technique in logistics (after manufacturing).

The dynamics of supply chain operation is characterised by the bullwhip effect that reflects an increase in demand variability while moving upwards the supply chain. This paper proposes an analytical model for the analysis and numerical evaluation of the bullwhip effect in supply chains. Simulation technique is used to validate the results obtained from the analytical model. Based on the validation results, the logic of the analytical model is examined, and some specifications of the analytical model are analysed and described.

## INTRODUCTION

Supply chain management is the term used to describe the management of materials and information across the entire supply chain, from suppliers to component producers to final assemblers to distribution (warehouse and retailers), and ultimately to the consumer. Supply chain management has generated much interest in recent years because of the realisation that actions taken by one member of the chain can influence the profitability of all others in the chain (Silver and Peterson 1985). The bullwhip effect is considered as one of the main supply chain operation stability and efficiency measures. It characterises an increase in demand variability through the entire supply chain.

Many companies implement the supply chain concept to achieve efficiency in system operation; i.e., instead of responding to unknown and highly variable demand, they share information so that the variability of the demand they observe is significantly lower. The assumption that a new level of efficiency can be simply attained by sharing information and forming *strategic alliances* with firm supply chain partners is wrong. Knowing what to do with the data is as important as

getting the data in the first place (Silver and Peterson 1985). Methods for coping with the bullwhip effect are discussed in (Simchi-Levi et al. 2000). They can significantly reduce, but will not eliminate, the bullwhip effect. It is important to investigate the magnitude of this effect to avoid holding an excessive inventory, insufficient capacities and high transportation costs. For better understanding and controlling the bullwhip effect it is useful to quantify it. Simchi-Levi et al. (2000) explain the increase in demand variability by the necessity for each supply chain stage to make orders based on the forecasted demand of the previous stage. They propose quantifying the magnitude of increase in variability between two neighbour supply chain stages by a function of the lead-time between the orders receipt and the number of demand observations on which a forecast is made. Disney and Towill (2002) developed an analytical expression for the bullwhip effect quantification from the control theory's point of view by using a z-transform model. Kelle and Milne (1999) suggest using approximations of the quantitative model, developed in accordance with asymptotic renewal theory, to evaluate a variance of placed orders (bullwhip effect) in inventory systems that implement the *S-s* inventory control policy.

This paper proposes a statistics-based analytical approach for evaluating the bullwhip effect in inventory systems. We focus on the supply chain from the perspective of inventory management. We consider the simplest of multi-echelon situations when the stocking points are serially connected.

The main cause of the bullwhip appearance in supply chains is uncertainty of demand inherent in supply chain operation environment. An analytical model for quantification of demand fluctuation magnification (the bullwhip effect) as orders move up the supply chain in case of stochastic demand is developed in this paper.

Simulation is a powerful tool for analysing inventory systems, because it is capable of capturing the uncertainty and complexity inherent in inventory systems. The ability to handle demand and lead time uncertainty is one of the main reasons why simulation is widely used for inventory systems (Bhaskaran 1998).

Banks and Malave (1984) identify inventory control problems as one of the most frequent areas of application for simulation methodology. They propose the following six categories of simulation techniques

usage assignments in modelling and analysing inventory systems:

1. Analytic solution impossible or analytic solution extremely complex. An analytic solution to a problem may not be available because of stochastic operating environment, extremely complex problem or a very specific problem.
2. Comparison of model. It is one of the most frequently observed uses of simulation in inventory systems. Simulation is used to compare alternative inventory control policies.
3. Validation of analytical solution. Simulation is used to validate the results obtained from an analytic model.
4. Variance reduction techniques. Increasing the statistical efficiency of a simulation by reducing the variance of the output random variables.
5. Model validation and verification. It is the most important part of a simulation study and enables determining whether a model performs as intended and is an accurate representation of the real-world system under study.
6. Optimisation techniques. Considering optimisation techniques for inventory simulation two aspects should be determined: the length of simulation run and comparison method of different alternatives.

The developed simulation model of the considered inventory system validates the results produced by the analytical model. The simulation implementation in this case corresponds to the 3<sup>rd</sup> category of simulation techniques usage assignments in modelling and analysing inventory systems.

The rest of the paper is organised as follows. The analytical model for numerical evaluation of the bullwhip effect in inventory systems that control their inventories by the *S-s* ordering policy is elaborated in the next section. The section also presents a sample application of the described analytical model aimed to get numerical results. The following sections relate to analytical model validation performed by using an appropriate simulation model, analyse the accuracy of the obtained analytical solution, and discuss a combined analytical/simulation approach for evaluating the increase in variability of placed orders in supply chains. Conclusions are presented in the final section.

## ANALYTICAL MODEL

A single-item, single-stage, multi-period inventory system is considered. The traditional *S-s* policy is used for inventory management. According to this policy, an order is placed when the stock declines to a lower control limit called the order point, *s*. The order quantity is the amount required to bring the inventory level to the order level, *S*. A more detailed description of the considered inventory control policy can be found in Merkurjev et al. (2004).

It is assumed that the demand  $X_1, X_2, \dots, X_i$  is a discrete random sample observed from some population. Accordingly, these data are independent and

identically distributed (IID) observations on some underlying random variable  $X$  whose distribution governs the population. Values that numerically characterise the population/distribution, such as an expected value  $E(X)$  and a variance  $D(X)$  of the discrete random variable  $X$  are given. The order quantity  $Q_i$  is demanded when the on-hand inventory drops below the reorder point. It is equal to the sum of the demand quantities between the order placements:

$$Q_i = X_1 + X_i + \dots + X_v, \quad (1)$$

where

$v$  – random variable, a period number when an order is placed.

Provided that the demand  $X$  is uncertain and the aforementioned inventory control method is employed, the placed order quantity  $Q$  is expected to be a random variable that depends on the demand quantities. The expected value  $E(Q)$  and variance  $D(Q)$  of the function  $Q = \varphi(X)$  are estimated using the following formulas proposed by Feller (1967):

$$E(Q) = E(X) * E(v) \quad (2)$$

and

$$D(Q) = E(v) * D(X) + D(v) * [E(X)]^2, \quad (3)$$

where

$E(v)$  – expected value of a period number when an order is placed;

$D(v)$  – variance of a period number when an order is placed.

To investigate a probabilistic behaviour of the discrete random variable  $v$  it suffices to estimate its numerical characteristics (an expected value and its variance). The difference between the order level  $S$  and order point  $s$  has to be established to find a time period when an order should be placed:

$$\Delta = S - s \quad (4)$$

The multi-experimental realisation of the following algorithm:

if  $X_1 > \Delta$  THEN  $v=1$  AND STOP

ELSE generate  $X_2$

if  $X_1 < \Delta$  and  $X_1 + X_2 > \Delta$  THEN  $v=2$  AND STOP

ELSE generate  $X_3$

...

if  $X_1 + X_2 + \dots + X_{n-1} < \Delta$  and  $X_1 + X_2 + \dots + X_n > \Delta$  THEN  $v=n$

STOP

allows one to collect statistics of  $v$  values ( $v_i, i = \overline{1, n}$ ) and evaluate their probabilities  $p_i$  by relative frequencies  $\hat{p}_i$  of their occurrences in the experiments performed.

The expected value of random variable is the weighted average of all possible values of the random variable, where the weights are the probabilities of the

value occurrence. The expected value  $E(v)$  of the  $v$  value population is estimated by this formula:

$$\hat{E}(v) = \sum_{i=1}^n v_i * \hat{p}_i \quad (5)$$

and its variance  $D(v)$  is estimated as follows:

$$\hat{D}(v) = \sum_{i=1}^n v_i^2 * \hat{p}_i - \hat{E}(v)^2, \quad (6)$$

where

$\hat{E}(v)$  and  $\hat{D}(v)$  - experimental estimation of  $E(v)$  and  $D(v)$ , respectively.

A numerical example of the developed analytical model implementation for the bullwhip effect quantification is given in the next subsection.

### Sample Application of Analytical Model

The performance of the inventory system is evaluated under various factors such as end customer mean demand  $E(X)$  and its standard deviation  $STD(X)$ , safety stock factor  $z$  and a lead time  $LT$ .

To collect a statistics of period numbers  $v$  when orders should be placed and experimentally estimate their expected value by formula (5) and variance by formula (6), it is supposed that the end customer demand is realised as a normal distribution, and 1000 experiments are performed. The minimal value of the observed  $v$  values for all alternatives is 1 time period and its relative frequency of occurrence is less than 0.007. The maximal value is 5 time periods and its relative frequency of occurrence does not exceed 0.004. Respectively, the most likely value is 2 time periods that can occur with the relative frequency greater than 0.5.

Experiments, when the standard deviation of the mean demand changes by the defined coefficient *Change Ratio* equal to 1.2 and remaining factors are considered to be constant numbers, are performed. The mean  $E(X)$  and the standard deviation  $STD(X)$  of the demand change proportionally, i.e. they are dependent through the *Signal To Noise* factor, equal to 5, that describes a variability of the demand:

$$STD(X) = \frac{E(X)}{\text{Signal To Noise}} \quad (7)$$

The experimentally estimated probability of the period number when an order is placed will be the same

for all alternatives because the  $S$ - $s$  level will change in accordance with a new mean demand value. Based on the observed experimental results the following hypothesis could be built up – the relative frequency  $\hat{p}$  of the random variable  $v$  occurrence corresponds to its probability  $p$  and its value depends only on the lead time length.

The estimation of orders variability  $D(Q)$  and its expected value  $E(Q)$  are calculated by formulas (3) and (2) respectively, while numerical results are given in Table 1.

By analysing the placed order variability for all the performed experiments we can conclude that even a small variation of the mean demand causes an increase in variability of the placed orders. The larger the initial value of the demand variation is, the more significant magnification of placed orders fluctuation will be observed.

### VALIDATION OF ANALYTICAL MODEL

The considered inventory system has an explicitly dynamic character. Simulation is used to capture this behaviour of the system and to provide a more realistic representation of the inventory system operation, namely information about demand and order quantities collection over time.

The developed simulation model was used to validate the analytical solution presented in the previous section.

### Conceptual Model of Inventory System

The structure of the considered inventory system corresponds to the analytical model described above.

It is assumed that end customer demands arrive with fixed time-intervals, and their quantity is variable and is derived from a normal distribution. A constant lead time between replenishment is considered. No order processing delay is taken into account, so all demand events are treated immediately by the inventory system. We will also assume no capacity constraints for supplier of the inventory system. In this case, stockouts will not lead to lost sales, but to backorders. We thus assume that we have loyal customers. It is worth noting that replenishment triggering will be based on the effective inventory level, which is the quantity on hand plus the quantity on order minus the unshipped backorders to customers.

Table 1: Placed Orders Variability Estimation by Analytical Model

Nr.	$E(X)$	$STD(X)$	$z$	$LT$	$s$	$S$	$S-s$	$\hat{E}(v)$	$\hat{D}(v)$	$D_{cal}(Q)$	$E_{cal}(Q)$
1	50	10	1.96	2	128	228	100	2.50	0.26	885.64	126.60
2	70	14	1.96	2	179	319	140	2.50	0.26	1743.30	178.15
3	90	18	1.96	2	230	410	180	2.50	0.26	2881.78	229.05
4	110	22	1.96	2	281	501	220	2.50	0.26	4304.88	279.95
5	130	26	1.96	2	332	592	260	2.50	0.26	6012.60	330.85

The objective of inventory management is to manage stable operation of the considered system, i.e., quantify and control the bullwhip effect.

### Simulation Model of Inventory System

The simulation model was developed using the ARENA 5.0 simulation modelling environment. The described conceptual model is converted into a computer model. Simulation is used to analyse and evaluate the increase in variability of placed orders in the described inventory system.

### Simulation Experiments

The objective of experimental studies is to determine the bullwhip effect magnitude in the inventory system that implements the  $S-s$  inventory control policy and validate the results produced by the analytical model of the same inventory system. For that purpose, a set of experiments with the simulation model is performed. The performance of the inventory system is evaluated

under various factors similar to the sample application of the analytical model for the quantification of the bullwhip effect (see Table 1).

The model was run for 5 replications. Each replication length is defined as 5000 time periods. The warm-up period is avoided by setting the initial inventory level equal to the order level  $S$  at the beginning of each replication.

The mean value and variance of placed orders during simulation are shown in Table 2.

Considering the first row in Table 2, one could observe that in this case 95% confidential intervals for the expected average order quantity  $E_{sim}(Q)$  and its variation  $D_{sim}(Q)$  are  $[125,78 \pm 0,9]$  and  $[311,77 \pm 4,5]$ , respectively. Generally speaking, observations within all 5 series of simulation experiments vary non-essentially. Namely, for each row in Table 2 a 95% confidential interval for variation of placed orders lies within 1.5% from its average value.

Table 2: Placed Orders Variability Estimation by Simulation Model

Nr.	$E(X)$	$STD(X)$	$z$	$LT$	$s$	$S$	$S-s$	$D_{sim}(Q)$	$E_{sim}(Q)$
1	50	10	1.96	2	128	228	100	311.77	125.78
2	70	14	1.96	2	179	319	140	603.03	176.53
3	90	18	1.96	2	230	410	180	996.55	226.75
4	110	22	1.96	2	281	501	220	1501.83	277.63
5	130	26	1.96	2	332	592	260	2082.93	328.10

### ANALYSIS OF SIMULATION RESULTS

The results given by the analytical model proved to be in disagreement with those given by the simulation model. The variance of placed orders calculated by analytical model (see Table 1) is approximately 3 times greater in all experiments than actual variance of placed orders derived from the simulation model (see Table 2). The reason for the inadequate bullwhip effect quantification by the analytical model is an existing dependence between a period number when an order is placed  $v$  and realisations of the end demand  $X_i$ . In other words, the proposed formula (3) assumes  $v$  and  $X$  independence, but in the described inventory control system they are dependent in the way of conditional probability of  $v$  occurrence  $p_v = P(X_1 + X_2 + \dots + X_v > S-s / X_1 + X_2 + \dots + X_v - I < S-s)$ . The period number when an order is placed directly depends on the demand quantity (the larger the demand quantity during the order cycle is, the faster inventory level reaches the order point  $s$  and frequency of orders increases; i.e,  $v$  decreases). From this it follows that random variables  $v$  and  $Q = \sum_{i=1}^v X_i$  are correlated. Random variables  $Q$  and  $v$  that are denoted by the expected values and standard

deviations  $M_Q$ ,  $\sigma_Q$  and  $M_v$ ,  $\sigma_v$  correspondingly, are dependent random variables.

In order to establish a statistical dependence between the placed order quantity and a period number when it is placed,  $Q$  and  $v$  are represented as a system of the two dependent normally distributed variables that have the following joint probability density function:

$$W(Qv) = \frac{1}{2\pi \sigma_Q \sigma_v \sqrt{1-r^2}} \exp\left\{-\frac{1}{2(1-r^2)} \left[ \frac{(Q-M_Q)^2}{\sigma_Q^2} - 2r \frac{(Q-M_Q)(v-M_v)}{\sigma_Q \sigma_v} + \frac{(v-M_v)^2}{\sigma_v^2} \right]\right\}$$

where

$r$  – correlation coefficient between  $Q$  и  $v$ ,  $-1 \leq r \leq 1$ .

It should be noted that for jointly distributed normal random variables concepts of independence and uncorrelation are the same. That is, if random variables are independent, they are uncorrelated and vice versa.

If the value of the random variable  $v$  is known, then the value of the random variable  $Q$  is conditional. In this case, it has a conditional probability density function:

$$W(Q/v) = \frac{W(Qv)}{W(v)} = \frac{1}{\sqrt{2\pi} \sigma_Q \sqrt{1-r^2}} * \exp \left\{ -\frac{1}{2(1-r^2)} \left[ \frac{(v-M_v)r}{\sigma_v} - \frac{Q-M_Q}{\sigma_Q} \right]^2 \right\}$$

The conditional random variable has a conditional expected value:

$$M(Q/v) = \int_{-\infty}^{\infty} QW(Q/v)dQ = M_Q + r \frac{\sigma_Q}{\sigma_v} [v - M_v]$$

and a conditional variance:

$$\sigma^2(Q/v) = \int_{-\infty}^{\infty} [Q - M(Q/v)]^2 W(Q/v)dQ = \sigma_Q^2(1-r^2)$$

Thus, the conditional variance of the random variables  $Q$  is independent of the  $v$  value. It is estimated by their own unconditional variance  $\sigma_Q^2$  and the correlation coefficient  $r$  between  $Q$  and  $v$ .

Analytical model implementation gives an unconditional variance of the placed orders  $\sigma_Q^2$ , as it is calculated for unknown period number  $v$  when the order should be placed for each order cycle.

Simulation model allows one to estimate a conditional variance of the placed orders  $\sigma^2(Q/v)$ . Based on the results obtained, it is possible to calculate the correlation coefficient  $r$  between  $Q$  and  $v$ , using this formula:

$$D(Q)_{sim} = D(Q)_{cal} * (1-r^2) \Rightarrow r = \sqrt{1 - \frac{D(Q)_{sim}}{D(Q)_{cal}}}$$

where

$D(Q)_{sim} = \sigma^2(Q/v)$  – variance of placed orders estimated by the simulation model with known  $v$  (conditional variance);

$D(Q)_{cal} = \sigma_Q^2$  – variance of placed orders estimated by the analytical model with unknown  $v$  (unconditional variance).

Table 3: Coefficient of Correlation between  $Q$  and  $v$

Nr.	$D(X)$	$D_{cal}(Q)$	$D_{sim}(Q)$	$r$
1	100	886	312	0.8
2	196	1743	603	0.8
3	324	2882	997	0.8
4	484	4305	1502	0.8
5	676	6013	2083	0.8

The calculated correlation coefficient between  $Q$  and  $v$  (see Table 3) in the inventory system that implements the  $S$ - $s$  inventory control policy when end customer mean demand and its standard deviation change proportionally, i.e. they are dependent through the *Signal To Noise* factor, is the same for all 5 experiments. It is supposed that the correlation coefficient depends only on the lead time length. A set of corresponding correlation coefficients for various lengths of the lead time could be estimated by the elaborated combined analytical/simulation approach. As soon as the dependence between the placed order quantity and a period number when it is placed is found, the described analytical model can be used for numerical evaluation of the bullwhip effect.

## CONCLUSIONS

The analytical model for the quantification of demand fluctuation magnification (the bullwhip effect) as orders move up the supply chain in case of stochastic demand is elaborated. A combined analytical/simulation approach is used to estimate the dependence between a period number when an order is placed and end customer demand quantities with a view to make the analytical solution more accurate.

The effect of different ordering policies and inventory system parameters on the above-mentioned dependence is a subject of future research.

## REFERENCES

- Banks J. and Malave C. O. "The simulation of inventory systems: An overview". *Simulation Councils, Inc.*, June, 1984, 283-290.
- Bhaskaran S. Simulation analysis of a manufacturing supply chain. *Decision Sciences*, 29 (3), 1998, 633-657.
- Disney S.M. and Towill D.R.. "A robust and stable analytical solution to the production and inventory control problem via z-transform approach". In: *Proceedings of the 12<sup>th</sup> International Working Conference on Production Economics*, Igls, Austria, February 18-22, 2002, 37-47.
- Feller W. "An introduction to probability theory and its applications". New York John Wiley&Sons, Inc., London Chapman&Hall, Limited, Vol. 1, 2nd ed., 1967.
- Kelle P. and Milne A. "The effect of (s,S) ordering policy on the supply chain". *International Journal of Production Economics*, No. 59, 1999, 113-122.
- Merkuryev Y., Petuhova J., and Buikis M. "Simulation-based statistical analysis of the bullwhip effect in supply chains". In: 18<sup>th</sup> European Simulation Multiconference "Networked Simulations and Simulated Networks" (ESM'2004). Magdeburg, Germany, June 13-16, 2004, 301-307.
- Robinson S. "Simulation. The Practice of Model Development and Use". John Wiley & Sons, West Sussex, England. 2003.
- Silver E.A. and Peterson R. "Decision systems for inventory management and production planning". 2nd ed. New York: Wiley, 1985.
- Simchi-Levi D., Kaminsky P., and Simchi-Levi E. "Designing and Managing the Supply Chain". McGraw-Hill Companies, Inc, USA, 2000.

# An integrated hardware/software platform for both Simulation and Real-Time Autonomus Guided Vehicles Navigation

Luca Baglivo  
Mariolino De Cecco  
Francesco Angrilli

CISAS, Center of Studies and Activities for Space  
Via Venezia 1, Padova, 35131 Italy.  
Phone +39-049-8276795; Fax +39-049-8276785  
E-mail: [mariolino.dececco@unipd.it](mailto:mariolino.dececco@unipd.it)

Francesco Tecchio  
Angelo Pivato  
ItalProject S.r.l.

Via L.Da Vinci, 35015 Galliera Veneta, Italy

## KEYWORDS

Simulation, AGV, potential field control.

## ABSTRACT

This paper presents an integrated hardware/software platform for the simulation of real-time Autonomus Guided Vehicles (AGV) navigation. The platform employs the same software architecture and code actually running onboard the real-time hardware to perform off-line the control and navigation simulation. By means of this approach it is possible to optimise the control parameters, simulate the logistics or the achievement of the path planned and also make the vehicle actually run with the same all-in-one software. This method allows to reduce development time necessary for debugging, optimising control algorithms and identifying system and/or control parameters; the same software is exploited also for variables flow control and for monitoring the whole state of the system.

The above platform was used for the optimisation of AGV potential field control for real-time obstacle avoidance.

## INTRODUCTION

The intense competition in manufacturing requires materials handling systems that are agile and capable of reconfiguration, in order to have manufacturing environments highly reconfigurable and responsive to accommodate product, factory and process changes. Modern AGV can exploit such capabilities as far as their autonomy is maximised; in this way the traditional disadvantages of rigid (wire guided) AGV-systems are overcome.

More flexible AGV systems employ more flexible measurement systems such as inertial-odometric, triangulating laser sensors (Kam et al. 1997) and more flexible control algorithms such as trajectory control and potential field control for unexpected obstacle avoidance and the navigation in partially-structured environments (Graf et al. 2001, Ge and Cui 2002, Tsourveloudis et al. 2001).

The augmented flexibility obviously increases the system complexity, so that it becomes difficult to optimise the entire algorithm and/or parameters on an h/w basis; or by means of standard control design tools:

non-holonomic systems are challenging from the control point of view, as they are globally controllable but not locally, and they are not feedback stabilizable by continuous time invariant feedback control laws (Brockett 1983). On the other side, simulation environments require to develop the simulation, to write it in the specific language, to validate it, to optimise the algorithm and to translate the result in the proper real-time software environment. Part of the above phases can be avoided if the simulation and the h/w environment are the same. For this reason it was developed an h/w and s/w platform that uses the same code for the simulation of the entire AGV functionalities and for the real-time control (simply enabling the h/w interfaces with the drivers and the sensors or the simulation engine).

Recent developments in simulation tools for manufacturing have improved production engineering development and the tools are being adopted more and more widely in industry. For the development of AGV systems this has not yet fully been exploited. Several studies and works on simulation concerned more with logistic and concurrent handling aspect of AGV systems than with the pure control aspect (Verbraeck et al. 2000, Saanen et al. 2000), but it is an essential part of AGV design which asks more time for development.

Traditional methods of s/w simulation involve modelling, identification, validation and use of the simulation structure for control or other specific purposes. The proposed method adds a first step where it is designed the whole software architecture that will embody both the simulation and the real-time control and a final step where the same platform is used for the RT control.

An alternative approach achieving a real-time simulation is indicated in (European Space Agency 2004): it consists in a real-time test bench interfaced with the system in order to simulate environment, kinematics, dynamics, sensors, etc. By this architecture part of the system h/w is replaced by an RT h/w simulator. It has a valuable application for space tests when it's impossible or very expensive to test system in the real environment on earth.

Simulation has been used in this multifunction software since the development phase of controlling the AGV

drivers. The software workframe has been gradually upgraded according to development phases. An advantage is that a problem that occurs on h/w control situation can be analysed in a simulation mode, and the simulations can also be used to test different solutions before they are implemented in the real system. A feature of this simulation system is the opportunity to switch the control software in simulation mode simply excluding the hardware interface (real sensors) and enabling the simulation engine which emulate system's kinematics and dynamics by means of sensor's virtual outputs.

In previous works it was analysed the calibration problems of the vehicle kinematics (Durrant-White 1994) whose parameters, coupled with the identification of the dynamics, allows to simulate the vehicle behaviour in the real environment. After validation of the kinematics- dynamical model, it was added a second layer of software for the real-time obstacle avoidance in semi-structured environments by means of potential field methods.

An important feature of the platform is the possibility to monitor also the state of the system and manage possible faults and warnings.

### H-S/W RT SIMULATION METHOD

The steps that the proposed h-s/w RT simulation method involve for the navigation control design are the following:

1. design of the software structure that embodies all the drivers and functionality for the Real-time system control;
2. mathematical modelling of the system plant;
3. system plant parameters identification;
4. system plant model validation;
5. use of the system plant model for navigation control design and related parameters optimisation [s/w mode];
6. experimental verification [h/w RT mode].

Traditional methods of s/w simulation involve the steps 2,3,4 and 5. The proposed method adds a first step where it is designed the whole software architecture that will embody both the simulation and the real-time control. A final step is added where the same platform used for the mathematical/numerical simulation development is used for the RT control.

The important advantage is that the control algorithms do not need to be implemented on another application for their simulation purpose. It is sufficient to launch the single application disabling the libraries that achieve the hardware I/O control and enabling the simulation engine libraries developed ad hoc, which calculate virtual output of sensors on the basis of the system's model implemented in the simulation engine.

The disadvantage is that a s/w platform developed for RT control purposes requires more time to develop the simulation engine than platforms developed for numerical or control purposes. However several companies, like for example National Instruments, are

adding tools that enable the simulation in parallel to the control.

### 1. Software structure design

As illustrated in Figure 1 the software (written in LabView®) has three hierarchical levels of control: the upper level is the planning one, at a lower level are the path control and task control algorithms (including the potential field method), the lowest level is assigned to the drivers control loops, the fastest ones. The last two tasks can exchange data with two possible systems: the hardware system, which is the real one and it is essentially made up of drivers and sensors, or the simulation engine, which embodies a model of the driver system. The application can run in real-time mode on the embedded AGV system as well as in the host client PC in simulation mode. By the same interface the user can choose to launch the application in simulated mode or in hardware mode: in the first case the software runs only on the client computer; if the interface selection switch is set on hardware mode the application runs in real-time on the on-board server computer and sends data to the user interface on the client computer.

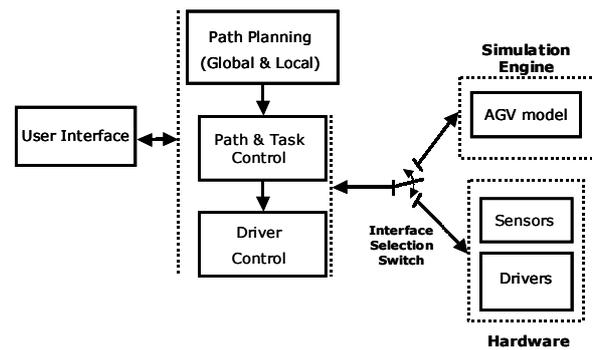


Figure 1: Block diagram description of the system.

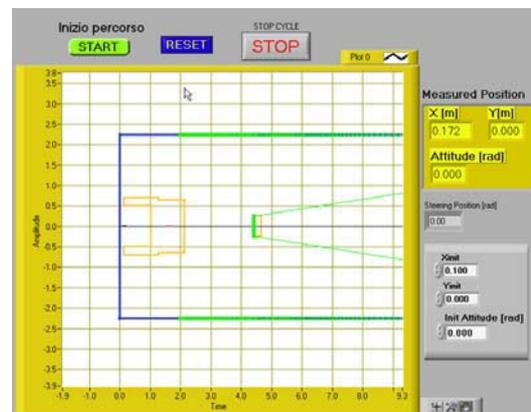


Figure 2: Control Panel example in the user interface

## 2. Mathematical modelling of the system plant

This section shows the kinematics of a three-wheeled vehicle like the ones that are often used in industrial environments for pallets handling and transport. Two encoders are used on the driver wheel: one to measure the steering angle, the second to measure the angular velocity.

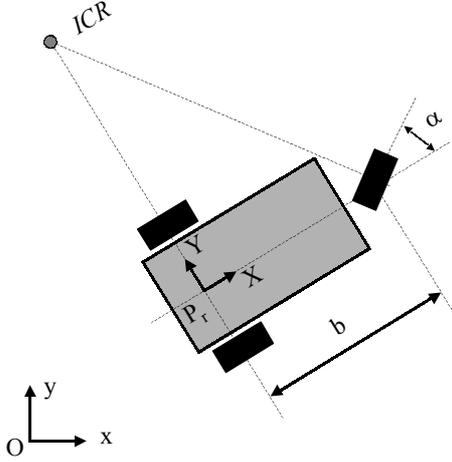


Figure 3: Kinematics scheme of a three-wheeled AGV

The Figure 3 shows the kinematics scheme of a three-wheeled AGV. The angular velocity  $\omega$  refers to the driving wheel. The attitude  $\delta$  is meant to be the angle between the absolute reference system  $xOy$  and the mobile reference system  $XP_rY$ . The position  $P_r$  is defined by the vector  $(x, y, \delta)$  which also takes the attitude of the mobile robot into account.

The kinematics behaviour is always controlled by two wheels. Therefore, one wheel bears slippage if it is not perfectly aligned with one of the other two. The discrete form of the inertial-odometric navigation equations is the following:

$$\begin{cases} x_{k+1} = x_k + \frac{2\pi}{n_{TOT}} \cdot n_k \cdot R \cdot \cos(\alpha_k + \alpha_0) \cdot \cos(\delta_k) \\ y_{k+1} = y_k + \frac{2\pi}{n_{TOT}} \cdot n_k \cdot R \cdot \cos(\alpha_k + \alpha_0) \cdot \sin(\delta_k) \\ \delta_{k+1}^E = \delta_k^E + \frac{2\pi}{n_{TOT}} \cdot n_k \cdot R \cdot \sin(\alpha_k + \alpha_0) \cdot \frac{1}{b} \\ \delta_{k+1}^G = \delta_k^G + T_C \cdot G(V_k^G) \\ \delta_{k+1} = DF(\delta_{k+1}^E, \delta_{k+1}^G) \end{cases} \quad (1)$$

Equation (1) is the characteristic which makes it possible to estimate the AGV's position and attitude. The steering motor is controlled in position and its model, used by the simulation engine, is a first order transfer function inserted in the PID control loop (Figure 4). The motor is in fact controlled in velocity by

a dedicated driver and the motor-driver behaviour in the position control chain is that of a delay.

- $\alpha_s, \alpha_m$  are respectively the angular setpoint for steering axis position and its measurement.
- $\kappa, \tau$  are the constants of motor-driver first order transfer function.

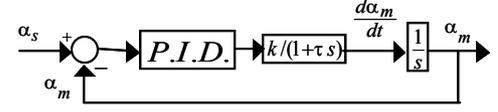


Figure 4: Control scheme of the steering motor system

The control variable for the driving motor is the angular velocity and its control scheme is similar to that of the steering motor system.

## 3. System plant parameters identification

The kinematics model calibration is discussed for example in (Durrant-White 1994, Borenstein and Feng 1995,1996): similar methods were implemented.

The motor model parameters identification was obtained by means of simple step input methods.

## 4. System plant model validation

Tests were made to validate the simulation kinematic and dynamical model of the system.

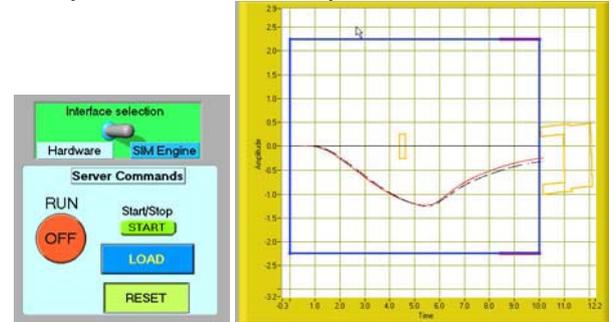


Figure 5: The Hardware/Sim Engine switching  
Figure 6: Comparison of Simulated trajectory (dotted black) vs Real one (continuous red)

In Figure 6 is presented a comparison between a simulated trajectory and a real one showing a good correspondence. The example is related to an unexpected central obstacle avoidance task.

## 5. Navigation control design and parameters optimisation [s/w mode]

The potential field method used for the task of obstacle avoidance is similar to the VFF method of Borenstein (Borenstein and Koren 1991). It is very simple to be implemented and it is efficient for local obstacle

avoidance if its parameters are well tuned. The main disadvantages of this method are three: the parameters are difficult to be tuned; it suffers of local minimum problems and oscillations in narrow passages. For the last two problems some technics were developed in order to eliminate local minimum and oscillations. The problem of tuning parameters is fully simplified by the simulation tool and the effect of changing parameters is verified simply by switching to hardware mode. Since this potential field method doesn't take into account the dimensions of the vehicle it was developed an algorithm for avoiding lateral and rear collisions with obstacles. Another problem of the potential field method is the lack of attitude control which is very negative for transpallet industrial vehicles in the load and the unload phases. By means of the h-s/w method it was possible to develop a method in order to solve also this problem while maintaining the other potential field method features untouched.

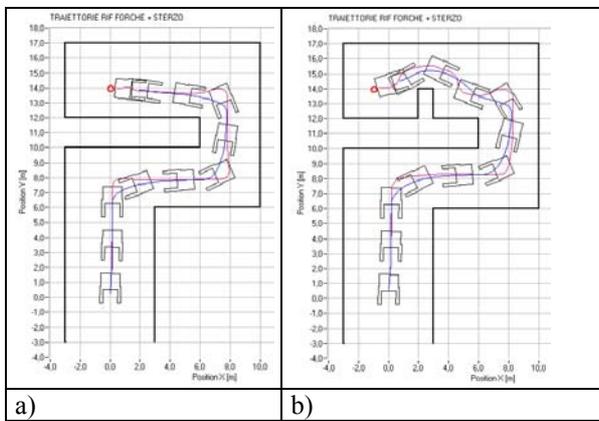
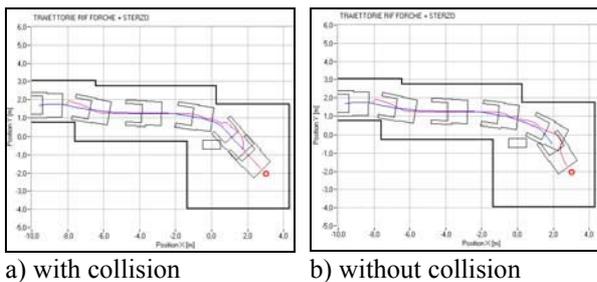


Figure 7: Simulation example: a) corridor without obstacle; b) when an obstacle appears. Trajectories of rear wheels' center and of steering axis are shown

In Figure 7 is shown the result of a simulation of the AGV performing a path in a corridor, first without obstacle and then with an unexpected obstacle appearing before approaching the target. This is a final result after parameter optimisation was utilised to solve problems of narrow passages inducing oscillations and of collisions with lateral object due to non-point shape of the vehicle.

The Figure 8 shows the first of the above problems and its solution before and after the introduction of the control corrective algorithms and their optimisation .



a) with collision b) without collision

Figure 8: Solution to the Problem of Lateral Collisions

## 6. Experimental verification [h/w RT mode]

The simulation software was used for testing a potential field method for local obstacle avoidance in partially structured environments. Before the algorithm was implemented in the actual AGV control software, it was simulated for debugging purpose, then simulation of control was used in order to optimise control parameters, finally the method was tested on the AGV. In Figure 9 we present a frame sequence (read from left to right and from top to bottom) showing an obstacle avoidance test carried out to compare simulation results with real ones. The task showed is the same simulated in Figure 6.

## CONCLUSION

In this paper, an integrated h-s/w platform has been proposed for both the simulation and the real-time implementation of Autonomous Guided Vehicles navigation. By means of this approach it is possible to employ the same software architecture and code for simulation purposes (control algorithm debugging and optimisation of parameters, path planning simulation) as well as for the real-time control on an embedded AGV system. The steps that this proposed h-s/w RT simulation method involve for the navigation control design are : design of the software structure; mathematical modelling of the system plant; system plant parameters identification; system plant model validation; use of the system plant model for navigation control design and related parameters optimisation [s/w mode]; experimental verification [h/w RT mode]. This method was successfully used for achieving the autonomous navigation of an industrial transpallet.

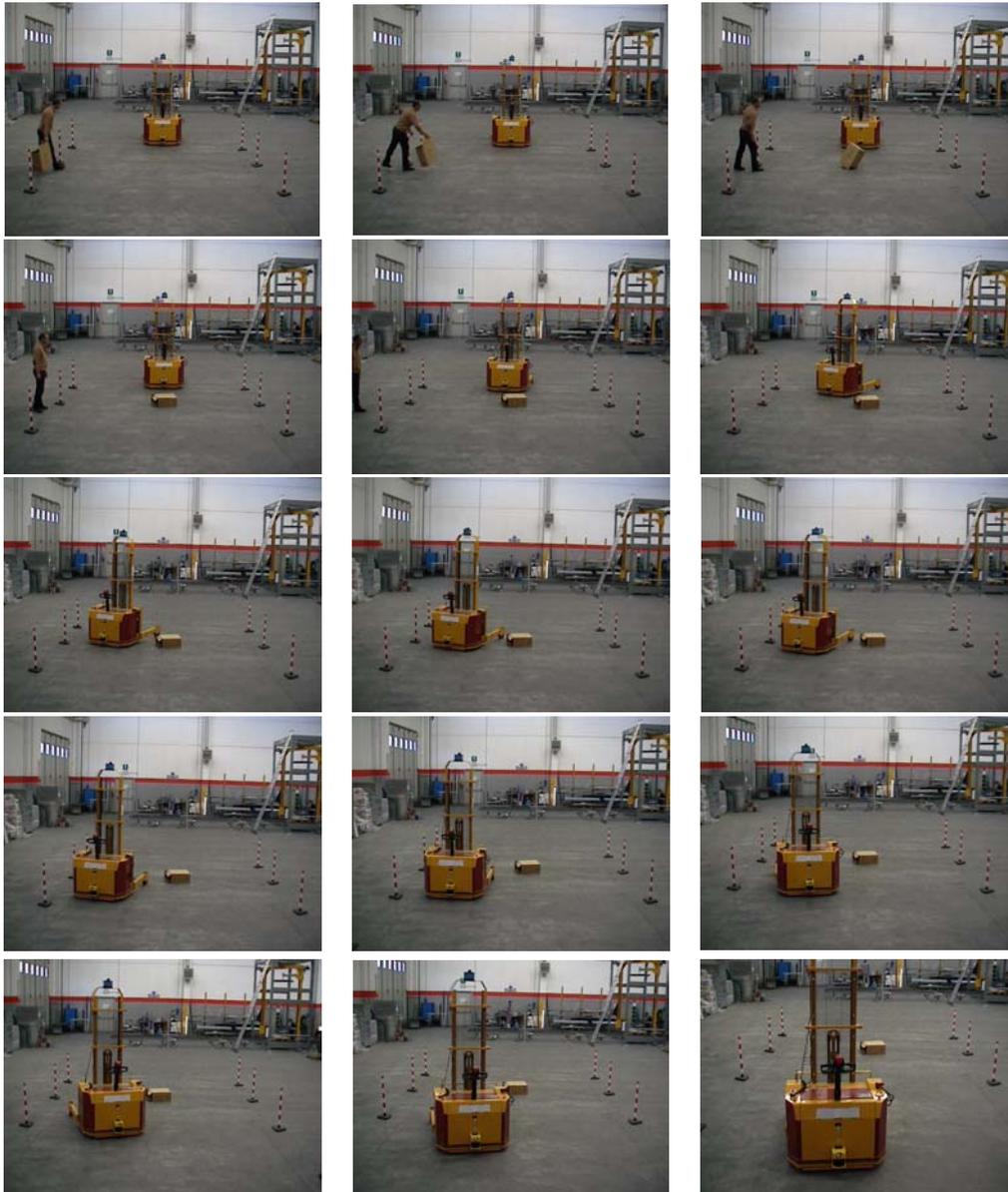


Figure 9: The result of the Real Implementation of a potential field method for Obstacle Avoidance

### Clarification

Only for academic purposes the following clarification is reported:

*Baglivo* implemented the h-s/w RT simulation and developed the obstacle avoidance algorithm, 40% of the whole work is assigned to him.

*De Cecco* conceived the integrated hardware and software method and the system modelling, 40% of the whole work is assigned to him.

*Angrilli* organised and supervised the research project both from managerial and technical point of view, 10% of the whole work is assigned to him.

*Tecchio* and *Pivato* organised and supervised the research project from the industrial point of view, 5% of the whole work is assigned to each of them.

The research was partially funded by a ministerial grant under the 46 law.

### References

- Borenstein J. and Koren Y. 1991, "Potential Field Methods and Their Inherent Limitations for Mobile Robot Navigation", In *IEEE Proceedings of the IEEE Conference on Robotics and Automation*, Sacramento, California, April 7-12, 1991, pp. 1398-1404
- Borenstein J. and Feng L. 1995, "UMBmark: A Benchmark Test for Measuring Odometry Errors in Mobile Robots", *SPIE Conference on Mobile Robots*, Philadelphia, October 22-26, 1995
- Borenstein J. and Feng L. 1996, "Measurement and Correction of Systematic Odometry Errors in Mobile Robots", *IEEE Transactions on Robotics and Automation*, Vol. 12, No 5, (Oct) 1996.

- Brockett R. 1983, "Asymptotic stability and feedback stabilization", *Differential geometric control theory*, Vol. 27, pp 181-208, Birkhauser, 1983.
- Durrant-Whyte, H. 1994. "Where am I? A tutorial on mobile vehicle localization", *Industrial Robot*, Vol. 21, No. 2, pp. 11-16.
- European Space Agency. 2004. *Multi-mission Hybryd Navigator Breadboard*, TEC-ECN-GPS-IMU 2510/2004 Issue 01
- Ge S.S. and Cui Y.J. 2002, "Dynamic Motion Planning for Mobile Robots Using Potential Field Method", In *Autonomous Robots* 13, 207-222.
- Graf B., Hostalet Wandosell J. M. and Schaeffer C. 2001. "Flexible Path Planning for Nonholonomic Mobile Robots", *EUROBOT* September 19 - 21, 2001, Lund, Sweden.
- Kam M., ZHU X. and Kalata P. 1997. "Sensor Fusion for Mobile Robot Navigation". In *Proceedings of the IEEE*, Vol. 85, No. 1 (Jan) 1997, pp. 108-119.
- Saenen Y.A., Verbraeck A. and Rijsenbrij J.C. 2000 "Simulation as a Control Device over Physical Systems". In *Proceedings ASIM2000*, (Berlin, March 2000) ASIM/SCS, 2000.
- Tsourveloudis N. C., Valavanis K. P. and Hebert T. 2001, "Autonomous Vehicle Navigation Utilizing Electrostatic Potential Fields and Fuzzy Logic", In *IEEE Transactions on Robotics and Automation*, Vol. 17, N. 4, August 2001.
- Verbraeck, A. et al. 2000: "Simulation as a Real Time Logistic Control System: AGV Control with Simple++". In *The New Simulation in Production and Logistics – Prospects, Views and Attitudes*. Mertins, K., Rabe, M. (eds.). Berlin, Germany, 2000. pp. 245-255.

## AUTHOR BIOGRAPHIES



**LUCA BAGLIVO** was born in Tricase (LE, Italy) in 1978. He studied mechanical engineering (automation) and obtained his degree in 2003. He is a Ph.D scholar toward the Research Doctorate in Space

Science, Technology and Measurement in Mechanical Engineering Dept at University of Padova. His interests include AGVs, sensor fusion, simulation and robotics. His e-mail address is: [luca.baglivo@unipd.it](mailto:luca.baglivo@unipd.it)



**MARIOLINO DE CECCO** was born in Pescara (Italy) in 1969. He studied Electronic Engineering and obtained his degree with first-class honours at the University of Ancona. He obtained

the PhD degree in 'Mechanical Measurement' at University of Padova and a. He is researcher of the CISAS. His current interests include mechanical measurements, sensor fusion, vision systems, signal processing, linear and non-linear control, adaptive control, space qualification and procedures and system simulation. He has been qualified as professor.



**FRANCESCO ANGRILLI** is full professor of Mechanical and Thermic Measurements at Padova University, Italy, he is also director of the Center of Studies and Activities for Space CISAS "G.

Colombo" of the Padova University; Director of Post-graduate courses for Research Doctorate in Space Science and Technology principal investigator in a number of researches sponsored by C.N.R., A.S.I, ESA, PMI and Private Industries; he has been member of technical team and program manager for a number of space missions. He is the Author of more than 140 papers published in italian and international journals.



**FRANCESCO TECCHIO** was born in Cittadella (PD, Italy) and he attended the University of Padova where he studied mechanical engineering and obtained his degree in 1996. Since 2001 he has been working at the Italproject srl where he is now leading the AGV (Automatic guided Vehicle) project in the field of end line packaging.



**ANGELO PIVATO** was born in Galliera V.ta (PD) .

He studied mechanical technology and applied himself to research and innovations in the packaging sector.

He is the chairman of the company Italproject srl (specialised in the design & manufacture of automatic packaging systems).

# STOCHASTIC MODELING AND OPTIMIZATION OF INDUSTRIAL STOCK

Vitalijs Jurenoks, Vladimir Janson,  
Faculty of Engineering Economics  
Riga Technical University  
Kalku iela 1, Riga, LV1658, Latvia  
vitalijs.jurenoks@rtu.lv  
vladjans@latnet.lv

## KEYWORDS

Modeling, stock management, optimization, scenario analysis

## ABSTRACT

The nonlinear stochastic model of updating of industrial stocks is considered in the paper. The purposes of work are:

1. To define an optimum strategy of updating of industrial stocks and reductions of the expenses connected to their storage;
2. To combine the modeling process with the real information describing the process of industrial stocks management in real time;
3. To take to account the nonlinearity dependences of factors of the model;
4. To construct the management process in online regime.

In the first scenario for incidental values modeling is used traditional methods, in particular, the method of the reverse transformation the Bellman's method (receipt-refusal). In the second scenario of modeling the authors have used modeling methods that allow to consider specific characteristics of changes of value  $P$  - order frequency (is measured as units of orders per unit of time), namely, irregularity of consumption intensity, different lengths of intervals between order points, inability to select an appropriate rule of distribution of value  $P$  for the whole modeling time interval.

For resolving the problem under given conditions, the method of imitation modeling was applied, which allows to develop (imitate) different options of organization of the process of stock management, taking into account the aforementioned specific characteristics of the particular scenario.

## BASIC MODEL OF STOCK MANAGEMENT

In any stock management system the level of stock changes in accordance with the respective cyclical model. The reduction of the level of stock is determined by the demand. At a specific moment, to replenish the stock after a definite period of time, termed delivery lead-time, a new order is made; the order is received and the stock is increased. After that a new stock cycle begins.

To simplify stock management modeling a, number of conditions are established:

- 1) The demand for products is constant. If the consumption coefficient is constant, then the stock level is also reducing at a constant rate.
- 2) It is assumed that delivery time is known and is a constant value, which means that the order may be made at the point corresponding to definite time parameters and stock amount (replenishment level) values, which ensure receipt of the required stock at the moment when stock level is 0;
- 3) Stock-out is not allowed;
- 4) A definite  $y$  amount of raw materials and materials is ordered during the stock cycle time.

The basic model of stock management is presented in fig. 1. All stock cycles are equal. The maximum amount of raw materials and materials available in stock correspond to order amount  $y$ .

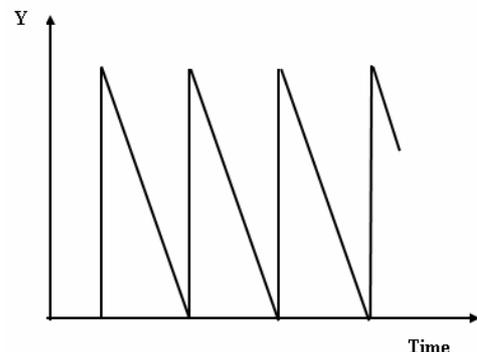


Figure 1. Scheme of the basic stock management model

A model has to be developed to describe costs over the whole stock storage period; the length of the period is not important – it may range from one day up to a year etc. In the particular case a period of one year is chosen and the following system designations are used:

- $D$  - annual demand for product;
- $C_o$  - variable costs of one order,  $n$  units / 1 order;
- $C_h$  - variable storage costs per one existing stock unit,  $n$  units per product unit per year;
- $C$  - acquisition price of one existing stock unit,  $n$  units per year;
- $y$  - order size in product units.

If the demand for the products is  $D$  units per year, but the size of each batch is  $y$  number of units, the annual demand is  $D/y$ .

The annual cost of an order ( $C_a$ ) is equal with the costs of one order multiplied with the number of orders made in a year:

$$C_a = C_0 * \frac{D}{y}. \quad (1)$$

When calculating annual stock storage costs, usually the average amount of products made from the stock during one cycle is taken into account. In the simplest case the level of stock reduces in a straight line from  $y$  till 0, thus the average level of stock is  $y/2$ .

Different companies use different  $C_h$  cost calculation methods, however, on the whole,  $C_h$  is expressed as a percentage of the cash loan amount that is frozen in the form of stocks, the costs of ensuring stock security or costs of damages, as well as costs of the stock storage system.

Annual storage costs of stocks  $C_s$  are equal with annual variable storage costs of one product unit  $C$ , multiplied with the average amount of stock per year:

$$C_s = C_h * \frac{y}{2}. \quad (2)$$

All the aforementioned allows to state that total annual costs of the stock of one product unit ( $TC$ ) is established as follows:

$$TC = C_0 * \frac{D}{y} + C_h * \frac{y}{2}. \quad (3)$$

This equation is called the equation of total costs of the basic stock management model. To establish the optimum order size ( $y_0$ ), the equation of total costs of the basic stock management model is differentiated. As a result of differentiation and numerous transformations it is possible to obtain the following formula allowing to establish the optimum order size:

$$y_0 = \pm \sqrt{\frac{2C_0D}{C_h}}. \quad (4)$$

Stock storage costs are proportionate to the size of the order, therefore graphically it is depicted as a straight line going out from the starting point of the coordinate axis. At the same time, order costs are proportionate to the value  $1/y$ .

If the order amount is not very big, then order costs are dominant – in this case orders are placed more frequently, but in small amounts. If the amount of product is comparatively big, the main component of total costs is storage costs – a small number of orders are made for big product amounts.

Annually, after equal time periods, it is necessary to place  $D/y$  orders, this means that a new order cycle always starts at point:

$$\frac{1}{D/y} = y/D. \quad (5)$$

As all order cycles are equal, then interval of repeat orders will also be equal with  $(y/D)$  years.

More simple models of stock management are characterized by constant demand, instantaneous replenishment of stock, and non-existence of stock-out.

The designations used are:

$P$  – order frequency (is measured as units of orders per unit of time);

$t_0$  – order cycle time (measured in units of time).

The level of stock changes dependent on the function (fig. 2), which showing also the above-mentioned designations.

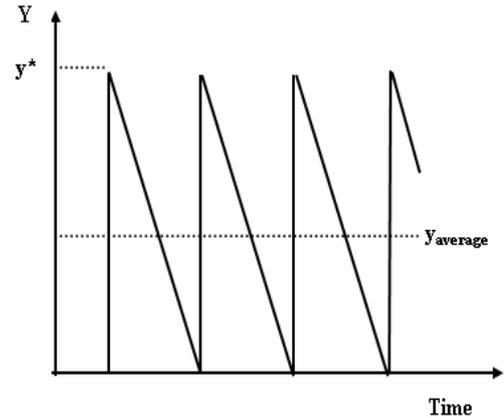


Figure 2. Stock levels

The stock units are replenished instantaneously, when the level of stock reaches 0.

After that stock is used with constant frequency  $P$ .

The length of the order cycle in this case may be calculated as follows:

$$t_0 = \frac{y}{P} \text{ units of time.} \quad (6)$$

Average stock level ( $y_{average}$ ) is established as follows:

$$y_{average} = \frac{y}{2} \text{ units.} \quad (7)$$

Total cost per unit time (TCU) may be expressed as the following function:

$$TCU = \frac{C_a + C_h * (\frac{y}{2}) * t_0}{t_0} = \frac{C_a}{\frac{y}{P}} + C_h * \frac{y}{2}. \quad (8)$$

The optimum size of order  $y$  is established by minimizing the function TCU by  $y$ . Assuming that  $y$  is a continuous value, it is possible to obtain the conditional minimum (in the form of an equation), which allows to derive an optimum value  $y$ :

$$\frac{dTCU}{dy} = -\frac{C_a P}{y^2} + \frac{C_h}{2} = 0. \quad (9)$$

This condition is sufficient, since the function TCU is a curve. The result of the equation determines the most economic size of the order  $y^*$ .

$$y^* = \sqrt{\frac{2C_a P}{C_h}}. \quad (10)$$

The optimum stock management strategy for this model is formulated as follows:

It is necessary to order  $y^* = \sqrt{\frac{2C_a P}{C_h}}$  product units

after every  $t_0^* = \frac{y^*}{P}$  units of time.

In real life, replenishment of stock cannot be effected instantaneously at the moment of placing the order, as it was thought earlier.

To make the situation more realistic, there is a condition that the positive time period for fulfilling the order is  $L$  (time deflection), which covers the time from placing the order up to physical delivery, as shown in fig.3.

In this case the order point is the instant, when the level of stock has dropped to  $L P$  units.

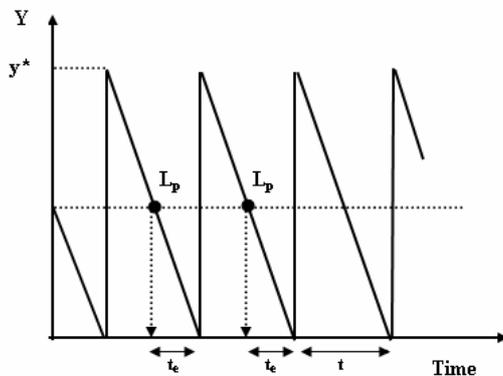


Figure 3. Stock replenishment points

Effective time  $L_e$  is established:

$$L_e = L - n t_0^*, \quad (11)$$

where  $n$  – the biggest round figure that does not exceed  $\frac{L}{t_0^*}$ . Such a solution is explained as follows:

after  $n$  cycles (each of them is  $t_0^* = \frac{y^*}{P}$  long) the

stock management situation becomes the same as the situation when the interval between placement of one order and receipt of another one were  $L_e$ .

Consequently, the order point is an instant, when the level of stock reaches  $L_e P$  product units.

Thus the stock management strategy may be formulated as follows: it is necessary to order

$y^* = \sqrt{\frac{2C_a P}{C_h}}$  product units, when the level of

stock drops to  $L_e P$  product units.

The above algorithm of optimization and calculation of order point is traditionally and most frequently used in real-time planning. In reality the function, from the point of view of order TCU, is to a great extent dependent on the value of  $P$  – frequency of orders. In order to consider this impact, the authors make an assumption about the incidental character of behavior of value  $P$ . Two modeling scenarios are considered:

Given the value  $P$ , the rule of distribution is known as  $F$ , i.e.  $P \sim F$  ( $F$  – distribution function). There is the following interdependence:

$$TCU = \Phi(P, t, t_0, \omega), \quad (12)$$

where  $w$  – incidental parameter.

The rule of distribution of value  $P$  is not known and it is necessary to model a different character of behavior of value  $P$ .

In the first scenario there is a sufficiently well developed mathematical mechanism of modeling and it is necessary only to implement and analyze the data obtained to meet the goals of optimization. When modeling incidental values, traditional methods are applied; in particular, the method of the reverse transformation the Bellman's method (receipt-refusal).

For theoretical modeling the authors have used MS Excel and the program MathCad. The analysis of the modeling results allows to choose the most optimum dynamic mode of replenishing the stock, required raw materials and materials, as well as to minimize total costs TCU, i.e. to choose an optimum strategy for stock management in a stochastic case (the first modeling scenario).

In the second scenario of modeling the authors have used modeling methods that allow to consider

specific characteristics of changes of value P, namely, irregularity of consumption intensity, different lengths of intervals between order points, inability to select an appropriate rule of distribution of value P for the whole modeling time interval.

For resolving the problem under given conditions, the method of imitation modeling was applied, which allows to develop (imitate) different options of organization of the process of stock management, taking into account the aforementioned specific characteristics of the particular scenario.

The process of development and implementation of the imitation model implies the following simplified algorithm (see fig. 4):

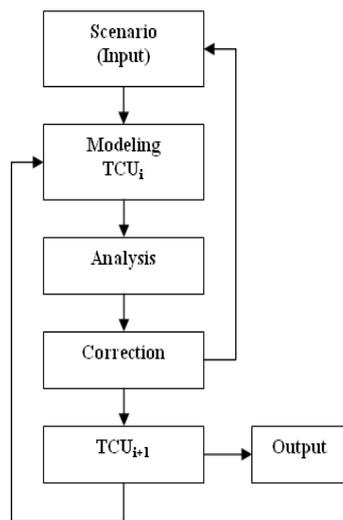


Figure 4. Algorithm of the implementation of the second scenario

Imitation modeling is one of the most widespread methods of research of economic objects and systems.

The selection of methods of modeling of the object under consideration depends on a great number of conditions (modeling components), e.g. complexity of the object or system being researched:

- the character of behavior of the object or system;
- the character of behavior and the impact of the factors on the changes of the entity or economic system being investigated;
- other similar conditions.

Where the relations between separate components forming the model are comparatively simple and can be accurately described, analytical models can be used for obtaining the required information.

However, most of the economic processes and systems are complex entities, consisting of a great number of interrelated sub-systems (which in their turn also are complex objects and require a detailed study), changing their positions in space and time. For researching such economic systems it is impossible to create an absolutely accurate effective model by applying analytical methods. Fig. 5 presents a process of creation of an imitation model.

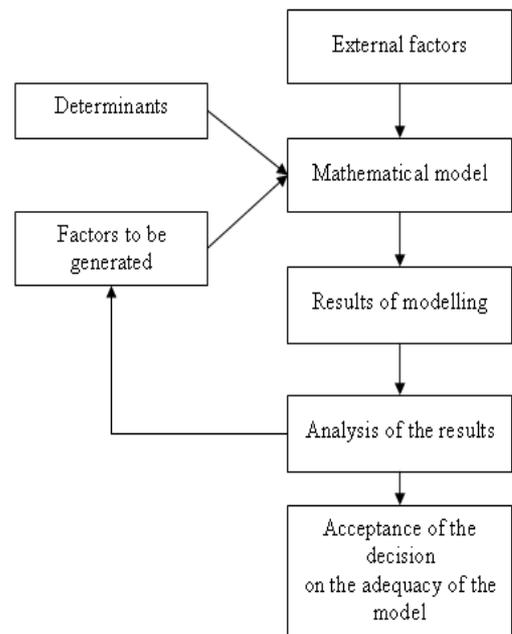


Figure 5. Process of creation of an imitation model

Where the relations between separate components forming the model are comparatively simple and can be accurately described, analytical models can be used for obtaining the required information.

However, most of the economic processes and systems are complex entities, consisting of a great number of interrelated sub-systems (which in their turn also are complex objects and require a detailed study), changing their positions in space and time. For researching such economic systems it is impossible to create an absolutely accurate effective model by applying analytical methods. In such cases it is necessary to use the methods of imitation modeling.

Imitation modeling is usually applied for researching economic processes and systems. Such factors are called incidental variables or incidental values, and their behavior is described by means of common probability distribution functions.

Imitation modeling may be used for tackling a wide range of economic problems (design and analysis of industrial systems, stock management, balancing of production capacities, allocation of investment funds, optimization of investment funds, optimization of flows of services etc.).

Imitation modeling is frequently associated with the factor of uncertainty, whose description goes outside the confines of the traditional statistical modeling, which, in its turn, complicates the imitation modeling process.

The aim of the research is to find innovative ways of using imitation modeling for investigating economic systems. As a result, it is possible to set the task of creating an efficient procedure for generating incidental parameter values constituting factors of an imitation model, to consider the asymmetric distribution of model factors, to create an adequate

model of non-linear dependence between the factors, to effectively use up-to-date information technologies, to ensure continuous control of the behavior of the specific economic system that is being researched.

The traditional scheme of imitation modeling is the formation (generation) of a mass of incidental parameter values featuring the changes of model factors (see fig. 6).

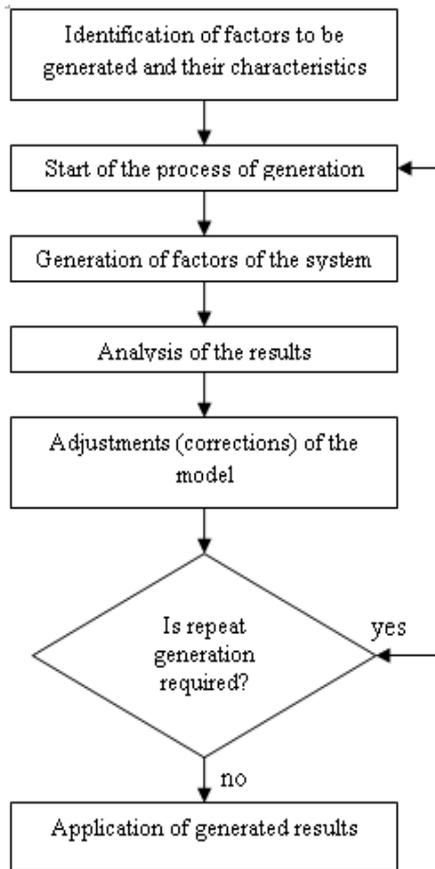


Figure 6. Algorithm of generation of incidental parameters

The algorithm of generation of incidental continuous value  $X$ , having continuous distribution function  $F$ , can be described in the following steps:

1. Let us generate, within an interval (0.1), an evenly distributed incidental parameter  $u \sim U(0.1)$ .
2. Let us calculate  $X = F^{-1}(u)$ .

The value of  $F^{-1}(u)$  will always be definite, since  $0 < u < 1$ , but the area of defining the function  $F$  is the interval  $[0,1]$ . The figure below presents the essence of the algorithm graphically; here incidental value may be assumed to be either positive or negative. This depends on the specific value of parameter  $u$ . In the figure, the value of parameter  $u_1$  produces a negative incidental value  $X_1$ , but parameter  $u_2$  yields a positive incidental value  $X_2$  (see fig. 7).

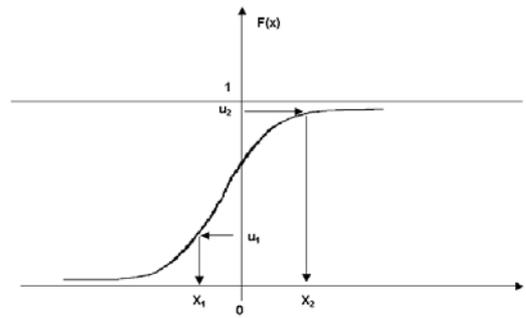


Figure 7. Scheme of reverse transformation

The method of reverse transformation may be also used if value  $X$  is discrete. In this case the distribution is as follows:

$$F(x) = P\{X \leq x\} = \sum_{x_i \leq x} p(x_i), \quad (13)$$

where  $p(x_i)$  is probability  $p(x_i) = P\{X = x_i\}$ .

It is admitted that incidental parameter  $X$  may have only such values as  $x_1, x_2, \dots$ , for which  $x_1 < x_2 < \dots$ . Thus the algorithm of developing the values of incidental parameter  $X$  will have the following consequences:

- Let us generate, within the interval (0.1), uniform distributed incidental parameter  $u \sim U(0.1)$ ;
- Let us establish the least positive round value  $I$ , for which  $u < F(x_i)$ , and assume that  $X = x_i$ .

Both options of the method of the reverse transformation for continuous and discrete values (at least formally) can be combined in one formula:

$$X = \min P\{x : F(x) \geq u\}, \quad (14)$$

which is true also for mixed distributions (i.e., containing both continuous and discrete components). In contrast to commonly used direct methods of generating incidental values (the method of the reverse transformation composition and implosion), for imitating the factors of the imitation model it is recommended to use the so-called indirect methods, namely, the acceptance-refusal method. This method may turn out to be suitable if due to certain reasons it is impossible to apply direct methods or if these methods are inefficient.

The "acceptance-refusal" principle is rather common. If the aforementioned algorithm is looked upon from a slightly different perspective, it is clear that it may be extended for generating incidental points in areas having higher dimensions – i.e. in multi-dimensional areas.

This is relevant in modeling real economic systems by applying the Monte Carlo method.

When using the MS Excel program, users usually apply standard functions for modeling incidental parameter values in a dynamic regime.

The variances of the random variables must be finite for the correlation to exist, and for fat-tailed distributions this cannot be the case (a bivariate t-distribution with 2 degrees of freedom, for example). Independence between two random variables implies that linear correlation is zero, but the converse is true only for a multivariate normal distribution. This does not hold when only the marginals are Gaussian while the joint distribution is not normal, because correlation reflects linear association and not non-linear dependency. Correlation is not invariant to strictly monotone transformations. This is because it depends not only on the joint distribution but also on the marginal distributions of the considered variables, so that changes of scales or other transformations in the marginals have an effect on correlation.

In order to overcome these problems we can resort to copula theory, since copulas capture those properties of the joint distribution which are invariant under strictly increasing transformation. A common dependence measure that can be expressed as a function of copula parameters and is scale invariant is Kendall's tau. It satisfies most of the desired properties that a dependence measure must have and it measures concordance between two random variables: concordance arises if large values of one variable are associated with large values of the other, and small ones occur with small values of the other; if this is not true the two variables are said to be discordant. It is for this reason that concordance can detect nonlinear association that correlation cannot see.

The main problems of the research are:

1. To combine the modeling process with the real information describing the process of industrial stocks management in real time.
2. To take to account the nonlinearity dependences of factors of the model
3. To construct the management process in online regime.

## CONCLUSION

Application of the imitation modeling allows:

1. To implement a continuous dynamic control of parameter TCU and, consequently, allows to optimize the process of industrial stock management;
2. To consider the inconsistent and frequently unpredictable behavior of value P, which is especially relevant for small and medium-size enterprises, which do not have guaranteed orders and are heavily dependent on the market conjuncture fluctuations.

The received theoretical results of work can be used in practical activities of the industrial enterprises, using standard programs of modeling.

## REFERENCES

Sklar, A., "Functions de repartition a n dimensions et leurs marges", Publ. Inst. Statist. Univ. Paris 8, 1959 – p. 229-231.

Sklar, A., "Random variables, joint distributions, and copulas", *Kybernetika* 9, 1973 – p.449-460.

Кельтон В., Лоу А. Имитационное моделирование. Классика CS. 3-е изд. – СПб.: Питер; Киев: Издательская группа BHV, 2004. – 847 с.: ил.

Мур, Джеффри, Уэдерфорд, Ларри., и др. Экономическое моделирование в Microsoft Excel, 6-е изд.: Пер. с англ. – М.: Издательский дом "Вильямс", 2004. – 1024 с.: ил.

Таха, Хемди, Введение в исследование операций, Кн. 2. Пер. с англ. – М.: Мир, 1985. -496с., ил.

## AUTHORS BIOGRAPHIES

**VITALIJS JURENOKS** was born in Riga, Latvia and went to Riga Technical University to the Faculty of Engineering Economics when he obtained his degree in 1976. He worked ten years for industrial company in Riga. From 1986 he is working in Riga Technical University where he obtained doctor degree in economical science in 1987. The main field of research is planning and optimization of economical processes and systems. His e-mail address is: [vitalijs.jurenoks@rtu.lv](mailto:vitalijs.jurenoks@rtu.lv).

**VLADIMIR JANSON** was born in Daugavpils, Latvia and went to Latvian University of Latvia, where he studied mathematical science and obtained his degree in 1970. He worked eight years for Latvian University Computational Centre. From 1978 he worked in Latvian Technical University where he obtained doctor degree in mathematical science in 1983. The main field of research is simulation and optimization of economical systems. His e-mail address is: [vladjans@latnet.lv](mailto:vladjans@latnet.lv).

# A VERIFICATION METHOD FOR THE SIMULATION OF SUPPLY CHAIN NETWORKS WITH UNRELIABLE LINKS

Katrien Ramaekers and Gerrit K. Janssens  
Operations Management and Logistics  
Limburgs Universitair Centrum,  
Diepenbeek, Belgium  
e-mail :  
{katrien.ramaekers,gerrit.janssens}@luc.ac.be

Kenneth Sørensen  
Faculty of Applied Economics  
University of Antwerp, Antwerp, Belgium  
e-mail: kenneth.sorensen@ua.ac.be

Rik Van Landeghem  
Industrial Management  
University of Ghent, Ghent, Belgium  
e-mail: hendrik.vanlandeghem@ugent.be

## KEYWORDS

Verification, logistics, supply chains, Petri nets

## ABSTRACT

As supply chains exceed the limits of a company their behaviour grows more complex. The design of an efficient global supply chain network is of great importance in a competitive environment. As many uncertainties occur in the flow of goods, information or payments, simulation is an appropriate tool to model the behaviour of such a network. A chain is known to be as good as its weakest link and these links have shown to be failure prone. Therefore our focus is on chains with unreliable links. This enlarges the complexity of the model. In order to obtain a correct model for the logistics reality to be studied, we propose a formal method, which is able to generate a simulation model in an automatic way. Petri nets are chosen as the formalism. The procedure how to build such a model is outlined in detail for a serial logistics system. Hints are given how to extend the idea in a context with alternatives in case of link failures.

## 1. INTRODUCTION

The supply chain of a company encompasses all entities, such as plants, distribution centres and transportation modes, that ensure a steady flow of products, from raw materials, up to the final delivery in the hands of the consumer. This means that most supply chains link multiple companies in a sequence of

supplier-customer relations (Beamon 1998, Slats et al. 1995).

The literature on designing a supply chain focuses mainly on the objectives of minimising cost or maximising profit. Most methodologies do not take into account reliability, although it has been identified as a key performance contributor (Vidal and Goetschalckx 1996). A reliable supply chain should assure on-time arrival of parts and subassemblies to the different locations, so that excessive work-in-process do not build up causing delays, inventory, costs or poor customer service. The notion of supply chain reaction time refers to the total elapsed time between receipt of the customer order and the final delivery of the complete order.

The major components of a generic supply chain are shown in Figure 1. They can be divided into: information flows (dashed line), material flows (solid line), process steps (rectangle) and storage locations (triangles). The directed network, formed by the sequence of flows in Figure 1, consists of two paths, representing the major flows through the supply chain.

A first flow (A) starts with the customer (the order), is converted by the Order Mix function (a planning step) into an assembly command that passes to the component storage. In the storage material is picked for the order at hand, and moved to the assembly line. Assembly can only start if all parts are available. Some of the parts are on-order only, which forms a second path (B): from the Order Mix, to Purchasing and the Suppliers.



neighbouring links too. If a previous operation does not provide any raw materials the production can get starved. If the storage at the wholesaler is full, the distribution company is blocked. Depending on the stochastic behaviour of each link and the sizes of the intermediate buffers the performance of the whole chain changes.

The Petri net is further extended with alternative choices. The alternatives are not choices, which are evaluated each time the link is used. They serve to resolve problems if it is likely that the link will get starved, blocked or will cause stoppages further in the chain. Examples are: if a ferry has to stay in the harbour, air carriage can be used; if a supplier is out of stock, goods can be bought from a secondary supplier. It is clear that the use of alternatives influences the chain performance, e.g. in terms of speed or cost.

### 3 MODELLING A GLOBAL SUPPLY CHAIN

The behavior of a supply chain with multiple links and intermediate storages is very complex. Logistics managers are interested in performance measures of such a system. *Availability*, or the percentage of time that the chain is supplying to the final customer, is used frequently as an important performance measure. However, analytical results exist only in very simple cases. In other cases either approximations or simulation models are used. If abstracted, the types of models studying a smooth and constant flow of goods from supplier to customer are similar to models of flow-line systems. The name of the systems is borrowed from serial production systems.

In the literature on multi-stage lines with finite intermediate buffers, De Koster (1989) distinguishes four classes of models. His third class deals with continuous flow models. Machine speeds are deterministic but machines may fail. This is the model class, which refers to our type of modeling. Goods flows are deterministic but supply chain links may fail. Some examples of these models can be found in Wijngaard (1979) and in Malathronas et al. (1983).

While obtaining analytical results for a system with many machines is considered to be an impossible task, also the approximative models or simulation models are to be questioned. Approximation models are of two types: aggregation models or decomposition models. The system with two machines and one buffer in a continuous flow-line, has been studied by Malathronas et al. (1983). An approximative model for three machines is formulated by Van Oudheusden and Janssens (1994) in which exponential uptimes and downtimes are assumed for aggregated machines. Both

approximation models and simulation models make use of system states. For lines with multiple machines and buffers, the number of states tends to become very large. A formal technique to generate all possible states of the system in an automatic way could be of great help. In the next sections we investigate the opportunities and limitations of a Petri net as a formal modeling tool for flow line systems within the context of a supply chain.

### 4 A PETRI NET MODEL FOR A SERIAL SUPPLY CHAIN

A Petri net in its graphical or mathematical form has no physical meaning. However, it is able to model systems in which some events occur concurrently, but in which there are constraints on the concurrence, precedence or frequency of these occurrences. This idea can be used to model supply chains. A place may represent a condition (e.g. a link is available or not), or it can represent a link status (e.g. the number of tokens represents the number of vehicles within the distribution system).

Van der Aalst (1994) has suggested the use of Petri nets for logistics systems. Goods and capacities can be represented by tokens; buffers, storage space and media by places, and operations by transitions. In continuous flow these definitions cannot be held. Individual items do not exist. Tokens and places require other meanings.

As far as we know, only one paper deals with Petri nets used to model automatic transfer lines. Al-Jaar and Desrochers (1990) present two stochastic Petri nets to evaluate the performance of transfer lines. The assumptions are comparable to ours, except for the fact that they work in a discrete production environment instead of a continuous one. Modelling a continuous chain including breakdowns is the main subject of this contribution.

The serial supply chain consists of  $n$  links, separated by  $n-1$  storages ( $n \geq 2$ ). The storages in the system can be in one of three states: empty, full or in an intermediary position (labelled 'half'). Links can be either up, down, starved or blocked. A link  $M$  is starved if one of the upstream links is down and all storages between this link and link  $M$  are empty. A link  $M$  is blocked if one of the downstream links is down and all storages between this link and link  $M$  are full. When a link is operational and neither starved nor blocked, it supplies goods from the upstream storage to the downstream storage in a continuous way at a constant rate. The situation is shown in Figure 2.



Figure 2: An  $n$ -link  $(n-1)$ -storage supply chain

### States Associated With a Link or a Storage

We propose a Petri net in which a *place* represents the state of a link or of a storage. For each storage, three places are added to the net, indicating that the storage is respectively full, empty or at an intermediate level.

For each link, except the first and the last, four places are added, indicating that the link is respectively up, down, blocked or starved. The first link does not have a place for the state 'starved' and the last one does not have a place for the state 'blocked'. Graphically, the set of places  $P$  is represented in Figure 3:

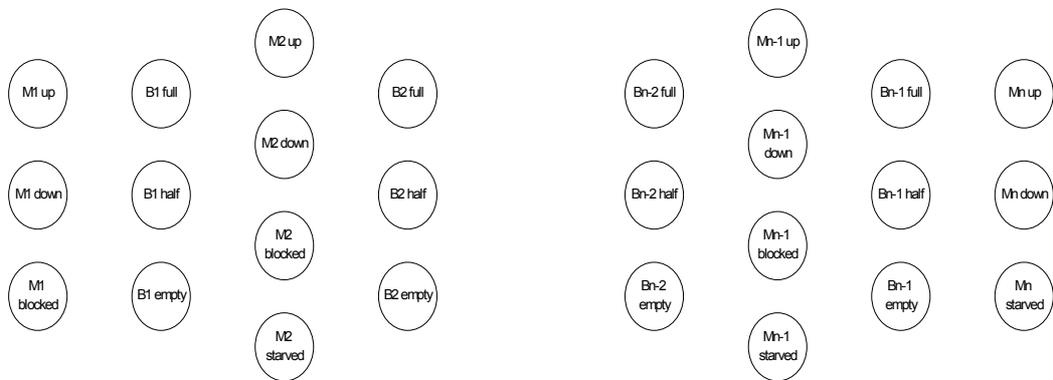


Figure 3: Places of a general Petri net representing the supply chain links and storages

### Transitions of the Petri Net Model

Failure or start-up of a link may influence the state of each other link or storage in the chain. Likewise, when a storage becomes empty, this may have an effect on each downstream link or storage. When a storage becomes full, this may have an effect on each upstream link or storage. It is impossible to define a general building block of places and transitions, which can be repeated to describe the behavior of a chain with multiple links and storages.

For example, assume all links are operational, but all storages are empty. If link 1 goes down, all other links become starved. This means that the effect of a single event can spread throughout the entire chain of links and storages and that it is impossible to define a building block of e.g. one link and one storage, which can be repeated to form the Petri net of a serial system with  $n$ -links and  $(n-1)$ -storages.

For this reason, we do not concentrate on the transitions of the Petri net, but on the rules to generate the transitions. The transitions model the events, which

occur while the system evolves. Four types of events need to be modelled:

1. a link goes down,
2. a link starts up (is repaired),
3. a storage becomes full,
4. a storage becomes empty.

All other changes of state occur as a result of one of these four events. E.g. a link becoming blocked is always the result of a storage becoming full. A storage changing state from full or empty to an intermediary position is always the result of a link failure or a link start-up or of another storage becoming full or empty.

### Rule schemata

The rules for generating the Petri net transitions are represented by schemata. A change of state for a specific link or storage is indicated by a square containing the names of two places, separated by the symbol  $V$ . Such a square is called a *block*. The upper place is added to the input places of the transition, the lower place is added to its output places. In Figure 4, the breakdown of link  $i$  is depicted.

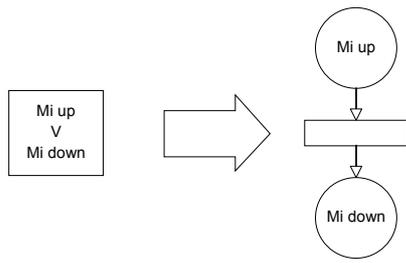


Figure 4: Failure of link  $i$

The Petri net transition equivalent to this event is also given. The arrow means 'gives rise to the transition'. As

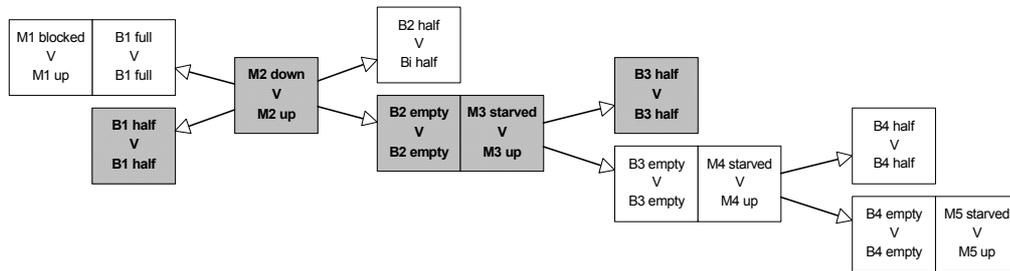


Figure 5: Reduced and expanded schema of link 2 starting up in a 5-link system with one possible path indicated

As an example, one path through the schema is indicated by the shaded blocks. The transition, which belongs to this path is shown in Figure 6.

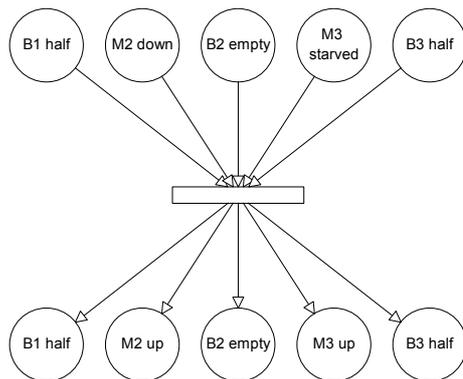


Figure 6: Transition equivalent to the shaded path in figure

There are eight possible paths through the schema in figure 5. Therefore, eight transitions are dressed up for the event *link 2 starts up*.

text, the block in Figure 4 can be denoted as  $[Mi\ up > Mi\ down]$ .

A block may contain two identical places. This means that there is no change of state for this storage or link, but the places are needed in the transition. Blocks that are adjacent but not connected by arrows can be considered to be one block. All places have to be added to the transition. The details of the rule schemata and the instructions when to use them fall beyond the scope of this paper. A complete description can be found in Sørensen and Janssens (2003).

For example, assume we are dressing up the transitions for the event *link 2 starts up* in a 5-link 4-storage system. The schema is given in Figure 5.

The Petri net transitions are timeless, i.e. they fire instantaneously, no matter how far-reaching their implications are. Also, transitions cannot fire simultaneously. This excludes some situations from occurring. E.g. If a link is down, the upstream storage cannot be empty, because this situation can only occur if the storage became empty at exactly the same time as the upstream link went down.

## 5 TRANSFORMING THE PETRI NET MODEL INTO A SIMULATION MODEL

### Token Attributes

To transform the Petri net model into a simulation model, a time aspect needs to be added to the Petri net. This can be reached by attaching one or more attributes to each token in the Petri net. A difference is made between static and dynamic attributes. Static attributes do not alter as the simulation progresses and can be seen as the parameters of the simulation model. Dynamic attributes change during the simulation.

A token belonging to a link invariant (a token that indicates the state of a link) has one dynamic attribute, which has a different meaning depending on the place the token is in (state the link is in). Their meanings are indicated in Table 1.

If the amount of goods in a storage is expressed in a certain volume unit  $V$  and the progress of time in a time unit  $T$ , then the speed with which links add or extract goods from the storage is expressed in "volume per time" units ( $V/T$ ). In a balanced chain, we can, without loss of generality, assume that this speed is one for all links.

Link state	Meaning of time property
Up	Time left until the link fails
Down	Time left until the link starts up
Starved / Blocked	Uptime remaining (time left until the link would fail if it were operating)

Table 1: Meaning of an attribute belonging to a link invariant for different link states

Tokens in storage invariants have two attributes. A first one indicates the maximal storage capacity and is of the static type. A second one indicates the current storage content and is of the dynamic type.

### Execution of the Simulation

The simulation is now executed according to the logic shown in Figure 7. Steps 1 and 5 are typical in the dynamics of a Petri net model. Step 3 is typical in any discrete event simulation logic. Steps 0, 2 and 4 require some further explanation.

In the explanation following notations are used:

NE	Next-Event time
$T_k$	Value of simulation clock after the $k$ -th transition
$t_{i,k}$	Value of the time property of link $i$ after the $k$ -th transition
$c_{i,k}$	Value of the storage content property of storage $i$ after the $k$ -th transition
$v_i$	Value of the storage capacity property of storage $i$ .

#### Step 0: Initialise The Simulation

Initialising the simulation entails the choice of starting values for the variables in the simulation program. A difference can be made between decision variables (which do not change as the system evolves) and other variables.

Decision variables concern:

- the number of links,
- the capacity of each storage,
- the types of distributions used for failure times and start-up times and their parameters.

Other variables are:

- the state of each machine,
- the content of each buffer.

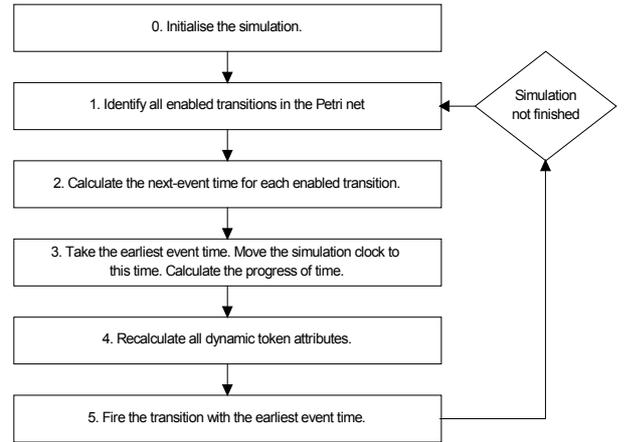


Figure 7: Simulation logic

The initial values of the latter variables should not have an impact on the results of the simulation, providing the simulation runs are long enough.

#### Step 2: Calculate The Next-event Time For Each Enabled Transition

Depending on the action performed by a transition, the next-event time for this transition is calculated as follows:

##### Link starts up

The next-event time is calculated by adding the time property of the token (the time remaining until this link goes down) to the current simulation time.

$$NE = T_k + t_{i,k}$$

##### Link fails

The next-event time is calculated by adding the time property of the token (the time remaining until this link is repaired) to the current simulation time.

$$NE = T_k + t_{i,k}$$

##### Storage becomes empty

The next-event time is calculated by adding the value of the content property to the current simulation time. This can be done because of the assumption that the speed with which links add or extract goods from the buffers equals one.

$$NE = T_k + c_{i,k}$$

##### Storage becomes full

The next event time is calculated by adding the difference between the values of the storage capacity and the storage content properties to the current simulation time.

$$NE = T_k + (v_i - c_{i,k})$$

#### Step 4: Recalculate all token properties

Because the system of links and storages has evolved for a certain period of time, some of the token properties have changed. Storages have become more full or empty, and links will have gotten closer to their failure or start-up event.

The amount of time, which has elapsed since the last transition is indicated by:  $\Delta T_k (\Delta T_k = T_{k+1} - T_k)$ .

#### Token properties of links, which have gone up or down

If the transition with the earliest next-event time is that of a link going up, the time property of its token is reset and assigned a new random drawing from its Mean Time To Failure (MTTF) distribution. If the transition is that of a link going down, the time property of its token is assigned a new random drawing from its Mean Time To Repair (MTTR) distribution.

#### Token properties of other links

If the link is up or down, the value of its time property is incremented with the progress in time.

$$t_{i,k+1} = t_{i,k} + \Delta T_k$$

If the link is starved or blocked, the value of its time property remains unchanged.

#### Token properties of buffers

For storage  $i$ , if link  $i$  is up and link  $i+1$  is not (it is down or blocked), then the storage is filling up and the content of the storage is increased with the increase in time.

$$c_{i,k+1} = c_{i,k} + \Delta T_k$$

If link  $i+1$  is up and link  $i$  is not (it is down or starved), then the storage is depleted and the content of the storage is decreased with the increase in time.

$$c_{i,k+1} = c_{i,k} - \Delta T_k$$

If links  $i$  and  $i+1$  are both up or both not up, then the value of the content property for storage  $i$  remains unchanged.

## 6 CONCLUSIONS AND EXTENSIONS FOR MORE COMPLEX SUPPLY CHAINS

Supply chain networks seldom show such a simple structure as the one explained in the previous sections. The reader should be aware that a lot of research has to be done before a general and suitable framework for verifying simulation models in this field is worked out completely. We show briefly how the option of alternatives can be embedded into the same framework.

Alternatives have different meanings in a logistics reality. They can refer to a different supplier or shipper, or to an alternative mode of transport. Each link in the chain (called primary link) can have any number of alternative links. They come into operation when a link

fails. When the alternative links are up but not operating (they are waiting for the primary or any other link to fail), they are said to be in 'standby mode'. In a real system both the alternatives and the intermediate storages can be combined, as shown in Figure 8.

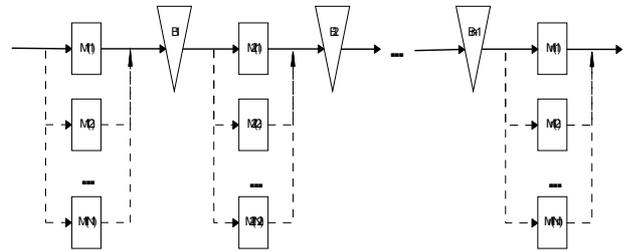


Figure 8: A supply chain network with alternative options and with intermediate storages

Also this type of network can be modeled by means of a Petri net, and its automatic generation is similar. In the case of alternative options, priorities have to be included.

Alternatives have a priority. This means that

- if a link fails, the link with the highest priority of all stand-by alternatives comes into operation.
- if a link with a higher priority than the link currently in operation is repaired, it will start up and interrupt the one currently in operation. The latter is put again in stand-by mode.

Many configurations of supply chain networks can be thought of, but simulation is only of efficient support to the designer if he is sure that the model can be verified. Automatic generation of the simulation model through a formalism, like Petri nets, is certainly of great importance in this field of practice.

## REFERENCES

- Al-Jaar R.Y. and A.A. Desrochers. 1990. "Performance evaluation of automated manufacturing systems using generalized stochastic Petri nets." *IEEE Transactions on Robotics and Automation*, vol. 6, no. 6, pp. 628-632.
- Beamon B. 1998. "Supply chain design and analysis: models and methods." *International Journal of Production Economics*, vol. 55, pp. 281-294.
- De Koster M.B.M. 1989. "Capacity Oriented Analysis and Design of Production Systems." *Lecture Notes in Economics and Mathematical Systems* 323, Springer-Verlag, 245 pp.
- Goldratt E.M. and R.E. Fox. 1986. *The Race*. North River Press, Croton-on-Hudson, N.Y.
- Malathronas J. P., Perkins J.D. and R.L. Smith. 1983.

“The availability of a system of two unreliable machines connected by an intermediate storage tank.” *AIIE Transactions*, vol.15, no. 3, pp. 195-201.

- Nicholson A., Schmöcker J.-D. and M.G.H. Bell. 2003. “Assessing transport reliability: malevolence and user knowledge.” in *The Network Reliability of Transport*, Bell M.G.H. and Iida Y. eds., Pergamon, pp. 1-22.
- Slats P.A., Bohla B., Evers J.M. and G. Dijkhuizen. 1995. “Logistic chain modelling.” *European Journal of Operational Research*, vol. 87, pp. 1-20.
- Sörensen K. and G.K. Janssens. 2003. “A Petri net model for a continuous flow transfer line with unreliable machines.” *European Journal of Operational Research*, vol. 151, pp. 253-264.
- Van der Aalst W.M.P. 1994. “Putting high-level Petri nets to work in industry.” *Computers in Industry*, vol. 25 no. 1, pp. 45-54.
- Van Landeghem R. 1998. “Experiments with MISTRAL, a supply chain simulator.” in *Proceedings of the International Workshop on Modelling and Simulation within a Maritime Environment*. September 1998, Riga, Latvia, pp. 71-75.
- Van Landeghem R. and M. Debuf. 1997. “Supply chain characterisation through Monte Carlo simulation.” In *Proceedings of the Production Economics Conference*. July 1997, Göteborg, Sweden
- Van Oudheusden D.L. and G.K. Janssens. 1994. “Availability of three-machine two-buffer systems.” *JORBEL (Belgian Journal of Operations Research, Statistics and Computer Science)*, vol. 34, no. 2, pp. 17-37.
- Vidal C.J. and M. Goetschalckx. 1996. “The role and limitations of quantitative techniques in the strategic design of global logistics systems.” School of Industrial and Systems Engineering Research Report, Georgia Institute of Technology, Atlanta, GA.
- Wijngaard J. 1979. “The effect of inter-stage buffer storage on the output of two unreliable production units in series, with different production rates.” *AIIE Transactions*, vol. 11, no. 1, pp. 42-47.

## AUTHOR BIOGRAPHIES

**Gerrit K. Janssens** (°1956) received degrees of M.Sc. in Engineering with Economy from the University of Antwerp (RUCA), Belgium, M.Sc. in Computer Science from the University of Ghent (RUG), Belgium, and Ph.D. from the Free University of Brussels (VUB),

Belgium. After some years of work at General Motors Continental, Antwerp, he joined the University of Antwerp until the year 2000. Currently he is Professor of Operations Management and Logistics at the Limburg University Centre (LUC) within the Faculty of Business Administration. He also holds the CPIM certificate of the American Production and Inventory Control Society (APICS). During the last fifteen years he has been several times visiting faculty in universities in Thailand, Vietnam, Cambodia and Zimbabwe. His main research interests include the development and application of operations research models in production and distribution logistics.

**Katrien Ramaekers** (°1979) holds a degree in commercial engineering from the Limburg University Centre (LUC). She started a Ph.D. on the integration of simulation and optimisation as a tool for complex logistics decision making in 2002.

**Kenneth Sörensen** holds a degree of Commercial Engineer in Information Systems from the University of Antwerp, RUCA (1997). He obtained his PhD in Applied Economics from the University of Antwerp in June 2003. His main research interests include optimization using metaheuristics, supply chain design and the design of production systems. He has published several papers in international journals on these topics. He is also founder and coordinator of the EURO working group on metaheuristics EU/ME.

**Rik Van Landeghem** holds a degree of Civil Engineer in Electro-Mechanics (RUG) and a MSc in Industrial Management (RUG). He obtained his PhD at the Ghent University in 1984. For several years he worked as a manufacturing consultant in the field of industrial informatics. He joined the BMT group in 1989, where he successively became EDP manager, Plant manager of the Sheet Metal division and finally MIS Director. Since 1993 he is a full time professor at the Ghent University, as well as Head of the Department of Industrial Management (Faculty of Applied Sciences). He is also director of the CIM-CIL Technology Transfer Center and member of the board of PICS Belgium. He is an expert in logistics systems and their application in industry. Other research interests include Concurrent Engineering. He is editor of a casebook on Computer Integrated Manufacturing, as well as co-author of a book on ERP Implementations.

# A RECONFIGURABLE COMPUTING ENVIRONMENT FOR URBAN TRAFFIC SYSTEMS

Mohamed Khalil and Evtim Peytchev  
School of Computing and Informatics  
Nottingham Trent University,  
Burton Street, Nottingham, NG1 4BU, UK  
E-mail: mohamed.khalil@ntu.ac.uk, evtim.peytchev@ntu.ac.uk

## KEYWORDS

Telematics, distributed shared memory, non-locking, partial replication, reconfigurable

## ABSTRACT

This paper presents a reconfigurable computing environment for building hierarchical traffic telematics distributed systems based on non-locking distributed shared memory algorithm. The algorithm aims mainly at minimising the total amount of time for data retrieval in network of workstations, considering the point of view of distributed traffic modules. The framework presented in this paper adopts a non-locking model to achieve the required performance. The presented framework develops further the successful features of DIME (developed and designed at SOCI, NTU) and at the same time avoids its shortcomings. The experimental results show that the new framework outperforms the old design of the system.

## INTRODUCTION

The potential for enhancement of the performance of current urban traffic control (UTC) systems, through supervisory control layer making use of in-vehicle route guidance, has created the need for a flexible computing environment in which various new applications can be integrated with existing traffic control systems without adversely affecting their performance. Building parallel and distributed applications such as UTC system on NOW requires a middleware of software that can efficiently manage exchanging messages and data between different applications running on different machines. Traditionally there are two paradigms in building such middleware in distributed systems – the message passing (MP) paradigm and the distributed shared memory (DSM) paradigm. The former was predominantly used for building distributed systems, in which the programmers have to be conscious of where the data is and how the processes communicate with each other, making it hard to construct distributed systems using this paradigm.

Further research in finding more convenient alternatives led to the introduction of the distributed

shared memory paradigm, which provides an illusion of shared memory based on non-physical shared memory in a network of workstations where shared data reside in different address spaces. Unlike the MP paradigm, DSM algorithm facilitates accessing the shared memory and exchanging data via normal read and write operations, making life easier for programmers of parallel and distributed applications. Also, exchanging complex data between processes in different locations is supported by DSM Algorithms (Protic et al. 1996).

Research efforts in DSM paradigm have resulted in presenting number of different algorithms applying the concept of DSM abstraction (Argile et al. 1996), (Amza et al. 1996), (Li 1988). One important conclusion of the research in DSM algorithm is that, building distributed systems on network of workstations with DSM algorithm is a viable alternative to the traditional message-passing paradigm.

The Distributed Memory Environment (DIME) (Argile et al. 1996) is one of those systems that use a DSM algorithm. It provides an interface between software modules that execute on networked workstations. DIME system was designed specifically to support vast range of transport telematics applications and to offer a convenient interface to the applications programmer. As it was built as a user-level software DSM system, DIME provides an easy to use communication interface that simply and reliably delivers data and messages to all nodes in the system. This interface was built on top of Transmission Control Protocol/Internet Protocol. The implementation of the communication interface supports a variety of platforms such as DOS, Windows, UNIX and EPOC (operating system for palm-top computers PSION) (Peytchev and Bargiela 1998).

The new framework of DIME presented in this paper; called DIME-II; uses an original approach for measuring the performance of DSM algorithms, unlike many different approaches that have been introduced and taken into account in past and recent research to measure the performance of DSM systems. For example, Munin (Carter et al. 1995) adopts multiple relaxed consistency protocols in order to achieve good

performance through reducing the number of messages exchanged in the network. On the other hand, TreadMarks (Amza et al. 1996) adopts the same means, but to speed up the distributed system as a whole.

In this paper, we use an original approach for quantifying the performance of the produced DSM framework. We assume that the most important factor of measuring the performance of the system is the reduction of data retrieval time from the viewpoint of user applications. The motive behind this assumption is that the user application can have more time for performing its native tasks, which time is very often wasted in network communication. This paper presents a new software DSM framework based on a reconfigurable non-locking and partially-replicated model. It is presented as an improvement and extension to the current implementation of DIME system. The framework reflects features specific to urban traffic distributed system. This paper refers to current implementation of DIME as DIME-I, and the new improved framework as DIME-II, and when talking about common features will be using DIME.

## RELATED WORKS

TreadMarks (Amza et al. 1996) is a software DSM system where messages and data traffic is reduced by relaxing consistency semantics of the shared memory. Also, it is a user-level implementation of DSM relies on UNIX standard libraries in order to accomplish remote process communication, and memory management, therefore no need to make modifications on the operating system kernel. In (Lu et al. 1995) experimental results have shown that the separation of synchronization and data transfer and the request-response nature of data communication are responsible for lower performance comparing with PVM message-passing model. In DIME-II, there is no need for synchronization mechanism for a user application to have an exclusive access to the shared memory, since each user application is associated with an intermediate memory where all its operations are performed.

BDSM (Auld 2001) is a broadcast-based, fully replicated software distributed shared memory system. Similar to our framework, each user process has an associated DSM subsystem that manages the shared memory, however, each user process has a complete copy of the shared memory where it processes all reads and writes locally. Also, unlike our system, all writes to memory modify the local copy and arrange to broadcast the updated values to all the other processes. Another major difference with the presented framework is that BDSM allows one user process to be executed on a workstation.

## AN OVERVIEW OF THE TRAFFIC CONTROL TRAFFIC INFORMATION FRAMEWORK IN NOTTINGHAM CITY

The ongoing collaboration between Nottingham Trent University (NTU) and the Nottingham Traffic Control Center (NTCC) has resulted in the prototype development of a distributed computers control environment. This environment offers an efficient access to real-time traffic data collected by the SCOOT Urban Traffic Control (UTC) computers and allows various telematics applications to communicate with each other while maintaining a unified logical view of the data. The communication harness is based on a standard TCP/IP protocol and client/server architecture, which makes it easily adaptable to various UTC systems, yet independent of the physical computing hardware and network type (Peytchev and Coggan, 1999).

The configuration of the traffic control system in Nottingham city contains an installation of the SCOOT system at NTCC, an installation of bus-tracking proprietary GPS system run by ACIS-UK, a distributed shared memory system developed by NTU (called DIME system) connected to both SCOOT and ACIS control centre in Cambridge and ATTAIN system - a supervisory layer of traffic control information service for serving user route finding requests (developed by NTU). On request the system is capable of accommodating variety of other software modules i.e. a predictive macro simulator PADSIM (developed by NTU), micro simulation program HUTSIM (developed at the Helsinki University of Technology) etc. The work presented in this paper is part of the ongoing collaboration between NTU and NTCC in the UTC research in Nottingham city, UK.

## DIME-I'S CONFIGURATION

DIME system (as first introduced in (Argile et al. 1996)) is a flexible computing environment that provides a communication harness for the execution of software modules of urban traffic control systems, and allows all these modules to be effortlessly integrated. As in TreadMarks (Amza et al. 1996), the system provides user-level runtime library routines to perform read/write operation on the shared memory. But, unlike IVY (Li 1988), these routines require no modifications on the underlying operating system as they can be used and run on different kinds of environments and platforms.

DIME system is designed to work with two functionally different read operations and two functionally different write operations, supporting two different types of complex data structures (Peytchev 1999). The attributes of a shared element of any data type is user-defined, thus, this system can also be categorised as object-based DSM system. Moreover,

DIME is designed to work in a heterogeneous computing environment, however, because it does not facilitate data transactions between different platforms all applications that use DIME system should perform data conversion upon exchanging data between different platforms. Furthermore, new modules can be added to the system at runtime with no disruption to the running modules.

DIME-I system adopts a sequential consistency model and a centralized architecture of the memory manager. In a centralized algorithm all accesses to a shared memory are directed to a central server that controls the whole shared memory. Thereby, DIME-I does not require any hardware routines to detect accesses to pages of shared memory; therefore it can be ported to both UNIX and DOS (or any other similar operating platform) based systems with no modifications to the kernel of the hosting operating system. In DIME-I, each user application has an additional component linked to it, which provides the communication interface via DIME-I API with the shared memory system (Figure 1). The DIME-I library transfers requests for reading/writing data from/to the shared memory over the network to the memory manager task, where they are being processed and replies are sent back. There are two components of DIME: a) The shared memory manager (i.e. DIME-II-server), which owns the shared area, and b) the communication DIME libraries (i.e. DIME-II-client), which are linked to user applications and the DIME-server in order to interface to the network (Peytchev 1999).

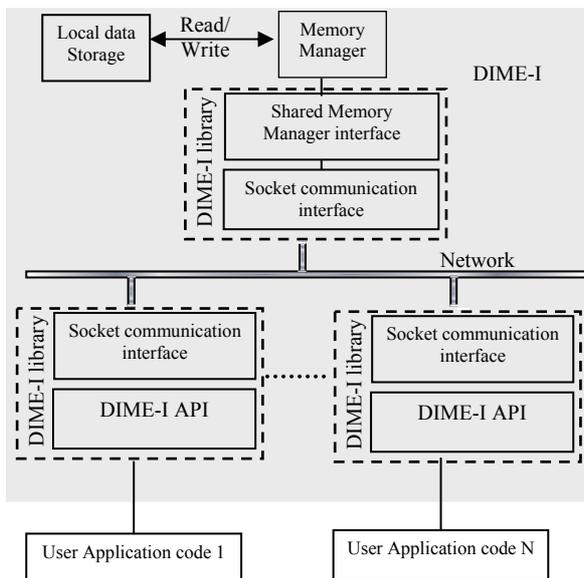


Figure1: DIME-I Configuration

In supporting the traffic simulation, monitoring and control applications, DIME system maintains two types of data structures. One type is dynamic data (data buffers), which are used for data stream collection by the real-time traffic control system. This type of data is

characterised by its high volume- in excess of 120 Mbytes per day. The other type is static data (data areas), and is updated in a much longer period of time. The purpose of this data structure, in general, is to make output results from the traffic modules available for reading by the other functional modules in the system. In the shared data areas every new write operation will destroy the old copy of data considering it obsolete (Peytchev 1999).

### THE LIMITATIONS OF DIME-I

DIME-I system adopts Sequential consistency (SC) model (Lamport 1979), which is the most commonly assumed model by the programmers due to its intuitively expected throughput (Weiwu et al. 1998), (Hill 1998). However, SC definition is very strict and usually has a problem of poor performance. As shown in (Lipton and Sandberg 1988) experiments proved that any attempt, in a sequentially consistent distributed shared memory system, to change the protocol in order to improve reading operations performance makes writing operations performance worse, and vice versa. Equally important, many performance enhancement techniques, such as prefetching, multiple-issue, and write buffer, are not allowed in a sequentially consistent machine (Tanenbaum and Steen 2002).

In brief, with the definition of the sequential consistency model, a DSM system will not provide a high-quality performance, and moreover, any attempt to improve the performance using improvement techniques may not be successful or may not even be possible to implement. Therefore, subsequent definitions have been introduced to provide a consistent memory in DSM systems with relaxed or weak constraints in accordance with the nature of the required computing environment, while maintaining the usability and the programmability of the consistency model.

The main memory is controlled by only one central manager that is responsible for servicing read/write requests from/into the shared memory from all the applications in the system. This central mechanism used by the memory manager is liable to bottleneck problem, because all requests are directed to only one server. Moreover, such mechanism is unreliable, particularly if the server crashes, in which case the distributed modules of the system will lose communication with each other, leading to the failure of the entire system.

### NON-LOCKING DSM MODEL

To overcome the previously detailed limitations of the current implementation of DIME-I system and in order to improve the performance of the system, DIME-I architecture has to be modified to support implementing non-locking approach, and the data

replication algorithm. This framework aims at improving the performance of DIME-I software DSM system mainly by minimizing the time of data retrieval from the viewpoint of user application. With DSM algorithms, distributed applications often waste valuable time while retrieving data from the shared memory, this time is spent by the middleware system during exchanging data and messages between different parts of the system to fetch the requested data. However, with the algorithm presented in this paper, an application retrieves the required data from a memory associated with that particular application. This intermediate memory contains copies of the data required by that application (not a whole replica of the main memory) and accessed only by that application.

The burden of making the memory consistent all around the system is entirely left to the middleware system, and applications can retrieve the data from their associated memories with no competition with each other. Thus, user applications will always find the required data without significant delay, bearing in mind that the data is retrieved directly from the intermediate memory. Providing an application with the requested data in a relatively short period of time is considered the one single most important factor of measuring the performance of the system. The motive behind this assumption is that user applications can have more time for performing their native tasks, which time is very often wasted in network communication.

Having its architecture in figure 1, DIME-I can be viewed as a system of three components, which are: DIME-I-server (the shared memory manager), DIME-I-client (DIME-I APIs), and user applications, wherein all shared data are resident in one machine controlled by one central memory manager. In DIME-I, DIME-I-client acts as an inactive process that is activated by a user application upon performing read/write operations on the central shared memory. Typically, a user application performs reads/writes on the shared memory via calling DIME-I-client routines that contact DIME-I-server to execute the prescribed operation, and then returning to its previous inactive state along with the result of the operation.

As illustrated in figure 2, the process of DIME-I-client is placed in a separate layer along with an intermediate memory holding copies of part of the whole shared memory in terms of data areas and buffers; this process is called DIME-II-client. This part of the shared memory contains the data required by the user application that uses the services of DIME-II system through its DIME-II-client. In the architecture, the shared memory consists of two types of data structures: data buffers that represent dynamic data, and data areas representing static data. DIME-II-client tasks are: establishing a persistent connection with DIME-II-server, saving updates in the intermediate memory as they are received from DIME-II-server,

sending updates to the server as they are received from its user application, sending data to the user application when requested, DIME-II-client is responsible for requests retransmission when no acknowledgement has been received as a response from the server in a predefined timeout.

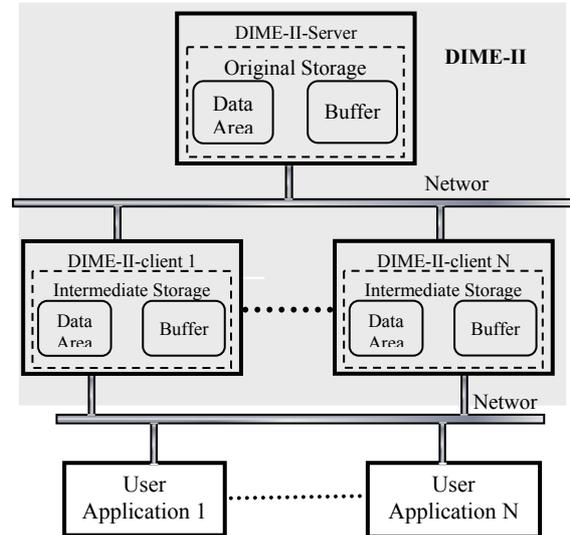


Figure 2: Non-Locking and Partially-replicated Model –DIME-II Structural Design

Thereby, a user application can perform any read or write operation on the intermediate memory rather than dealing directly with the main server, saving valuable time that can be used to perform its native tasks. In this framework, user applications; which are traffic control system modules; perform any operation on the local shared memory leaving the time delay burden of contacting the server to DIME-II-client for making the intermediate memory up to date, and reflecting updates on the original memory. On the other hand, DIME-II-server takes control over the original shared memory system, and has the role of monitoring any modification on the central memory in order to keep all intermediate memories throughout the system consistent with the original one. Unlike DIME-I, the shared memory is split over the network, and controlled by at least one memory manager. This support of the existence of several data replicas needs a special care to be taken into account to avoid data inconsistency in such architecture; a consistency model was introduced in (Khalil and Peytchev 2003) to avoid such inconsistency.

One important feature inherited from the DIME-I, is that each read/write command is considered as a single atomic operation, therefore no resources locking takes place and the system relies on the natural sequencing of the commands occurring in TCP/IP-networked environment. In regard with this feature, and because there is no need for any explicit synchronization for an application to have an exclusive access on the shared memory, the improved architecture of DIME (i.e. DIME-II) is categorized as a non-locking algorithm.

## EVALUATING THE NEW FRAMEWORK

To examine the new framework, experiments were launched to quantify the performance of DIME-II system in comparison with the old system DIME-I. Both systems employ a non-locking algorithm as no synchronization mechanisms needed for having exclusive access to the shared memory, but unlike DIME-I, DIME-II system uses an approach that allows applications to have replicas of the shared memory in their proximity. This approach is expected to speed up data retrieval rates as an application can retrieve data from the local replica rather than fetching the data from the main server over the network. Therefore, in these experiments the data retrieval time from the viewpoint of the applications is considered as a major measurement factor for the performance.

For the experiments two kinds of codes was used to compare the performance of the two systems. The first code, writer, continuously performs write operations on the shared memory, whereas, the second code, reader, keeps reading from it. As it can be perceived, such codes make a significant overhead on the system, and make the network busy all the time, as the read/write operations require a persistent communication and continuous data exchanging between DIME-II-server and its DIME-II-clients. It can also provide a good idea about how the system performs with highly demanding applications. The codes were executed on three Windows2000-based machines, while the server was based on a UNIX-based machine. The experiments were conducted on the university LAN at different times of the day (i.e. included peak and off-peak time). The evaluation was based on different workloads; each workload executes only one writer and different number of readers. Workload 1 means one reader and one writer, and workload 2 means two readers and one writer, and so on.

## EXPERIMENTAL RESULTS

The individual execution of the two systems with the codes on different workloads yields the results shown in table 1. The results have shown that with DIME-II system, an application can retrieve more data from the shared memory comparing to the old implementation of DIME-I. With DIME-II, data retrieval rate ranges between 5.3 Kilo Byte/ second and 26.3 Kilo Byte/ second, whereas, with DIME-I, it has a range of 1.7-2.7 KB/second. The performance chart in figure 3 shows that the performance of the system deteriorates sharply as the number of running applications increase. This undesirable scalability is explained in the next section.

Table 1: The Performance in DIME-I and DIME-II Measured as Data Retrieval Rates (Kilobytes/Second)

Workload	DIME-I	DIME-II
Workload1	2.7	26.3
Workload2	2.5	23.4
Workload3	2.4	21.6
Workload4	2.3	16.3
Workload5	2.3	12.9
Workload6	2.0	12.6
Workload7	1.7	5.3

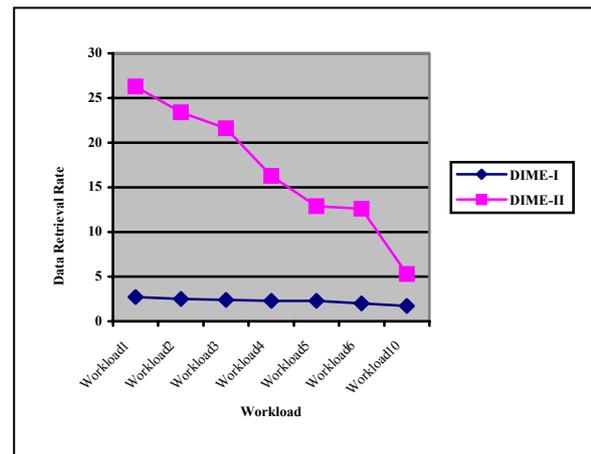


Figure 3: The Performances of DIME-II in Comparison with DIME-I

## SUMMARIZING AND EVALUATING THE RESULTS USING CONFIDENCE INTERVAL METHOD

To evaluate the performance of the DIME-II system with DIME-I, a method of confidence interval is used (Weisong et al. 1998). The basic idea behind the use of this method is that a definite statement can not be made about the characteristics of all DSM systems, but a probabilistic statement about the range in which the characteristics of most systems would drop can be made. The variety of applications determines that it is not possible for one system to be better than others in all cases (Adve et al. 1996). In this evaluation, the steps of the paired confidence interval method are literally followed.

- The differences between the two systems are:  
23.6 20.8 19.2 13.9 10.7 10.6 3.6
- Sample mean = 14.6
- Sample variance = 49.2
- Sample standard deviation = 7.0
- The 0.95-quantile of a t-variate with 6 degrees of freedom is 1.9.
- 90% Confidence interval for difference =  
 $14.6 \pm t \sqrt{(49.2/7)} = 14.6 \pm 2.65 \times 1.94 = (9.459, 19.74)$

According to the paired confidence level method we can draw the conclusion that with 90% confidence level DIME-II is better than DIME-I considering that the performance is measured as data retrieval rates from the viewpoint of system modules. However, the results show that, unlike DIME-II, there is no great variation in the rates using DIME-I system as the number of applications increases. Therefore, DIME-I scales better than DIME-II.

### THE TIME (IN MILLISECONDS) DIME SERVER SPENT IN LISTENING TO MESSAGES FROM THE NETWORK

In order to find out the reason of the high variation in data retrieval rates in DIME-II system, and based on the fact that the time of computation is far less than the time of communication, the time that DIME-II-server spent on the socket listening and waiting for messages from the network is roughly measured (table 2).

Table 2: The Time DIME-II-Server Spent Listening To Messages from the Network

Workload	Time	Time%
Workload1	238747	79.6%
Workload2	269753	89.9%
Workload3	282269	94.1%
Workload4	217572	72.5%
Workload5	243709	81.2%
Workload6	274755	91.6%
Workload7	234964	78.3%

This experiment shows that DIME-server approximately spends 72%-94% from the execution time listening to the network - depending on the state of the network, which means it has few time to perform the main task of executing commands on the shared memory. Put in other words, as the number of applications in the system increases the rates of data retrieval decrease due to the fact that DIME server always has the same few amount of time to share out between the increased number of applications. This kind of implicit waiting is one of the factors that degrade the performance of parallel-processing systems (Lai et al. 1997). Therefore, as the size of the system becomes large the likelihood of performance deterioration in DIME-II system becomes unavoidable. This limitation of the framework is overcome by adopting an optimization algorithm as shown next.

### FRAMEWORK RE-CONFIGURABILITY

Khalil M. et al (Khalil and Peytchev 2005) has introduced a round-trip time-based algorithm that scales up the performance of the system by adjusting to the demands of the applications and the current state of the network. This algorithm aims at optimizing the performance of DSM systems by reconfiguring the system connectivity (i.e. the communication paths

connecting different parts of the system) to adjust to the current state of the network while preserving the main structure of the system. This reconfiguration can be achieved by initiating another level of service that can supply the running applications with the service rather than taking it directly from the main server.

In the presence of intermediate level of control, the communication paths of applications can be diverted from the main server and connected to nearby intermediate servers. The location of an intermediate server can be identified according to round-trip times between different parts of the system allowing the system to adjust to the current state of the system. Having several levels of control also allows more applications to be added to the system while maintaining good performance (i.e. good scalability). As shown in (Khalil and Peytchev 2005), the major requirement to implementing the algorithm is that the structural design of the DSM system has to be flexible, in order to reconfigure the communication paths without adversely deteriorating its performance. In other words, the DSM system must be flexible enough to allow the algorithm to redirect connection routes between the applications and the main server.

Having the architecture of DIME-II, it can be noticed that application communicates with the system via its DSM client (DIME-II client), and the DSM clients take the burden of sending/receiving updates to/from the main server. Therefore, it can be concluded that the structure of the DIME-II system is flexible in the sense that DSM clients can take the service from other servers residing in between DSM clients and the main server, as long as these intermediate servers have a tunnel for communication with the main server. Such flexibility is needed to implement the heuristic algorithm. On the basis of these findings, DIME-II system has been examined with and without the adaptive algorithm with 10 applications shown in figure 4.

The architectures appeared in the mentioned figure represent the different structures for DIME-II DSM system produced by the algorithm at run-time. It can be seen from the produced architectures that the algorithm has boosted the performance by up to 34% with 10 applications (figure 4b). Therefore, due to reconfigurability of the structural design of the DIME-II system, the heuristic algorithm can be implemented and improve the performance of the system, and then overcome the limitation of the original structure of DIME-II system.

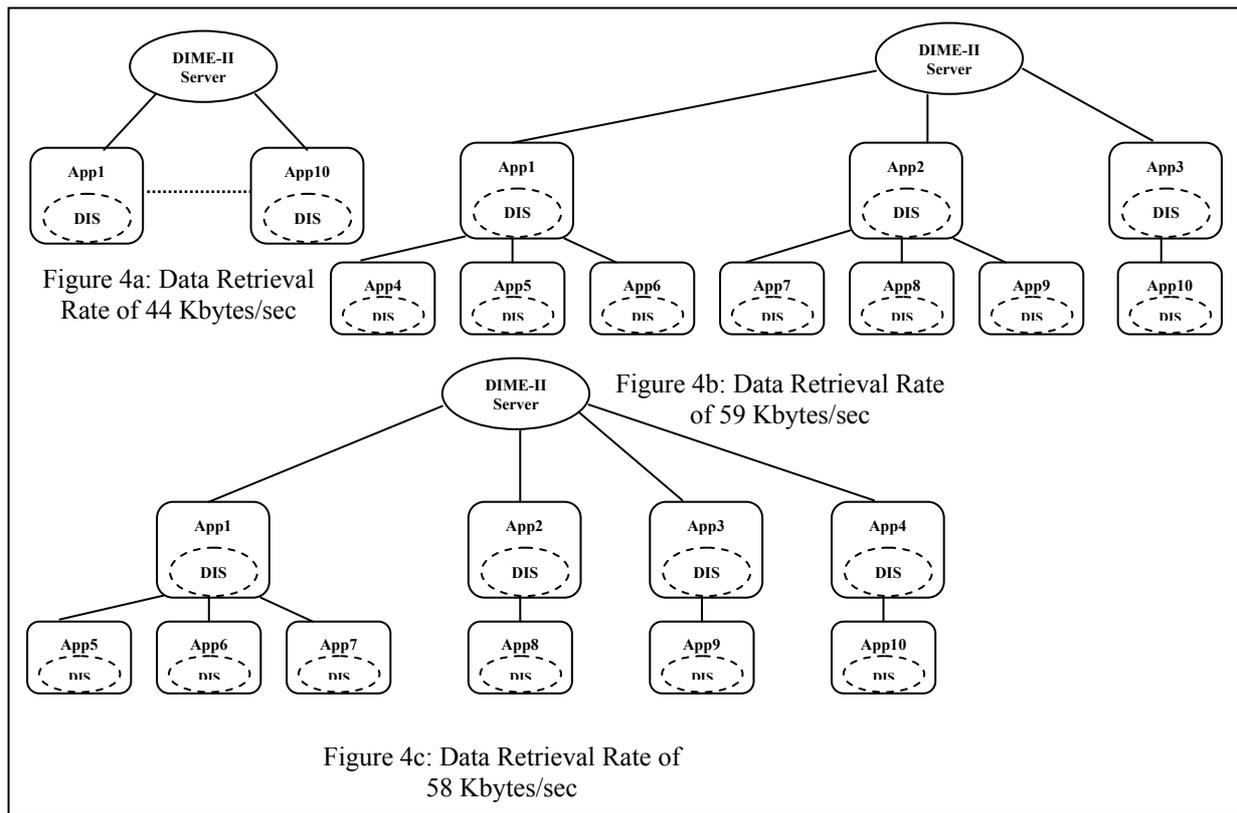


Figure 4: Different Structures of DIME-II Produced by the Heuristic Algorithm (10 applications)

## CONCLUSIONS

This paper presents a description of a software DSM prototype for traffic telematics control systems. This prototype is sought to improve the performance of the whole system in many aspects: saving network resources by the exchanging the minimum number of data and messages, reducing data retrieval from user application viewpoint.

The framework of DIME-II with the non-locking approach and locality of references has show a minimization in the time of data retrieval from the viewpoint of the applications in the system. Therefore, in DIME-II, an application can have more time to execute its native tasks. On the other hand, DIME-I scales better than DIME-II in the sense that its performance in terms of data retrieval time does not change significantly. DIME-II has less scalability due to the fact that DIME server has to maintain replicas of the data required by the applications and with increased number of applications the server must maintain increased number of replicas that lead to the great variation in the data retrieval rates in the system. This limitation has been overcome by implementing a heuristic algorithm.

## REFERENCES

- Adve S., A. L. Cox, S. Dwarkadas, R. Rajamony and W. Zwaenepoel, 1996 "A comparison of entry consistency and lazy release consistency implementations, *Proceedings of the 2nd High Performance Computer Architecture Conference*, 26-37, (Feb).
- Amza C., A. Cox, S. Dwarkadas, P. Keleher, H. Lu., R. Rajamony, W. Yu and W. Zwaenepoel, 1996 "TreadMarks: shared memory computing on networks of workstations", *IEEE Computer*, Vol. 29, No.2, 18-28, (Feb).
- Argile A., E. Peytchev, A. Bargiela and I. Kosonen, 1996 "DIME: A Shared Memory Environment for Distributed Simulation, Monitoring and Control of Urban Traffic, *Proceedings of European Simulation Symposium ESS'96*, Genoa, ISBN 1-56555-099-4, Vol.1, 152-156, (Oct).
- Auld P., 2001 "Broadcast Distributed Shared Memory", *PhD Thesis*, Department of Computer Science, The college of William and Mary, USA.
- Bargiela A., Peytchev E., "Intelligent Transportation Systems – towards integrated framework for traffic/transport telematics applications", *IEEE Semi-annual Vehicular Technology Conference* 6-11 October, 2001, New Jersey, USA.
- Carter J.B., J.K. Bennett and W. Zwaenepoel, 1995 "Techniques for Reducing Consistency-Related Information in Distributed Shared Memory Systems", *ACM Transactions on Computer Systems*, Vol. 13, No. 3, 205-243, (Aug).
- Hill M., 1998 "Multiprocessors Should Support Simple Memory Consistency Models", *IEEE Computer*

*Journal*, Vol.31, No.8, Publisher: IEEE Computer Society, USA, 28-34, (Aug).

- Khalil M., E Peytchev 2003 "Traffic Telematics Computing Framework Based on Non-Locking and Housekeeping Distributed Shared Memory Algorithm", *Sixth United Kingdom Simulation Society Conference (UKSIM2003)*, Emmanuel college, Cambridge, UK. 201-206, (Apr).
- Khalil M., E. Peytchev 2005 "Heuristic Algorithm for Performance Optimization in Distributed Shared Memory Systems", *the Eighth International Conference on Computer Modelling and Simulation (8<sup>th</sup> ICCMS)*, St. John's College, Oxford, UK, 110-115, (Apr).
- Lai A. C., C. K. Shieh and Y. T. Kok, 1997 "Load Balancing in Distributed Shared Memory Systems", *The 1997 IEEE International Performance, Computing, and Communications Conference*, 152-158, (Feb).
- Lampert L., 1979, "How to make a Multiprocessor Computer that correctly executes Multiprocessor Programs", *IEEE Transactions on Computer*, Vol. C-29, No. 9, 690-691, (Sep).
- Li K., IVY, 1988 "A Shared Memory Virtual Memory System for Parallel Computing", *Proceedings of the 1988 International Conference on Parallel Processing*, II-94 - II-101.
- Lipton R. and J. Sandberg, 1988 "PRAM: a Scalable Shared Memory", *Technical Report CS-TR*, Princeton University, 180-188, (Sep).
- Lu H., S. Dwarkadas, A.L. Cox and W. Zwaenepoel, 1995, "Message Passing versus Distributed Shared Memory on Networks of Workstations", *Proceedings of the 1995 ACM/IEEE Supercomputing Conference (IEEE Cat. No. 95CB35990)*. ACM Part, New York, NY, USA, Vol.1, 865-906.
- Peytchev E. and A. Bargiela, 1998 "Traffic Telematics Software Environment, Simulation Technology: Science and Art *10th European Simulation Symposium ESS'98*, San Diego, CA, USA, 378-82.
- Peytchev E., 1999 "Integrative Framework for Discrete Systems Simulation and Monitoring", *Ph.D. thesis, Department of Computing, Nottingham Trent University*, Nottingham, England, (Feb).
- Peytchev E. and, J. Coggan, 1999, "See before you go", *Traffic Technology International* - December 1999/Jan 2000 Issue. ISSN 1356-9252, 69-72 (Dec).
- Protic J., M. Tomasevic and V. Milutinovic, 1996 "Distributed Shared Memory: Concepts and Systems", IEEE, USA.
- Tanenbaum A. and M. Steen, 2002 "Distributed Systems: Principles and Paradigm", New Jersey, Prentice Hall.
- Weisong S., H. Weiwu, T. Zhimin, 1998 "Using confidence Interval to Summarize the Evaluating Results of DSM Systems", *Technical Report, Center of High Performance Computing, Institute of Computing Technology, Chinese Academy of Sciences*, (Sep).
- Weiwu H., S. Weisong, T. Zhimin, 1998 "A Framework of Memory Consistency Models", *Journal of Computer Science & Technology, Publisher: Science Press, China*, Vol.13, No.2, 110-124, (Mar).

## AUTHOR BIOGRAPHIES



**DR. MOHAMED KHALIL** is a Research Fellow at the School of Computing and Informatics, Nottingham Trent University. Dr. Khalil was awarded his PhD July 2004 for a research titled "Integrative Monitoring and Control Framework Based on Software Distributed Shared Memory Non-Locking Model" under the supervision of Dr. Evtim Peytchev and Prof. Andrzej Bargiela. He is currently working in the Traffimatics project aiming at providing traffic information to the public as well as the traffic control for decision-making. His research interests are: Distributed shared memory algorithms: prototyping and optimization, Traffic Telematics Systems, and Web Services & Technologies.



**DR. EVTIM PEYTCHEV** is a Senior Lecturer at the School of Computing and Mathematics, the Nottingham Trent University and has been a member of Intelligent Simulation and Modelling group for 11 years. Most of the recent research work in the group, dealing with the traffic control telematics, has been carried out by Dr. Peytchev under the supervision and leadership of the head of the RTTS group Prof. Andrzej Bargiela. As a result of the research work Dr. E. Peytchev has successfully presented his Ph.D. work entitled "Integrative Framework for Discrete Systems Simulation and Monitoring". He worked as a researcher for the successful conclusion of an EPSRC project "Integrative framework for the predictive evaluation of traffic control strategies" (GR/K16593) and most of his publications reflect the work under this project. Dr. Peytchev's interests span: traffics simulation modelling, traffic Telematics, mathematical modelling of the uncertainties in traffic, distributed computing environments, shared memory design, Telematics technology application in the urban traffic control. He is involved in International collaboration with the Transportation Systems Laboratory at the Helsinki University of Technology (Dr. I. Kossonen) and he is principal investigator for Nottingham Trent University in the DTI funded 'Traffimatics' project.

# **Vision and Visualization**



# *gSysC*: A GRAPHICAL FRONT END FOR SYSTEMC

Christian J. Eibl, Carsten Albrecht, and Rainer Hagenau  
Institute of Computer Engineering  
University of Lübeck  
Ratzeburger Allee 160, D-23538 Lübeck, Germany  
Email: {eibl, albrecht, hagenau}@iti.uni-luebeck.de

## KEYWORDS

SystemC, GUI, Simulation Controller

## ABSTRACT

Nowadays, hardware development bases on high-level methods with appropriate tool support. SystemC, a C++ class library, provides a high-level interface to model and simulate hardware designs on different levels. Unfortunately, there is no graphical interface included for demonstration, debugging, or educational purposes. *gSysC* presented here forms a GUI to SystemC. It allows the programmer to watch the interaction of the simulated design parts and provides more runtime control features such as single-step simulation or breakpoints.

## 1 INTRODUCTION

Simulation is a state-of-the-art process to test, verify, and profile newly designed hardware models. Generally, it provides exhaustive views and in-depth analysis of crucial, unapproachable system parts. Nowadays, an increasing number of hardware design tools using hardware description languages are available. But there is a demand for higher-level methods of modelling and simulating, especially for hybrid hardware/software systems. SystemC [SystemC 2002] brought out

by a pool of companies is a C++ class library that allows simulation of systems compounded of modules modelled on varying abstraction levels. It backs the top-down design methodology so that each module can be iteratively redesigned. Unfortunately, SystemC models can only be analysed by trace and log files. Visualisations of simulated modules and their interaction is not provided but they can be of course introduced utilising any library for graphical output. A fixed graphical user interface (GUI) would help to benefit even more of SystemC simulation models. Especially in the areas of presentation, demonstration, and education, visual support would be helpful.

In the following, similar systems are introduced. Their strengths and weaknesses are shortly discussed and, in contrast to them, the benefit of *gSysC* is explained. Moreover, the main details are shown presenting first the concept, further on, its implementation and way of application and a case study utilising a simple model of a CPU with cache connected to a RAM module via a bus.

## 2 RELATED WORK

Currently, there are both commercial and non-commercial approaches to a GUI for SystemC. Commercial approaches generally integrate SystemC into their hardware design environments. It is used for fast functional analy-

sis and verification. For example, Prosilog Magillem [Prosilog 2005] allows the user to automatically generate SystemC as well as Verilog and VHDL code for the graphical hardware design. Moreover, simulators such as the Incisive Unified Simulator of Cadence support SystemC as well. They are able to simulate mixed-language designs so that test benches can be easily defined using SystemC. But the simulation engine is a proprietary one.

A GUI based on the open SystemC library is described in [Charest et al. 2001, Reid et al. 2001b]. It is a self-made, Qt-based front end. Qt is a platform independent GUI library for C++ [Trolltech 2002]. The focus is laid on the graphical run-time observation of signals. The interconnection of GUI and SystemC is implemented by adapting the SystemC library so that the GUI is notified of signal-value alterations by the simulation engine. The GUI provides a simulation controller for configuration, initialisation, and run-time control. In contrast to this system, the GUI presented in [Große et al. 2003] shows the full system architecture using the interactive visualisation tool SpiceVision<sup>TM</sup>. All information on system model and run-time signal values is extracted by a modification of the SystemC library.

So, both visualisation systems have a firm bond to the SystemC library because of the necessary modifications. Since 1999, several updates and revisions of SystemC were released so that both GUIs require high effort to keep them up to date.

### 3 CONCEPT OF *gSysC*

In contrast to these tightly-coupled approaches, *gSysC* provides a loosely-coupled GUI for SystemC. It is based on Qt as well and can be fully removed by a compiler flag without changing the code.

The main goals of *gSysC* are to be independent of different releases, the programmer's permission, and control of what parts are shown, and an

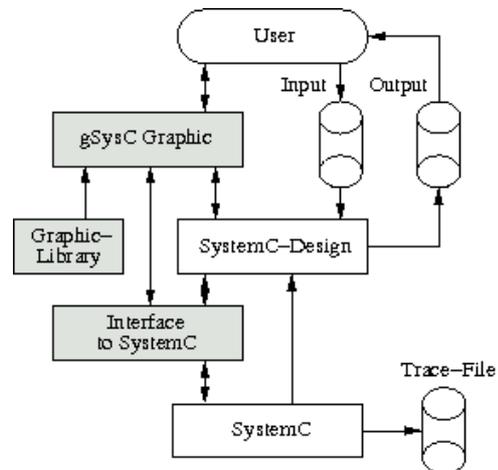


Figure 1: Programmer's View of SystemC and *gSysC*.

option of removing the visualisation. Additionally, because of the high portability of SystemC, the GUI should be supported by most platforms supporting SystemC.

Figure 1 shows the interfaces between user, SystemC model, SystemC simulation kernel represented by white boxes and the *gSysC* extensions in grey boxes. The user interface of SystemC is limited to reading a configuration at the simulation start and writing textual information or signal traces to the console or hard disk. The user cannot interact with the running simulation. The SystemC design is the software model of the simulated hardware and SystemC represents the simulation kernel and library. *gSysC* based on the graphic library provides besides graphical presentations of the design run-time access to the simulation. It introduces a simulation controller shown in figure 2 that provides single-step simulation, simulation of a certain number of clock cycles, or conditional break points, e. g. the simulation halts at a certain signal value. For an automatic and continuous simulation the time delay of each cycle can be determined. Further on, the GUI shows the simulated models with its modules and allows the user to browse the different levels and properties of modules and signals.

The GUI is attached to the simulation program by preprocessor macros. They register

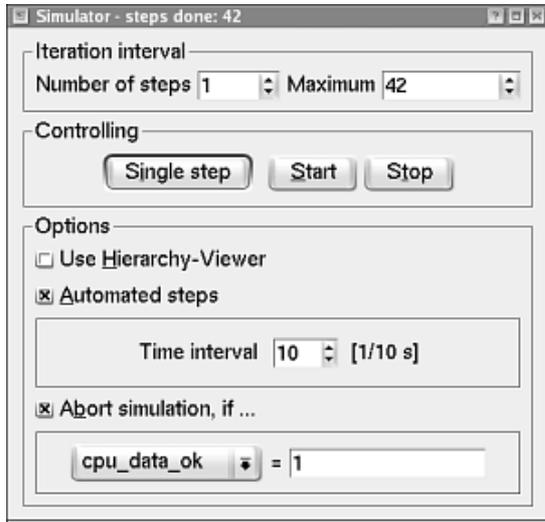


Figure 2: Simulation Controller.

each module and signal ports as well as their relation to others. Especially for demonstration purposes, the programmer can focus on important parts and leave out less important ones. The macros include code to indicate any value changes to the GUI kernel. The intervals of indication can be configured as a number of clock cycles. The SystemC-equivalent port classes in the lower layer are derived from them and apply the SystemC classes. *gSysC* takes control of the simulation system by overloading SystemC's control functions such as `sc_start()`. A base set of library methods can be used to show certain details within a simulation run, e.g. fill level of buffers. Due to the variety of simulation opportunities the programmer can enlarge this set by own application-specific extensions.

In case of an almost bug-free SystemC simulation model, *gSysC* can be removed at compile time by setting a define flag. Then, an unmodified SystemC simulation is created. This is suitable for long-term runs in particular.

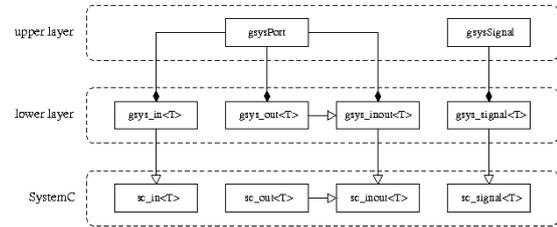


Figure 3: Architecture of the Interface Between *gSysC* and SystemC.

## 4 IMPLEMENTATION AND APPLICATION

The implementation of *gSysC* has to perform a balancing act between SystemC, Qt and its goals previously defined. The avoidance of additional programmer's code for visualisation leads to a hierarchical structure for the interface between SystemC and *gSysC*. Nevertheless, applying *gSysC* to SystemC models requires some additional code lines.

### 4.1 Interface Between *gSysC* and SystemC

The interface architecture is shown in figure 3 including all classes and their relations. It is divided into two layers. In the lower layer, which is next to SystemC, port and signal classes of SystemC are derived in order to receive new values written on them. In the upper layer there are port and signal equivalents of *gSysC* for processing purposes in *gSysC*. In addition to the values, the *gSysC* classes provide information and functions for visualisation. The included functions perform port and signal highlighting, emphasize the position and the connected neighbour modules, and open property information windows. The lower layer contains all classes derived from SystemC. These wrapper classes are necessary to achieve a data-type independent implementation and to get the opportunity to easily use sets and lists, even with ports and signals of different data types. Access to the SystemC layer is only performed by the derived *gSysC* classes so that all

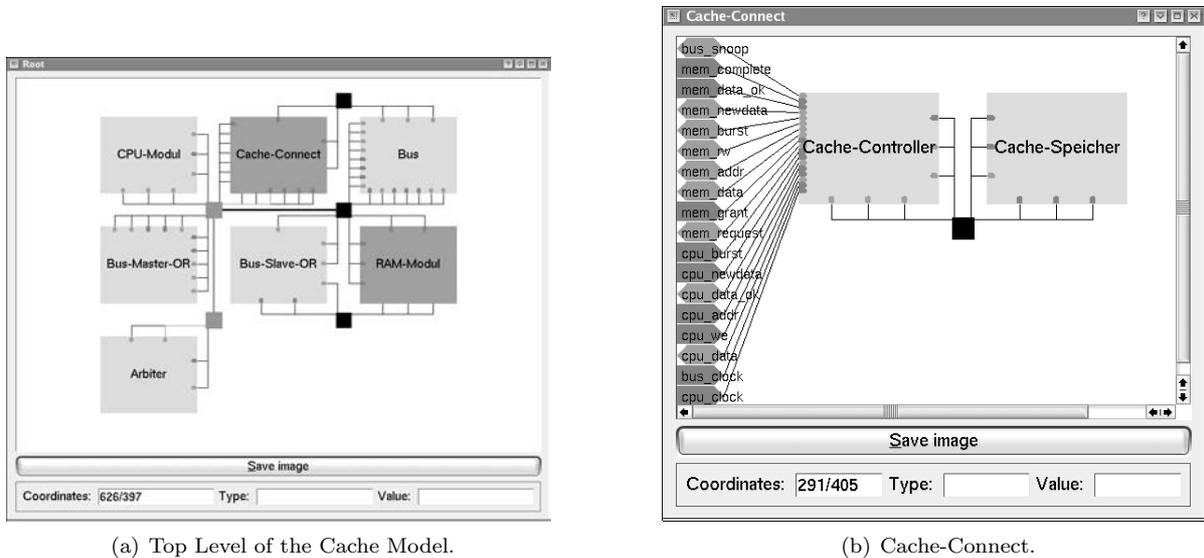


Figure 4: SystemC Model Visualisation of *gSysC*.

value changes can be tracked.

## 4.2 Registration

Hardware modules may have a huge number of interfaces for control and data exchange purposes. So, there are a couple of reasons to leave out some ports and signals respectively in the visualisation. Fewer observed objects ease the overview of the demonstrated design, allow easier debugging by concentrating on the chosen view, and speed up the simulation. All tracked parts of the design must be registered for the visualisation. So, the registration can become a clumsy procedure, especially for large designs. The programmer has to add a code line per module or port that activates *gSysC* features for this object and integrates it into the hierarchical structure of the design. The provided macros reduce the effort to a minimum:

- `REG_MODULE(module, name, parent)` registers the module on the subsequent level of `parent`. Root-level modules are indicated by the `NULL` pointer. The `name` is used in the visualisation.

- `REG_PORT(port, module, signal)` activates the visualisation of the used ports in the SystemC design. The `port` of `module` is connected to the `signal`. More distinguished macros such as `REG_IN_PORT(port, module, signal)`, `REG_OUT_PORT(port, module, signal)`, and `REG_INOUT_PORT(port, module, signal)` including the direction of the ports are available as well.
- `RENAME_SIGNAL(object, name)` and `RENAME_PORT(object, name)` allow the programmer to give signals and ports self-documenting names.

Assuming that all modules, ports, and signals are supposed to be visualised, the registration process could be performed in a preprocessing step. But, at the moment, there has not yet been provided any support tool.

## 5 CASE STUDY

The hardware design of this SystemC model consists of a CPU directly connected to a memory

cache and bus modules. The cache sends to and receives data from the RAM module using a simple bus. Figure 4(a) shows all modules of the top level model. Here, one can see how the modules with ports and interconnections are displayed. The place and route strategy is simplified using clustered signals with central crossing points. During the simulation used signals, ports, and modules may be highlighted. Figure 4(b) discloses the module 'Cache-Connect' of figure 4(a). It is made of two linked modules unseen on the top level view and has a number of in and out ports shown on the left side. If the view of a module is opened its interior behaviour is highlighted as well during the simulation.

The simplified program code of the SystemC model presented here shows the important parts of applying *gSysC*:

```
#include "gsysc.h"
#include ...

int sc_main(int argc, char* argv[])
{
    sc_clock cpu_clk("CPU-Clock");

    // signal declarations
    sc_signal<sc_bv<32> > addr_sig;
    sc_signal<bool> we_sig;
    ...
    cache_connect* c;
    ...

    REG_MODULE(c, "Cache-Connect", NULL);
    REG_MODULE(c->ctrl, "CController", c);
    REG_MODULE(c->memory, "CMemory", c);

    sc_signal<bool> bus_clk_sig;

    bus_bus b("bus");
    b.m_dt(or_mb_dt);
    b.clk(bus_clk_sig);

    REG_MODULE(&b, "Bus", NULL);
    REG_INOUT_PORT(&b.m_dt, &b, &or_mb_dt);
    ...

    c->cpu_clk(cpu_clk);
    c->bus_clk(bus_clk_sig);
}
```

Port name	Value	ID
1 cpu_addr	00000000000000000000000000000000	136372760
2 cpu_newdata	0	136373616
3 mem_request	000	136374400
4 mem_burst	011	136376704
5 mem_complete	0	136378104
6 mem_grant	0	136374904
7 mem_newdata	1	136377136
8 mem_data_ok	0	136377712
9 mem_rw	1	136376152
10 mem_addr	000000000000000000000000000011000000	136375688
11 cpu_burst	001	136374008

mem\_data Add Remove

Figure 5: Table of Signals for Value Tracking.

```
REG_IN_PORT(&c->cpu_clk, c, &cpu_clk);
REG_IN_PORT(&c->bus_clk, c, &bus_clk_sig);

bus_master_or mor("master_or");
REG_MODULE(&mor, "Bus-Master-OR", NULL);
...

return 0;
}
```

First, clock and signals for module interconnection are declared. Then, the modules are defined and their ports are connected to the signals. Last, the module and its ports are registered. Here, `cache_connect` is built of a controller and memory so that these modules are registered for a subsequent level. These three steps are done for all modules. After the declaration, connection, and registering phases, the usual SystemC code which is not shown here is required.

One can keep an eye on port and signal values utilizing their property windows or watching the full port/signal list. A snapshot of this list is shown in figure 5.

In combination with the existing opportunities such as VCD trace files and text messages, SystemC becomes with *gSysC* a more powerful tool for debugging and functional verification. Nevertheless, the costs of *gSysC* are partially high. The simulation run-time can be slowed down about 50 per cent depending on used options. Because of the opportunity to remove *gSysC* from the SystemC-model code, long-term simulations without GUI are performed without

any performance loss.

## 6 CONCLUSION

*gSysC* is a GUI extension for SystemC based on Qt, a platform independent GUI library for C++. The shown extension does not alter the SystemC library including the simulation kernel. The features for graphical representation are introduced by macros and redefined functions overloading but calling the ones provided by SystemC. As expected, the performance decreases by about 50 per cent in several cases. The library is open source and can be found at the web pages of the Institute of Computer Engineering, University of Lübeck ([www.itl.uni-luebeck.de](http://www.itl.uni-luebeck.de)).

## REFERENCES

- [Cadence 2005] Cadence Design Systems Inc., *Incisive Unified Simulator*, Datasheet 5418C 04/05, 2005.
- [Charest et al. 2001] Charest, L.; M. Reid; E.M. Aboulhamid; and G. Bois. 2001. *Methodology for Interfacing Open Source SystemC with a Third Party Software*. Proceedings of Design Automation and Test in Europe Conference & Exhibition, 16-20, Munich, Germany.
- [Eibl 2004] Eibl, C.J. 2004. *gSysC – Visualisierung von SystemC-Projekten*. Student Project, Institute of Computer Engineering, University of Lübeck, Germany.
- [Große et al. 2003] Große, D.; R. Drechsler; L. Linhard; and G. Angst. *Efficient Automatic Visualization of SystemC Designs*. *Forum on Specification & Design Languages*, Frankfurt, Germany.
- [Prosilog 2005] Prosilog, *Magillem*, Product Brief, 2005.
- [Reid et al. 2001b] Reid, M.; L. Charest; M. Aboulhamid; G. Bois; and A. Tsikhanovich, 2001. *Implementing a*

*Graphical User Interface for SystemC*. Proceedings of the 10th International HDL Conference, 224-231, Santa Clara, CA, USA.

[SystemC 2002] Open SystemC Initiative (OSCI): *SystemC Version 2.0.1 User's Guide*. Technical Report, 2002.

[Trolltech 2002] Trolltech AS. *Qt 3.1*. Whitepaper, 2002.

# SIMULATION OF MUSICAL CONTENT BY 3-D VISUALISATION

Jacek Grekow  
Faculty of Computer Science  
Technical University of Bialystok  
Wiejska 45A, Bialystok 15-351, Poland  
E-mail: grekowj@ii.pb.bialystok.pl

## ABSTRACT

Audio information we receive is a complicated phenomenon. Consequently, an extensive analysis of a piece of music is a complex task due to the sheer number of data attributes such as tempo, rhythm, pitch, consonance, harmony, dynamics. That is the reason why all the research conducted to date in this field has concentrated solely on selected parameters, and that fact determined the aim of the experiments as well as their results. This paper presents a totally new approach to creating 3-D images to the analysis and simulation of musical content. As the main criterion for the synthesis of the 3-D images we have chosen harmonic content, which has close relation to the musical content of a piece. Additionally, the graphic display of its main components in shape of 3-D figures is a crucial aspect in the presented approach. The figures, which were named AKWETs, are applicable in simulation and analyse musical content. The paper also states that the the order of AKWETs identified in a given phrase forms a meaningful visual sequence, connected with its musical original.

## INTRODUCTION

The following questions have been asked for a long time: why do people admire a particular piece of music, whilst are not attracted to another? What criteria need to be fulfilled for the listener to accept or reject a given performance? By what means do people's hearing and brain analyse the information carried by sound? Only recently academic papers have emerged that study the musical content with application of computer techniques.

Audio information we receive is a complicated phenomenon. Consequently, an extensive analysis of a piece of music is a complex task due to the sheer number of data attributes such as tempo, rhythm, pitch, consonance, harmony, dynamics. That is the reason why all the research conducted to date in this field has concentrated solely on selected parameters and that fact determined the aim of the experiments as well as their results. Some authors analyse different performances of one piece of music, restricting their attention to two expressive dimensions: tempo and loudness (Dixon et al. 2002). Typical example, where musical and spatial parameters are related one to another and form an interactive visual tool are also served (Graves et al. 1997).

This paper presents a an entirely new approach to

creating 3-D images to the analysis and simulation of musical content. As the main criterion for the synthesis of the 3-D images we have chosen harmonic content, which has close relation to the musical content of a piece. Additionally, the graphic display of its main components in shape of 3-D figures is a crucial aspect in the presented approach. The figures, which were named AKWETs, are used to simulation and analyse musical content. The paper also states that the the order of AKWETs identified in a given phrase forms a meaningful visual sequence, connected with its musical original.

Visualisation is often used in academic research to illustrate relation between various parameters. Applied to analysis of musical content, visualisation reveals its aesthetical value, as it combines two independent human senses. What is more, it helps obtain information that is not always clearly perceived when considerable amount of data is given. We can assume that we obtain information of a different level and our cognitive potential extends. Consequently, the presented AKWETs enable us to:

- demonstrate correlation between 3-D images and music
- obtain additional information about chords - their type, internal consonance and dissonance etc.
- draw conclusions about how chords follow in succession
- get a new look at the overall structure of a piece of music

This area is still open for further research.

## METHOD OF CREATING 3-D IMAGES

Harmonic content of a piece of music consists of chords that are related to its consecutive segments, and that relation depends on the sounds of a chord as well as other principles of musical theory (Sikorski 1965).

The main idea of the method presented is based on chords. As mentioned above, chords divide the analysed musical content into segments, the length of which is not equal. It is determined by the mentioned principles (Sikorski 1965). The segments may include a part of a bar or several bars, so that the difference in length between particular segments may be considerable. Our method of audio analysis is completely different in comparison with the traditional ones (Czyżewski 2001).

The fact that most chords contain three notes or it is possible to choose three crucial components from a

multi-note chord was taken into account (Sikorski 1965).

Every musical note can be represented in a simplified form as a sine curve of frequency  $f$ . Consequently, each component of chord  $A$  is described by means of sine function  $S_i(t)$  of frequency  $f_i$ .

Each function obtained is associated with one of the axes (X, Y or Z) of Cartesian coordinates  $\{U\}$ , and a 3-D figure  $\Phi_A$  assigned to chord  $A$  is formed.

$$A \rightarrow \Phi_A \quad (1)$$

We name figure "AKWET" = AKordWizualnyETalon.

The forming of each AKWET may be described as the result of the movement of point  $P$ , spatial position of which is determined by function  $S_i(t)$  values put on X, Y and Z axes respectively (Figure.1).

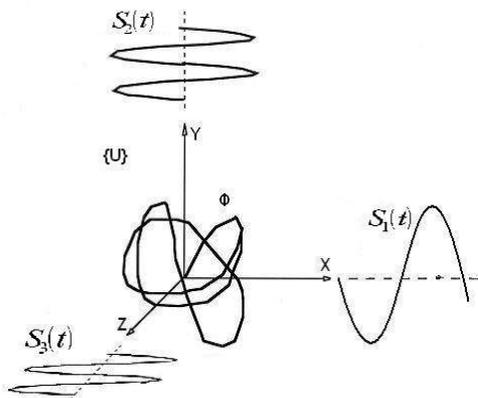


Figure 1: Forming of an AKWET

Each axis of the  $\{U\}$  coordinates is related to a sinusoidal signal  $S_i(t)$  of frequency  $f_i$  where  $i = 1,2,3$ . The signals illustrate consecutive components of the AKWET. When digitised with frequency  $F \gg f_i$ , the signals  $S_i(t)$  form a sequence of samples  $s_{ji}$ . Each element of the sequence is a three-dimensional vector that determines position of point  $P_j$ .

The coordinates of a single point  $P_j (P_{jx}, P_{jy}, P_{jz})$ , which are calculated on the basis on common time  $t$ , and the signal function  $S_i(t)$  may be recorded as follows:

$$\begin{aligned} P_j &= (P_{jx}, P_{jy}, P_{jz}) \\ P_{jx} &= S_1(t_j) = A \sin \omega_1 t_j \\ P_{jy} &= S_2(t_j) = A \sin \omega_2 t_j \\ P_{jz} &= S_3(t_j) = A \sin \omega_3 t_j \end{aligned} \quad (2)$$

Since all the components taken into account are equally important in an AKWET's structure, we investigate three functions of identical amplitude  $A$ .

AKWET  $\Phi$  is a spatial and periodic figure (Figure 1) and its period depends on the values of component

signals.

## ACHIEVEMENTS TO DATE AND FUTURE WORK

AKWET's structure is similar to Lissajous figures (Szczeniowski 1972). Significant difference lies, however, in the fact that we achieve 3-D figures, which are then used to visualise and analyse sophisticated musical content.

### Correlation between AKWET's shape and the sound of chords

Classical music theory uses two basic keys – minor and major, to describe the scale of a particular piece. The keys are represented by minor and major chords, the sound of which substantially differs; simplifying the matter we can state that a minor key is "sad", while a major – "cheerful" (Sikorski 1965). We can consider keys basic AKWETs.

It results that the figures representing major and minor chords respectively differ significantly (see Figure.2). The major chord's shape is simpler and clearer. The minor chord, on the other hand, is denoted with a more complex, multidimensional figure, which cannot, despite its regular shape, be associated with peace.

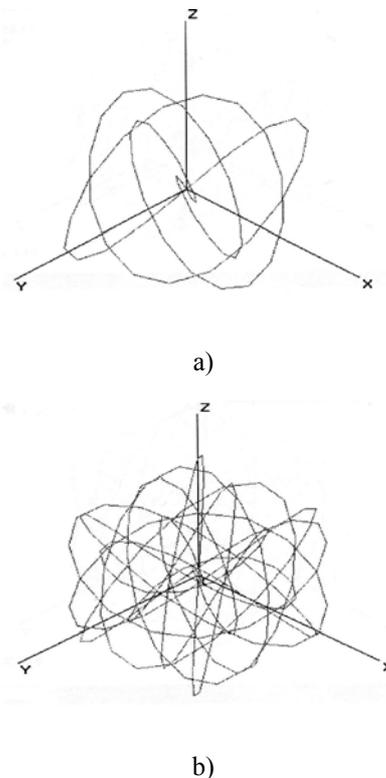


Figure 2: Model patterns a) major AKWET b) minor AKWET

This example is the first evidence that the AKWET is based on the right criterion.

Further development of AKWETs might make use of another criterion – aesthetic qualities of a 3-D image.

**Additional information about chords that might be extracted from the figures' shape**

An AKWET simulator has been designed and built for further experiments and all the basic, according to the musical laws (Sikorski 1965), chords have been tested. Furthermore, simulator-generated graphics originated ideas of new AKWETs. Figure 3 shows several model patterns depicting internal consonance (a) and dissonance (b) as well as two additional AKWETs (c) and (d), not subjected to musical laws, but interesting for further research.

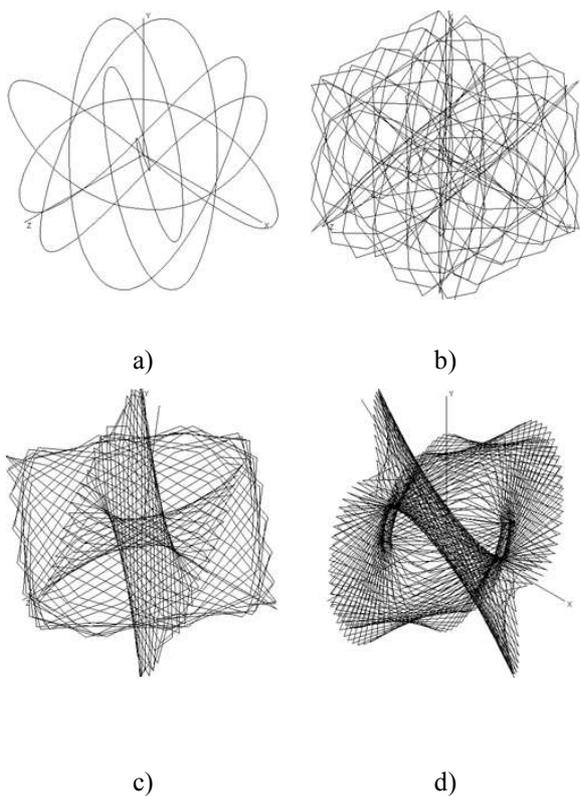


Figure 3: Several model AKWETs: a) consonance, b) dissonance, c) new 1, d) new 2.

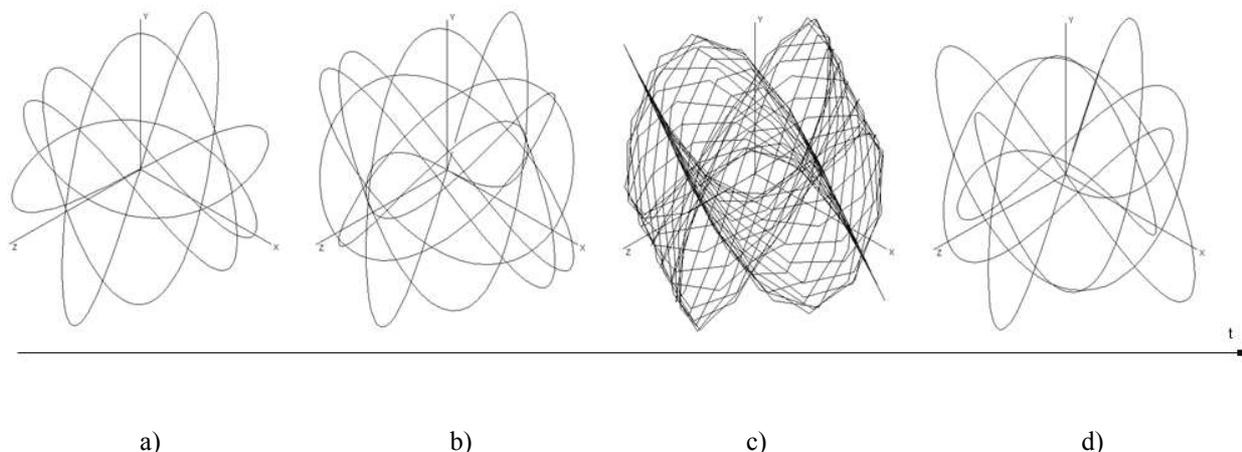


Figure 4: Visualisation of a model cadence  $K_d$  (T S D<sup>7</sup> T): a) T (I), b) Subdominant (IV), c) Dominant<sup>7</sup> (V<sup>7</sup>), d) T (I).

**Simulation of musical content**

The sequence of appearance of static 3-D images is an equally interesting field of research. One of the basic patterns of development of a musical piece is called cadence (Sikorski 1965). Figure 4 enables us to visually track the development of a major cadence  $K_d$  (T S D<sup>7</sup> T). This cadence is a common harmonic description of the ending of a musical piece. Consecutive AKWETs form a visual sequence that is temporally connected with the musical original.

Each chord from the sequence  $K_d$  has meaningful content reflected in a visual sequence: introduction, development, climax and denouement. A developed AKWET simulator can play the sequences of chords and simultaneously draw figures, creating the videoclip of a musical piece. The order of AKWETs identified in a given phrase forms a meaningful visual sequence, connected with its musical original.

**CONCLUSIONS**

The method presented in this paper is an innovative approach to simulation and analyse pieces of music. It is based on combining two human senses: hearing and vision, which results in mutually linked abstract figures that illustrate music and to analyse musical content (AKWETs).

The AKWET method may be applied in:

- visualising music;
- studies and prediction of the impact of music on listeners;
- supporting composer's work in evaluating combination of chords in a composition by visual simulation

We are certain that continuation of this research may bring, especially when the computer programmes are developed, new discoveries in the area that combines music with technology.

## REFERENCES

- Czyżewski A. 2001. *Dźwięk cyfrowy*. EXIT, Warszawa, 3-40.
- Dixon S.; Goebel W. and Widmer G. 2002. "Real time tracking and visualisation of musical expression." Technical Report 2002-04, Austrian Research Institute for Artificial Intelligence.
- Graves A.; Hand C. and Hugill A. 1997. "Interactive Visualisation of Musical Form using VRML" *The Fourth UK VR-SIG Conference*, Brunel University.
- Grekow J 2004. "The method of conversion an audio signals." *Digital Signal Processing and its Applications*, Moskwa.
- Sikorski K. 1965. *Harmonia*. Polskie Wydawnictwo Muzyczne, Kraków, 25-99.
- Szczeniowski S. 1972. *Fizyka doświadczalna*. Wydawnictwo Naukowe PWN, Warszawa, 120-129.



**JACEK GREKOW** was born in Białystok, Poland. In 1994 graduated from the Technical University of Informatics Sofia, Bulgaria. Graduated also of the Vienna Conservatory and after obtaining his musical degree in 1997, he worked for a year in IBM Vienna, as computer programmer. Since 1998 he worked for a couple of years as computer programmer in T-matic Company before moving to the Technical University in Białystok, where he is now doing his researches in music visualisation and analysis. His e-mail address is: grekowj@ii.pb.bialystok.pl.

This paper is supported by the S/WI/5/03

# METHODS TO LEAD THE USER TO SIGNIFICANT PROCESSES IN A 3D MATERIAL FLOW SIMULATION

Wilhelm Dangelmaier  
Bengt Mueck  
Matthias Fischer  
Kiran R Mahajan  
Christoph Laroque

Heinz Nixdorf Institute, University of Paderborn  
Fürstenallee 11  
33102 Paderborn, Germany  
{ whd, mueck, mafi, kiran, laro }@hni.uni-paderborn.de

## OBJECTIVES

Simulation of material flow process and the visualization are quite common tools to improve the planning of new plants or the reworking of existing processes. 3D-visualisations of such simulation runs are more and more common. This allows a faster understanding and better communication of the processes. However, up to now this analysis process is carried out unguided. A user does not know where he has to look in a virtual environment. For this, we developed a set of helping tools to support the user during the analyzing process (like not synchronized cycle times). We are detecting processes that are going wrong and being good candidates for taking more attention of the user. After this detection, we try to show the user these candidates and give him the option to get guidance to the processes with automatically generated additional elements in the 3D environment.

## STATE OF THE ART

Today many material flow simulation tools offer integrated 3D-modules (Klingstam and Gullander 1999 or Mueck and Dittmann 2003). Examples are simulation tools such as QUEST by Deneb, Taylor ED by Enterprise Dynamics (Nordgren 2001), eM-plant by Technomatix or Automod with integrated virtual environments. The user can build a model, see, and analyze it in a 3D environment. However, an active support for the user does not take place. If the viewer moves in the relevant system, he cannot experience critical processes that take place in his back. They are not visible for him. User guidance to significant points is unknown. The locating of critical objects is thus left to the experience of the viewer or based on random.

## SIGNIFICANT PROCESSES

If a material flow process is sketched and modeled, a simulation run can be used to find out processes, which are not quite well balanced. E.g., a machine runs empty of incoming parts, a machine breaks down or a stock is much too big. Like this, several other rules for “not well balanced behavior” can be defined. If something like this happens, it could have a huge influence on the simulation run and especially on the key production figures like parts in process or throughput time. If a change of a process leads to a fast changing production figure, we call the process significant. So our significance is based on a dynamic process. In addition, the level of significance can change during the simulation run.

If the user wants to improve these production figures, the significant processes are the processes he should take care of. In our approach, we detect these processes automatically and help the user to solve potential problems.

Each significant process belongs to a machine, a stock or any other static part of the simulation model. If we talk in this paper about a significant machine, we also include all the other cases.

## VISUALISATION OF SIGNIFICANT OBJECTS

Significant processes have a high influence on the key production figures. Therefore, they are important for the user to watch. However, typically the user is not standing direct in front of these processes. The chances are not too bad that some other machines are hiding the significant process. In this case, the user has at least to be notified, where significant processes occur. It would be better, if he can see them directly.

## Columns on Top of Significant Processes

Typically, the user is walking through his production hall. Moreover, he cannot see very far. Machines surround him. Nevertheless, he can look over the machines. In this approach, the machines belonging to significant processes are getting columns on top. These columns are high enough that the user can see them from every point of the scene. Because all columns have the same wide, the user can easily see which are close to him. Additional captions on the columns can show the name of the significant machine.

## Semitransparent Visualization

If the user is close to a significant process but an object hides the process, parts of the object will be visualized semitransparent in this approach. Therefore, the user can see the significant process through a hole in the object (c.p. Figure 1). If the user is moving, the hole is following (in the object) him.

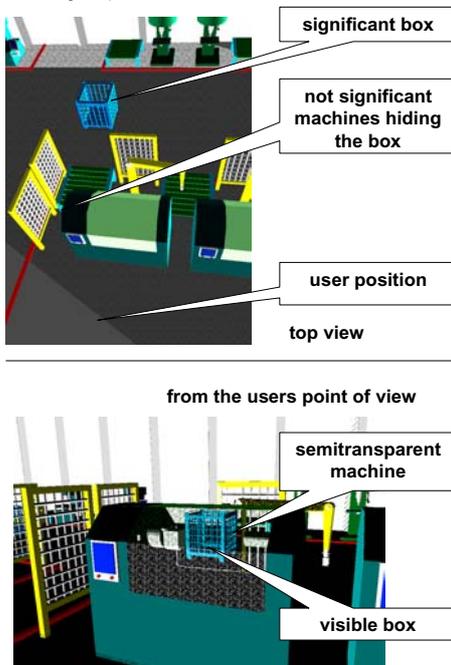


Figure 1: Machine is hiding the box. Parts of the machine will be visualized semitransparent.

For significant processes that are close to the user this is working fine. However, if there are too many objects between them, semi transparency does not work, because several semitransparent objects are leading to a non-transparency. With a complete transparency, the user cannot see the machine direct in front of him. The visualization cannot separate where the user is looking at. Therefore, it must be possible to see everything.

## Arrows Pointing to Nearby Significant Processes

Columns on the top of significant objects works well for far away objects. If the user is nearby the object, columns on the top are too high above the objects. The user cannot identify the significant object, because the column of the nearby significant object is difficult to distinguish from the columns of far away significant objects. Therefore, we use another means to point to nearby significant objects. We highlight significant objects in the scene by red arrows that are placed at the top of a significant object (see Figure 2 for an example), similar to the arrows suggested by Darken and Peterson (Darken and Peterson 2002)). Thus, the user can easily get a visual overview which of the machines is significant. The visual detection of significant machines works faster than a text message.

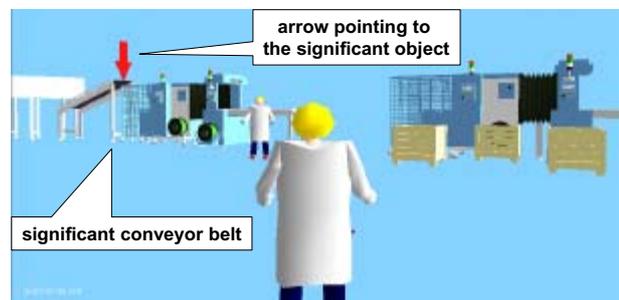


Figure 2: The conveyor belt is significant. The red arrow points to the significant object.

## GUIDANCE OF USER TO SIGNIFICANT POINTS

In traditional 3D simulation environments, the user is walking through his process. The way he is walking is in best case based on experience and some expectations. However, in many cases his way is also influence by luck. This is not structured and efficient. In our approach, the user should be guided to the significant processes. For these we developed two visualization techniques (Guidance by Lines and guidance with signs). For a better overlook, orientation, and to avoid a disorientation of the user, we implemented a minimap that shows a top view of the scene.

### Guidance by Lines

In this approach the way between the significant processes and the user are shown with lines on the ground. If the user wants to go to significant process, he only has to follow these lines. Figure 3 shows an example for this. It also shows what is happening if two or more processes are significant. In this figure, the different ways are carrying captions, which contain the name of the process. This is difficult to realize for the user. In further steps, we will work with different colored lines. Each color represents one process and the lines will be shown side by side.

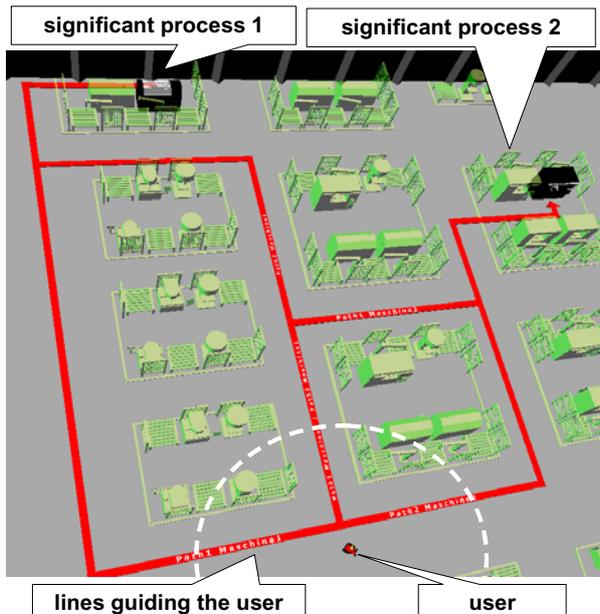


Figure 3: Lines on the ground are visualizing the shortest path to the significant processes

### Guidance with Signs

In the first approach, the user has to look on the floor. This is not where he is typically looking. He also only needs information about the way on junctions. Therefore, we developed 3D signs that guide the user to the significant processes. On each junction, these signs or arrows are showing the shortest way to the most significant points to the user (cp. Figure 4). Like traffic signs, they can show more information than the direction and the name of the process. Therefore, it is possible to provide the user with additional information (e.g. the distance or the level of significance).

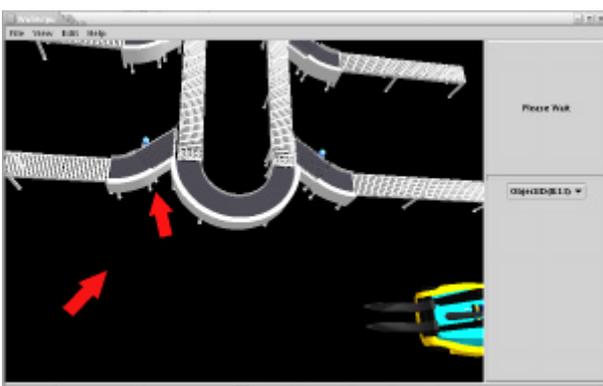


Figure 4: Arrows showing the user the way to a significant process

Both approaches (guidance by lines and guidance by signs) are implemented in our software. However, the effects for the user are not yet fully analyzed. Especially the speed measurements for the orientation of the user have to be made in the close future.

### Overall View to Orientate the User

A typical problem of 3D environments is the orientation in the virtual world. Especially in environments of material flow processes, e.g., in a large factory hall, the area is large, and during the planning process, often changes so the user can lose his way and orientation. To prevent disorientation, the user must concentrate on finding the right way to the wanted destination. However, the purpose of the 3D visualization is to observe and analyze processes and not to waste time with finding and exploring a way through the scene. Therefore, the system must make the orientation as easily as possible.

Our system supports an easy orientation by the implementation of a minimap that shows an overlook of the scene. The upper image of Figure 5 shows the 3D scene and the lower image shows the corresponding minimap. The overlook of the minimap consists of a two-dimensional floor plan of the scene, e.g., the factory floor. Additionally, we draw the skeletal silhouettes of all static objects such as machines and conveyor belts, and of all moving objects such as forklifts and users. The users are rendered as thumbnails in the minimap.

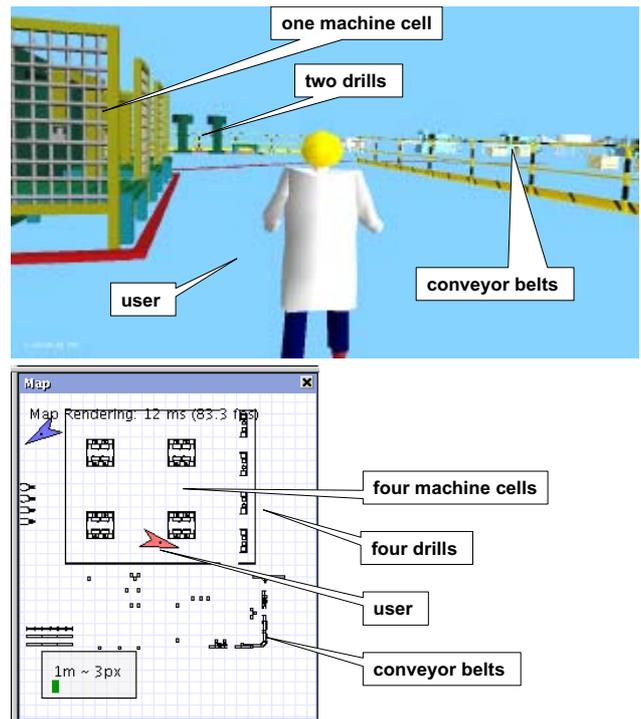


Figure 5: A user navigates through a 3D scene of a manufacturing facility seeing only one machine cell and two drills. A minimap shows the skeleton of four machine cells and four drills.

In the example rendering of Figure 5, the user sees one machine cell and two drills. On the minimap, he sees his neighborhood including all occluded objects. Therefore, he sees four machine cells and four drills. Thus, a user can easily find a way to neighbored machines that are

occluded by large machines or other objects. Without a minimap, the user would waste his time by searching a way as in a labyrinth. Our minimap supports in an easy manner:

- to find a way to the destination,
- to explore the factory and environment,
- to find neighboring users.

Currently we use arrows, which look like compass needles, as thumbnails for the users. The compass needle shows the line of sight of the user. This supports an easier orientation in the scene.

## CONCLUSION

This paper describes several methods to support a user during his analyze of a material flow process. For this first process of potential interest for the user have to be detected. This has been done on the base of performance measurements. We call these processes significant processes. The developed tool is only a supporting system. We cannot detect everything the user might be interested in. Therefore, we offer our significant processes to the user and give him the option to take a closer look on them. For this, some methods to show him the location of the processes and the way to the processes have been discussed in this paper.

Such methods are quite new to material flow simulations. The developed methods do not claim that they are optimal or even complete. Especially measurements of the speed up of the work with and without our methods are still missing (we are planning to do this in the near future). If it is possible to find problems faster with these visualization techniques, this could reduce the efforts and the costs for analyzing material flow simulations.

## LITERATURE

- Darken, R. P. and Peterson, B. 2002 Spatial Orientation, Way-finding, and Representation, In Handbook of virtual environments : design, implementation, and applications, ed. K. M. Stanney, 493-518. New Jersey: Lawrence Erlbaum Associates
- Klingstam, P. and Gullander, P. 1999. Overview of simulation tools for computer aided production engineering, In: Computers in Industry, No. 38, p. 173-186
- Mueck, B; Dittmann, N.: Marktanalyse Materialfluss-Simulatoren, Band 11, ALB-HNI Verlagsschriftenreihe, Paderborn, 2003
- Nordgren, W. B.: Taylor Enterprise Dynamics, In: Peters, B. A.; Smith, J. S.; Medeiros, D. J. and Rohrer, M. W. (Hrsg.): Proceedings of the 2001 Winter Simulation Conference, p. 269-271, SCS-international, San Diego, California, 2001

## AUTHOR BIOGRAPHIES



**Wilhelm Dangelmaier** was director and head of the Department for Cooperate Planning and Control at the Fraunhofer-Institute for Manufacturing. In 1990 he became Professor for Facility Planning and Production Scheduling at the University of Stuttgart. In 1991, Dr. Dangelmaier has become Professor for Business informatics at the HEINZ NIXDORF INSTITUTE; University of Paderborn, Germany. 1996, Prof. Dangelmaier founded the Fraunhofer-Anwendungszentrum für Logistikorientierte Betriebswirtschaft.



**Bengt Mueck** studied computer science at the University of Paderborn, Germany. Since 1999 he is a research assistant at the group of Prof. Dangelmaier, Business Computing, esp. CIM at the HEINZ NIXDORF INSTITUTE of the University of Paderborn. At the end of 2004 he received his phd in the area of discrete materialflow simulations. His main research interests are logistics systems and tools to simulate those systems.



**Matthias Fischer** studied computer science at the University of Paderborn, Germany. Since 1995 he is a research assistant at the Heinz Nixdorf Institute. In 2005, he received a PhD for his work on distributed virtual environments. His main research interests are computer graphics, real-time rendering algorithms, and distributed computing.



**Kiran R. Mahajan** studied Mechanical Engineering at the Delft University of Technology in the Netherlands with a specialization in production engineering. Since 2004 he is a research assistant and a PhD student at the graduate school of dynamic intelligent systems at the University of Paderborn. His PhD work and his research interests are “interactive analysis of 3D material flow simulations based on significant processes”.



**Christoph Laroque** studied business computing at the University of Paderborn, Germany. Since 2003 he is PHD student in the graduate school of dynamic intelligent systems and research assistant at the group of Prof. Dangelmaier, Business Computing, esp. CIM. He is mainly interested in material flow simulation models and the “digital factory”.

# A MIXED REALITY FRAMEWORK FOR VISUALIZATION AND EXECUTION OF DEVS-BASED SIMULATION MODELS

Arnis Lektuers  
Riga Technical University  
1 Kalku street, Riga, LV-1658, Latvia  
E-mail: arnis@itl.rtu.lv

## KEYWORDS

Augmented Reality, Mixed Reality, Simulation Modeling, Discrete Event System Specification.

## ABSTRACT

A mixed reality (MR) represents an environment composed both by real and virtual objects. MR applications are more and more used in different scientific and practical areas, for instance, in medicine, architecture, entertainment, manufacturing, etc. This paper describes development of mixed reality framework for visualization and execution of discrete event simulation models based on the Discrete Event System Specification (DEVS) formalism. The mixed reality framework provides a 3D augmented/virtual environment representing and executing simulation models that encapsulate the static and dynamic behavior of the simulated system. Basing on the DEVS formalism, there is also presented a new methodological concept, called Visual-DEVS (VDEVS).

## INTRODUCTION

Simulation is an important tool in the analysis and design of complex systems. Various types of simulation techniques and tools have been developed, in order to deal with complex systems. In order to help better to understand the behavior of complex systems, visualization tools have a crucial role that facilitates problem solving and decision-making. Combining real and virtual visualization objects in a mixed reality (MR) environment allows model users better to interact with the objects in the simulation environment. Today there exist several works, dealing with support of virtual, augmented and mixed reality for simulation and modeling (Fischer et al. 2003; Rehn et al. 2004). Formalism, called Discrete Event Specification (DEVS) (Zeigler 2003) is gaining popularity in recent years, enabling to construct discrete-event hierarchical modular simulation models. In this work there is introduced a visual extension for DEVS that enables 2D and 3D mixed reality simulation visualization. 3D mixed reality simulation visualization framework is developed basing on the presented VDEVS formalism.

## Mixed Reality

The concept *Mixed Reality* is based on the classification presented by (Milgram and Kishino 1994) and (Ohta

and Tamura 1999). Mixed reality refers to the combination of computer graphics and real-world objects. This encompasses both augmented reality and augmented virtuality that places computer graphic objects into the real world placing real-world objects into virtual environments.

A wide range of mixed reality applications have been proposed, ranging from industrial to medical and entertainment applications. This framework is primarily concerned with one particular type of application that is known as monitor-based (non-immersive) or “window-on-the-world” (WOW) (Milgram and Kishino 1994) augmented reality. In this paradigm the data is passed from the video camera to the computer, modified and displayed in the monitor in real-time.

## DEVS Formalism

Discrete Event System Specification (DEVS) formalism (Zeigler 2003) is a general formalism for modeling discrete event system. A basic DEVS model has such a structure:

$$AM = \langle X, S, Y, \delta_{int}, \delta_{ext}, \lambda, ta \rangle$$

Where

$X$  is the set of input values,

$S$  is a set of states,

$Y$  is the set of output values,

$\delta_{int} : S \rightarrow S$  is the internal transition function,

$\delta_{ext} : Q \times X \rightarrow S$  is the external transition function, where

$Q = \{(s, e) \mid s \in S, 0 \leq e \leq ta(s)\}$  is the total state set,

$e$  is the time elapsed since last transaction,

$\lambda : S \rightarrow Y$  is the output function,

$ta : S \rightarrow R_{0,\infty}^+$  is the time advance function.

The interpretation of these elements is illustrated in Figure 1.

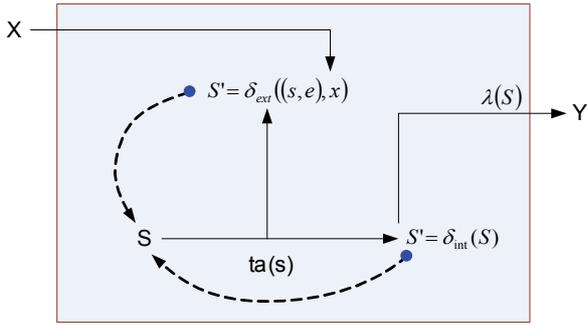


Figure 1: Elements of DEVS Structure (Zeigler 2003)

### VDEVS

In order to enable linking the DEVS formalism and visualization concepts into unified visual simulation framework, a Visual-DEVS (VDEVS) methodology is proposed. VDEVS extends the DEVS formalism supporting 2D/3D visualization of simulation models. A VDEVS model can be described as:

$$VM = \langle X, S, Y, \delta_{int}, \delta_{ext}, f, \lambda, ta \rangle$$

$S = \langle S^{disc}, S^{cont} \rangle$  is the sequential state set,

$S^{disc}$  is the discrete state set,

$S^{cont} = \langle T, G, V \rangle$  is the geometric

visualization set,

$$T = \begin{bmatrix} R & P \\ 0 & 0 & 0 & 1 \end{bmatrix} \text{ are the local}$$

coordinates,

$R$  is the local orientation (3x3 matrix),

$P$  is the local position (3x1 vector),

$G$  is the geometry data,  
 $V = \langle Color, Texture \rangle$  describes the visualization attributes,

$f : Q \rightarrow S^{cont}$  is the rate of change function.

In order to model the continuous change required by animation purposes, the sequential state set  $S$  is decomposed into the discrete state and continuous state (Figure 2).

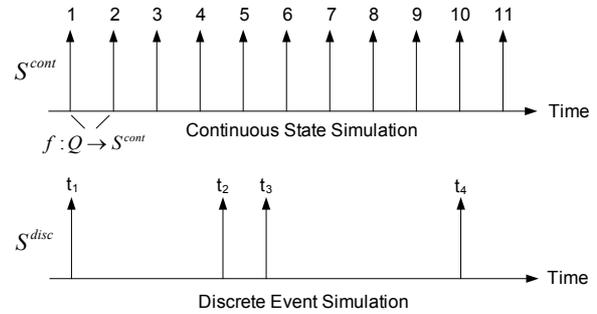


Figure 2: Continuous State and Discrete Event Simulation

### SYSTEM ARCHITECTURE

The proposed framework implements the VDEVS formalism and consists of several components supporting simulation design, execution and control, as well as 2D/3D visualization and mixed reality interaction capabilities. The architecture of the given system is made by four base components – modeling module, visualization module, DEVS simulation module and mixed reality module. In Figure 3 is

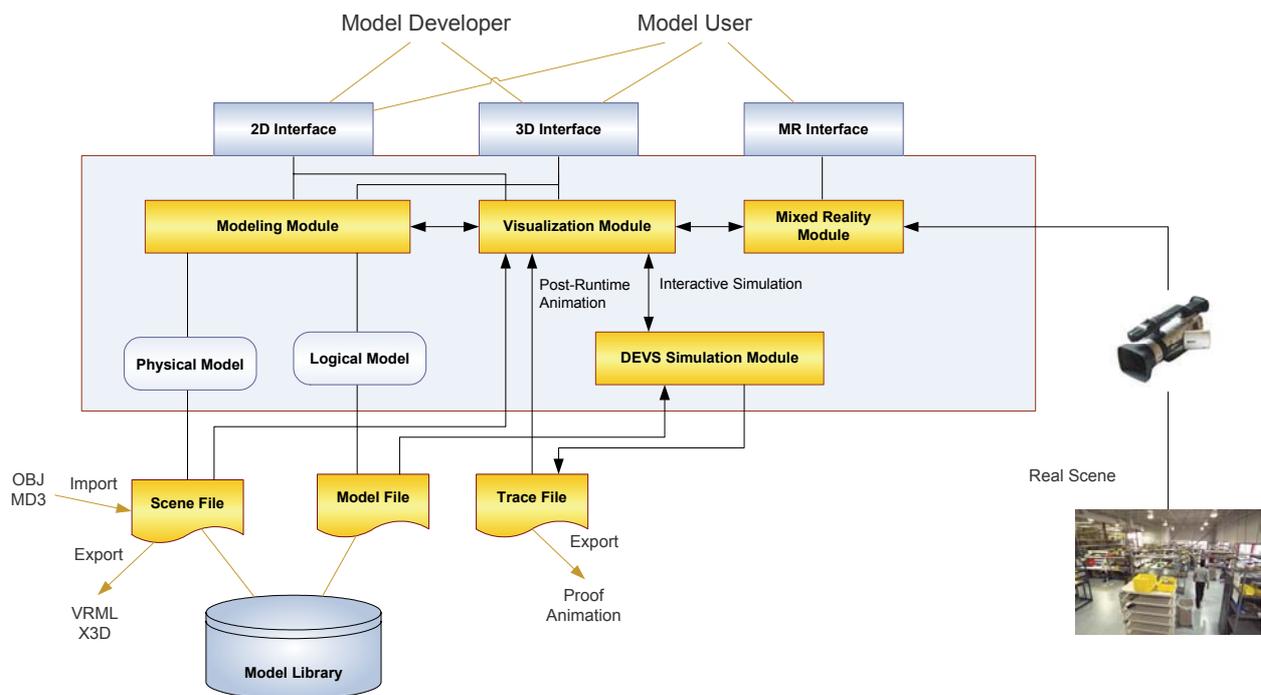


Figure 3: Architecture of Mixed Reality DEVS-based Simulation Environment

illustrated the architectural structure of the proposed MR simulation environment.

### Modeling Module

The modeling module is provided for building of simulation models in virtual 3D modeling environment, including static and dynamic images, as well as 3D geometric objects. Simulation models are based on the physical and logical description of the simulated system. The physical model defines the layout and geometry of virtual scene as well as the visual attributes of virtual objects. It is possible to make the physical model with the tools that are built in the virtual modeling environment or to import from external 3D modeling and CAD software, using widely used 3D file formats, such as MD3, OBJ. The created physical model is saved in scene file or exported, for example, in X3D or VRML formats for use in other 3D software tools.

The logical model contains all the necessary information about the dynamical behavior of the simulated system. For visual process-oriented as well as DEVS based simulation modeling purposes there are suitable 2D/3D simplified hierarchical activity cycle diagrams (SH-ACDS) (Odhabi et al. 1998; Lektauers 2003). To each functional simulation object corresponds a determined 3D graphical object. With the help of 3D computer graphics it is possible to improve the symbolism and esthetical aspects of the model. By graphical linking of the chosen model objects, there is created an simulation model diagram (Figure 4) that defines the model element couplings and execution logic. Every model element has certain subordinated metadata that contains run time parameters for the given element, for example, processing duration, generation frequency etc.

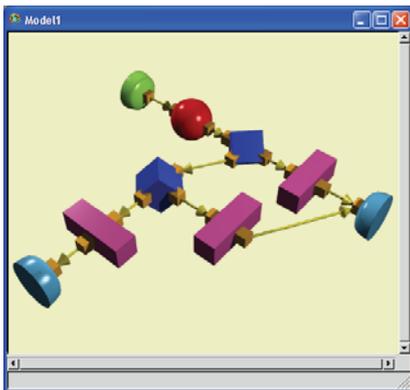


Figure 4: An Example of 3D SH-ACDS Simulation Model Diagram

### DEVS Simulation Module

DEVS simulation module is the main component of discrete event modeling system that manages the execution time processing of simulation model and the generation and acquisition of simulation results. This simulation module implements the above described VDEVS formalism and allows to combine the processes

of simulation and visualization into a unified visual simulation process. Still the functions of simulation modeling and animation are separated, therefore the simulation model can be activated independently from other system modules, but at the same time during its operating it can use visualization model, in order to provide interactive runtime animation. For execution of simulation animation there is necessary a synchronization between simulation model and visualization module. This task is directed and controlled by the given simulation system, implementing VDEVS formalism.

Model file is used as a source of input-information that contains the data about logical activities of simulation model. The results of simulation modeling execution can be saved in Trace file that allows post-runtime activating of animation without need to activate simulation process.

### Visualization Engine

Visualization module is used for generating of 3D objects in mixed reality environment (for example, assembly parts, vehicles, user interface elements), executing all the necessary tasks of computer graphics rendering and animation. Visualization model is used in all simulation modeling phases, beginning with building of simulation model, ending with representing and analysis of modeling results. One of this model's tasks is to provide realistic synthetic object illustration, so that this object can be better virtually integrated in the flow of real video frames.

Rendering of real objects in a virtual environment allows getting more united and more realistic view at simulated system activities. Visualization model includes scene graph structure concept that is well known in computer graphics. Scene graph  $G = (V, E)$  is oriented graph that contains a set of graphics nodes  $u \in V$  and links  $e(u, v) \in E$ . Scene graph contains information about the parameters and properties of graphical objects – geometry, spatial position and orientation in 3D environment, colors and texture parameters.

For the visualization system there are established high graphical rendering performance requirements because practical 3D graphic scenes contain objects with big amount of polygons that makes the rendering process more difficult. In order to provide acceptable speed of graphic depiction, there can be used level-of-detail (LOD) criteria, allowing reducing the total amount of polygons.

Visual modeling deals with static and dynamic objects (Chawla and Banerjee 2001). Static objects remain in immutable state during all the modeling phases, but dynamic objects can change their spatial or visual state during user interaction and animation execution. These dynamic objects can move and rotate or they can contain movable elements.

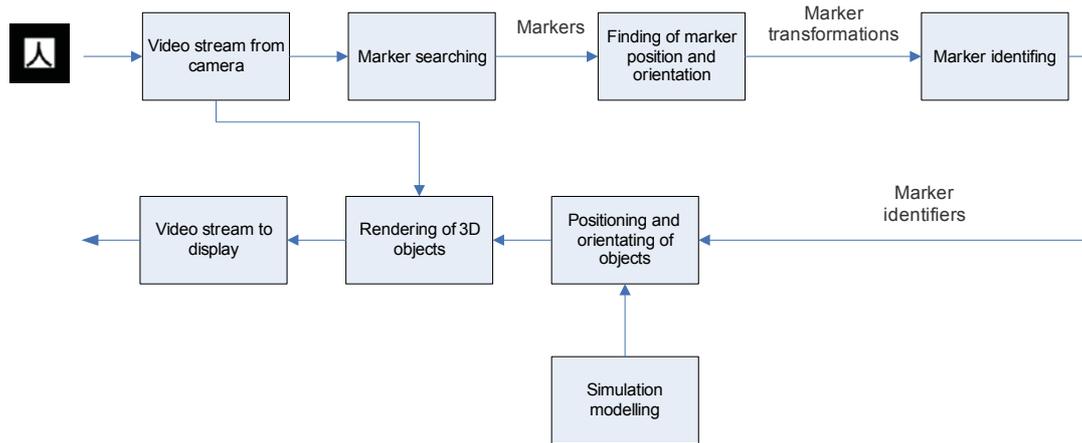


Figure 5: Image Acquisition and Processing using Mixed Reality Module

### Mixed Reality Module

The main task of the mixed reality module is to provide the independence of visualization module from different possible real world data acquisition modes. The first working phase of MR module is to obtain the real world images, using the computer-attached video camera (Figure 5). The obtained frames are sent to visualization module where they are depicted in the background of virtual environment. The mixed reality interface module creates a list of detected objects that contains the identifiers as well as data of the spatial position and orientation of the given objects. This set of detected objects is used in visualization module, in order to make depiction of virtual 3D objects. Basing on the detected object identifier there is depicted a correspondent 3D object, using according transformation matrix. So that visual objects are depicted in accordance with markers, then, by manipulating with marked physical objects, the user can simultaneously manipulate with virtual objects

without a necessity for special input devices.

### IMPLEMENTATION DETAILS

The mixed reality simulation program prototype SharpSim is developed in the C#/.NET 2.0 programming environment, including the following packages: SharpDEVs, Sharp3D, SharpODE, ShARpToolkit (Figure 6). Although there exist many software libraries for visualization and VR/AR purposes (VRJuggler, OpenSceneGraph etc.), C#/.NET software system was chosen for development of such software prototype in order to check the possibilities of this progressive software development environment. In the future there is a plan to port this software to the open source C# programming environment Mono, allowing to work in Linux operating system environment. Creation of the 3D computer graphics objects is achieved with the Sharp3D package, which is built atop the industry standard OpenGL graphics library. Video

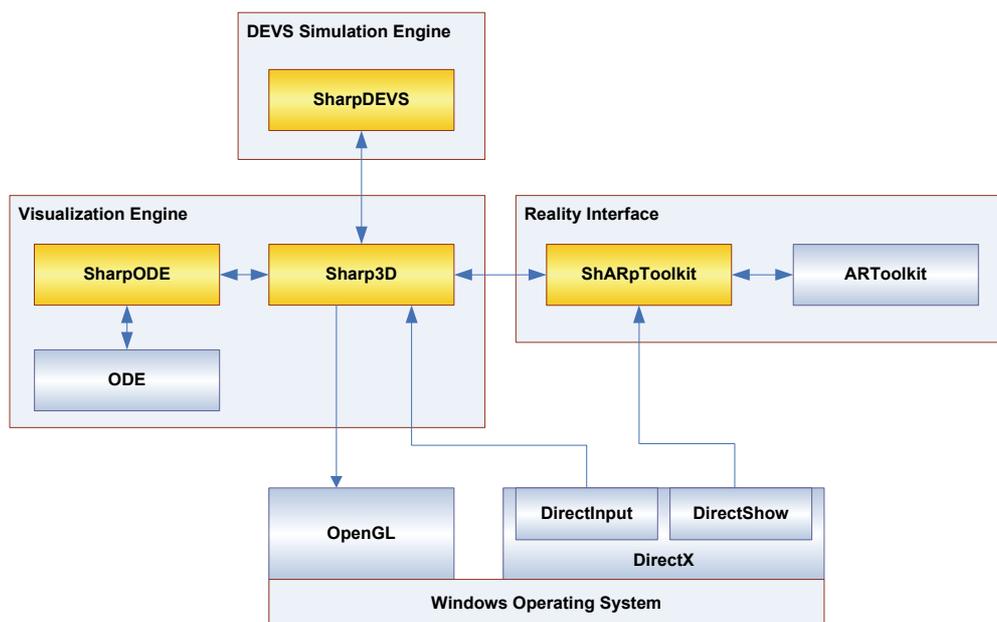


Figure 6: Architecture of SharpSim Simulation Environment

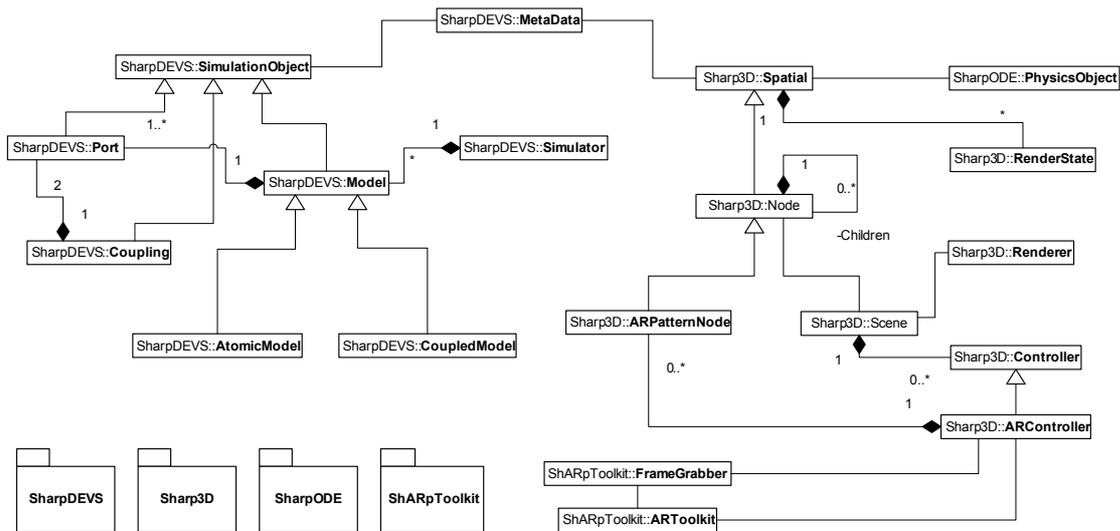


Figure 7: Generalized SharpSim Class Structure Diagram

data processing of captured frames is based on Microsoft DirectX 9.0 framework by using DirectShow capabilities. DirectInput system is used to react on the changes in the input of the mouse and the keyboard. In Figure 7 there is shown the general structure of SharpSim classes.

The given software prototype uses augmented reality library ARToolkit (Billinghurst et al. 2001) as a tool for real world object detection, using fiducial markers that are placed near the detectable objects. The data from video camera is passed to the computer, modified and displayed in the monitor in real-time by visual combining of captured video frames and 3D graphical objects. The obtained information about the detected objects and their position in 3D space is then used by scene graph module Sharp3D. ShARpToolkit package serves as ARToolkit interface for Sharp3D visualization module that by using of C#.NET resources allows implementing of ARToolkit functions and related data exchange. In ShARpToolkit module there is provided also video frame obtaining in real-time from the connected video camera, using Microsoft DirectShow technology. The obtained video frames are sent to ARToolkit library for marker detection and to Sharp3D, where by combining with virtual objects they are depicted as mixed reality scenes.

An open source Open Dynamics Engine (ODE) is used for simulating of physical object dynamics. SharpODE module serves as integration layer between ODE and Sharp3D by wrapping in C# methods the ODE function calls.

**PRACTICAL EXAMPLE**

For testing and verifying purposes of the proposed framework there is chosen a practical warehouse storage system, where the acquired video frames are used as background for simulation of warehouse working processes. But the fiducial markers that are placed in the

warehouse storage room allow mixing the given real images with virtual graphical objects.

In the Figure 8 an example model of warehouse storage system is shown. It contains source, queue, sink and two activity elements. The source element Source1 generates flow of incoming goods. Received goods are processed by queue, resource and operator elements.

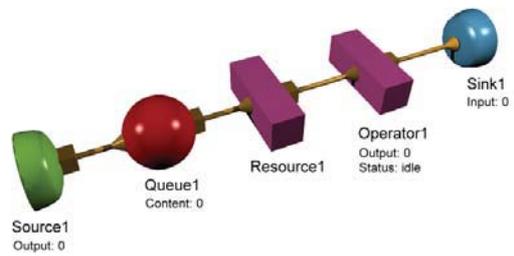


Figure 8: Simulation Model Diagram of Warehouse Storage System

In Figure 9 is shown a captured image of warehouse storage room containing two fiducial markers.



Figure 9: Captured Video Image of Warehouse Storage Room containing Fiducial Markers

In Figure 10 is illustrated a combined image, where virtual 3D objects are merged in to the real video captured image.



Figure 10: Mixed Reality Scene of Warehouse Storage Room

In Figure 11 is shown another image of the mixed reality scene representing working operations in warehouse storage room.



Figure 11: Example of the Combined Image with Virtual 3D Objects

## CONCLUSIONS AND FUTURE RESEARCH

Mixed reality is a popular topic for researchers, a lot of applications and hardware for Mixed Reality can be found. But there are not much studies of AR/MR usefulness in simulation modeling. This article offers a framework for developing of DEVS-based discrete event simulation models in virtual 3D environment allowing the user to use a mixed reality interface to build complex simulation models.

The long-term vision of this work is to develop a full-scale mixed reality simulation environment for discrete event simulation of logistics systems. Potential benefits of achieving this vision would include understanding and applying the relationship between simulation effectiveness, visualization quality and user interaction.

## REFERENCES

- Billinghamurst, M.; K. Hirkazu; I. Poupyrev. 2001. "The MagicBook: Moving Seamlessly between Reality and Virtuality". In *IEEE Computer Graphics and Applications*, (May-June). 2-4.
- Chawla, R.; A. Banerjee. 2001. "A Virtual Environment for Simulating Manufacturing Operations in 3D". In *Proceedings of the 33th conference on Winter Simulation* (Arlington, VA, Dec.9-12). 991-997.
- Fischer, M.; M. Grafe; C. Matyszcok; B. Mueck; and M. Schoo. 2003. "Virtual & Augmented Reality Support for Discrete Manufacturing System Simulation". In *Proceedings of the International Working Conference of the IFIP WG 5.7* (Karlsruhe, Germany, Oct.5-9). 170-178.
- Lektauers, A. 2003. "A Virtual 3D Environment for Logistics Modelling". In *Proceedings of the International Workshop on Harbour, Maritime and Multimodal Logistics Modelling & Simulation* (Riga, Latvia, Sept 18-20), 242-248.
- Milgram P. and F. Kishino. 1994. "A taxonomy of mixed reality visual displays". *IEICE Transactions on Information Systems*, No.12 (Dec), 95, 96.
- Odhabi, H.I.; R.J. Paul; R.D. Macredie. 1998. „Developing a Graphical User Interface for Discrete Event Simulation“. In *Proceedings of the 30<sup>th</sup> conference on Winter Simulation* (Washington, DC, Dec.13-16). 429-436
- Ohta Y. and H. Tamura 1999. *Mixed Reality – Merging Real and virtual Worlds*. Springer Verlag, Berlin, 95, 96.
- Rehn, G.D.; M. Lemessi; J.M. Vance; D.V. Dorozhkin. 2004. „Integrating Operations Simulation Results with an Immersive Virtual Reality Environment“. In *Proceedings of the 2004 Winter Simulation Conference* (Orlando, FL, Dec.4-7).
- Zeigler, B.P. 2003. „DEVS Today: Recent Advances in Discrete Event-Based Information Technology“. In *Proceedings of the 11<sup>th</sup> IEEE/ACM International Symposium On Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (Orlando, FL, Oct.12-15). IEEE, Orlando, 148-161.

## AUTHOR BIOGRAPHY



**ARNIS LEKTAUERS** received the M.Sc. in Computational Visualistics from Otto-von-Guericke University Magdeburg, Germany (1998). In 1999 he received a M.Sc. in Information Technology from the Department of Modeling and Simulation, Riga Technical University. He is a PhD student and part-time scientific assistant at the Department of Modeling and Simulation of the Riga Technical University. His e-mail address is : arnis@it1.rtu.lv

# saLib – A TOOLBOX AND VISUALISATION TOOL FOR IMAGE PROCESSING ON SPIRAL ARCHITECTURE

Stefan Bobe and Gerald Schaefer  
School of Computing and Informatics  
Nottingham Trent University, Nottingham, UK  
e-mail: Gerald.Schaefer@ntu.ac.uk

## KEYWORDS

Spiral architecture, hexagonal pixel, image processing toolbox, Matlab.

## ABSTRACT

The spiral architecture (SA) represents an alternative model of image representation which uses hexagonal rather than square picture elements. This data structure is not only closely related to biological vision systems but also offers many advantages compared to the normal rectangular representation.

This paper presents saLib, a toolbox and visualisation tool for image processing purposes on the spiral architecture under Matlab. saLib provides basic functionality such as storing, translating and rotating hexagon-based images, as well as methods to display them. Additionally, a visualisation tool is provided which can be used for the convenient operation on images or single addresses and the presentation of the results. saLib is available to fellow researchers for download.

## 1. INTRODUCTION

The common representation structure for digital images is that of a rectangular grid of square-shaped picture elements. While this has advantages it also has certain drawbacks such as that every pixel has to be identified by its row and column (i.e. two) co-ordinates or that pixels adjacent to one central pixel have different distances to the centre of the central pixel (the diagonal elements are further away than the horizontal or vertical adjacent pixels). To find a better representation, the idea of a hexagonal lattice has been introduced (Sheridan, 1996). In this model the basic picture element has the form of a hexagon. Adjacent hexagons all have the same distance from each other. Furthermore a special addressing algorithm, the so-called spiral architecture (SA), addresses every hexagon with only one co-ordinate and allows the use of a special algebra. This algebra can be used to operate on the address values independent from the actual co-ordinates of their position.

Based on this new data structure several other applications have been developed, providing basic tools for image processing like translation or rotation of image content or the transformation of the rectangular representation to the spiral architecture.

In this paper we present saLib, a Matlab toolbox which provides access to various image processing

functionality based on the spiral architecture. SA-based data structures are available as are functions for operations such as image translation and rotation. A graphical user interface is also provided as a visualisation tool to demonstrate the functionality provided.

The rest of the paper is organised as follows: Section 2 provides an overview of the theory behind spiral architecture image processing. Section 3 describes the library part of saLib while Section 4 covers the visualisation tool. Section 5 concludes the paper.

## 2. THEORY OF SPIRAL ARCHITECTURE

Sheridan (1996) introduced the spiral architecture (SA) as a concept for machine vision on a hexagonal lattice which not only provides a model closer to biological vision systems but also has other advantages over the conventional square pixel-based image representation. The basic picture element in SA has the form of a hexagon. It follows then that each element has six direct neighbouring cells (one on each side) and that these are exactly the same distance away (in contrast to the 8-neighbourhood of a square pixel). While the basic concept of hexagonal pixels was not new, Sheridan's work introduced a special addressing algorithm which identifies each hexagon of the structure with a unique one-dimensional address in base 7. The addressing algorithm, along with special addition and multiplication algorithms, provides the spiral architecture with the algebraic feature of a Euclidean ring. Several other algorithms based on this new data structure have been introduced since.

### 2.1. Spiral Counting

The addresses in SA are in base 7 and arranged in a spiralling way (see Figure 1 for the first 49 elements on the SA). Spiral counting (Sheridan, 1996) can be used to walk along the addresses in spiral architecture consecutively. The algorithm starts at the hexagon with address 0 and goes from this address to the address 1, called the 'key' as it defines the rest of the algorithm. Addresses 2 to 6 are arranged in 60° steps clockwise in the same distance as address 1 around the starting address. The next address, 10, is then found by going from 6 to 1 and twice the distance further in the same direction. Addresses 20 to 60 are again arranged in 60° steps clockwise around address 0 with the same distance as 10. At each of these addresses, addresses 11 to 16, 21 to 26 etc are arranged in the same distances

and directions as 1 to 6 around 0. Address 100 can be found in the same way as 10, this time using addresses 60 and 10 instead of 6 and 1, etc.

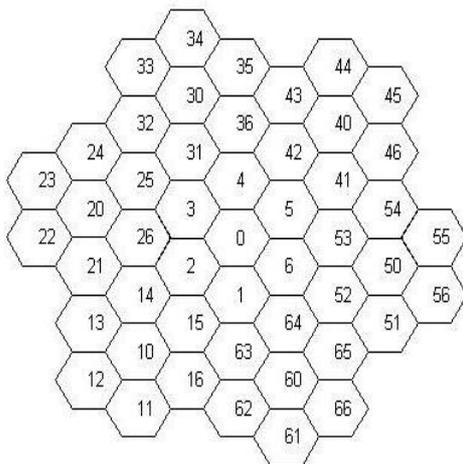
It was shown (Sheridan, 1996) that spiral counting can be generalised by either a variable starting address or a variable key. These two cases define spiral addition and spiral multiplication respectively.

## 2.2 Spiral Addition

Spiral addition (Sheridan and Hintz, 1999) of two spiral addresses  $x$  and  $y$  is equivalent to spiral counting in the key of 1 starting at address  $x$  for  $y$  addresses. Sheridan also suggested a 'carry rule' instead of spiral counting, which allows algebraic operations on the addresses independent from their co-ordinate representation (and is also computational more efficient). For the addition, the two addresses are decomposed to their digits and each pair of digits is added separately, using a special table for spiral addition (Sheridan, Hintz and Alexander, 2000). For results with a 2-digit length the carry rule spiral adds the higher decimal power to the next higher decimal power of one of the values.

A special form of spiral addition is the modulus of it, which uses the normal modulus function with a decimal power as modulus.

The use of modulus spiral addition on the whole spiral architecture results in a translation operation which shifts the contents of the image. Hexagons that would be shifted out of the normal range are wrapped around the whole spiral structure.



Figures 1: The first 49 elements on the Spiral Architecture

## 2.3 Spiral Multiplication

Spiral multiplication (Sheridan and Hintz, 1999) uses a variable key for spiral counting with the constant starting address 0. Spiral multiplication of  $x$  and  $y$  is equivalent to spiral counting of  $y$  addresses in the key

of  $x$ . Again the two values can be decomposed to their digits, but in this case every digit of the first value is spiral multiplied with every digit of the second value. The spiral multiplication uses again a special multiplication table for each pair of digits (Sheridan, Hintz and Alexander, 2000). Decimal powers are multiplied in normal fashion. Afterwards the results for each digit of the first address are spiral added.

A special modulus function is also defined to reduce the result of spiral multiplication. If the address is not a multiple of 10 then the normal modulus is used. In case the first address is a multiple of 10 the result after spiral multiplication  $p$  is reduced by  $[(p + (p/\text{modulus})) \bmod \text{modulus}]$ .

Modulus spiral multiplication can also be applied to the whole spiral structure and leads for a multiplication of address 1 to 6 to a rotation of the image by a multiple of  $60^\circ$ . For other addresses the modulus spiral multiplication produces several rotated copies of the original image or a new distribution of the old hexagons throughout the structure.

## 2.4. Virtual Spiral Architecture

(Wu, He and Hintz, 2004) introduced an algorithm to translate the rectangular to the hexagonal image structure allowing conventional images to be converted to SA images. The two structures are overlaid and the pixel values of each hexagonal pixel evaluated by interpolating between the pixel values of the underlying square cells. The interpolation is performed by splitting the original pixels into several smaller elements with the same intensity. The values and the number of the underlying points determine the value of each hexagon.

## 2.5. Log Space Based Transformations

Certain special addresses in the spiral architecture can be used to describe every other address (Sheridan, Hintz and Alexander, 2000). This can be accomplished by repeatedly applying modulus spiral multiplication and modulus spiral addition respectively with the same address. This repeated application cycles through the structure. The number of repeated operations to reach an address depends on the size of the structure and is used as a new address to describe the spiral address. This new address space was termed 'log space'. Repeated modulus spiral addition starts at address 0 and cycles through the whole structure whereas repeated modulus spiral multiplication only cycles through the addresses that are not a multiple of 10. Therefore the addresses that are multiples of 10 have to be split into their order of magnitude and the multiple of it. Not every address can be used to build these log spaces for every size of the structure however certain addresses such as address 1 for spiral addition and address 62 for spiral multiplication can be found. Application of log space transformations results in a vast gain of improvement in terms of computational complexity. Using log space operations such as image rotation can

be performed in a fraction of the time required by conventional methods.

## 2.6. Rotation Without Scaling

Spiral multiplication on the whole image structure has the restriction that only the spiral multiplication with addresses 1 to 6 results in a rotation (by a multiple of  $60^\circ$ ) without a scaling effect on the image. Wu, He and Hintz (2002) developed a special method which allows a rotation by any address or angle without scaling. For this purpose, the rotation angle is split into its multiple of  $60^\circ$  and the remainder. The multiple of  $60^\circ$  is used for spiral multiplication. The rotation by the fraction is then reached by rotating the centre of each hexagon of the whole structure by this angle around address 0. After this rotation each centre of a hexagon gets a new point and for each of these points the closest hexagon has to be found. This so called ‘pull back’ can map several points to one address whereas other hexagons do not get assigned. Applying this operation to the output image results in the hexagon of the input image.

## 3. saLib LIBRARY

This paper presents saLib, a Matlab toolbox and visualisation tool for image processing on the spiral architecture. Matlab (Mathworks, 2005) is a matrix-based mathematical programming language and widely used in fields such as engineering and the computer sciences including image processing and machine vision. saLib is intended as a toolbox to be used for image processing and machine vision research based on SA as well as a tool to visualise certain SA operations that can be used for educational purposes.

Table 1: Pre-defined data structures in saLib

Data structure	Description
<b>sa_ADDTABLE</b>	Matrix used for spiral addition .
<b>sa_MULTTABLE</b>	Matrix used for spiral multiplication.
<b>sa_HEXADDR</b>	Position of the hexagons (e.g. for the display with <code>sa_imview()</code> ).
<b>sa_LOGADD</b>	Matrices used for log space transformation to shift an image.
<b>sa_LOGMULT</b>	Matrices used for spiral multiplying an image by log space transformation.
<b>sa_OCTAVEPOL</b>	Contains polar co-ordinates of the first hexagon of each octave.

saLib includes the functionality of all the operations covered in Section 2 as well as special algorithms developed for the understanding of the spiral architecture and the presentation of the images under

Matlab. A full list of all functions provided and all data structures of saLib is provided in Tables 1 to 3.

The algorithms that are included in the toolbox can be divided into two groups: those that operate on spiral addresses and those that are applied to the whole spiral structure.

The remainder of this section explains the functionality of the library part of saLib while Section 4 describes the visualisation tool coming with saLib. saLib is available to fellow researchers for download from <http://vision.doc.ntu.ac.uk/>.

### 3.1 Displaying the Spiral Architecture

The `sa_imview()` function is used to display an image on the spiral architecture in order to be able to visualise the outcome of image processing operations on spiral images. Image can be either greyscale or full (RGB) colour images and all image data types provided in Matlab are supported. An example of an image displayed using `sa_imview()` is shown in Figure 2.

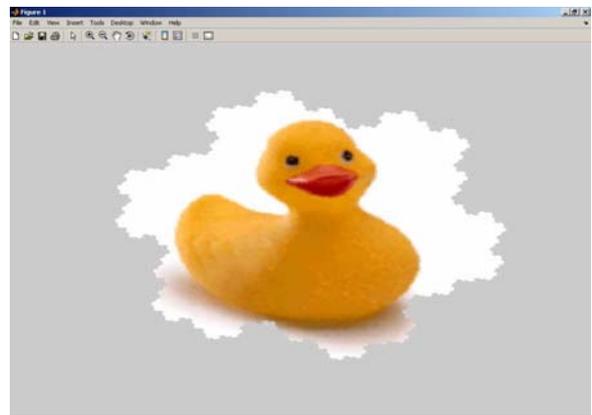


Figure 2: Display of an Image on Spiral Architecture

### 3.2. Converting between Rectangular and Spiral Images

Since at the moment there is little hardware support for imaging based on hexagonal pixels and as virtually all images in existence are based on the rectangular square-pixel structure functionality for converting rectangular images to spiral equivalents and back is needed. In saLib the `sa_rect2spiral()` and `sa_spiral2rect()` functions provide this functionality of converting from a rectangular image to its spiral counterpart and backwards respectively. For converting square-pixel based images to hexagon-based ones the virtual spiral architecture method described in Section 2.4 is used, i.e. the rectangular and the spiral structure are overlaid with each other, square pixels further divided and then averaged to provide the hexagonal pixel values. In order to obtain an appropriate size of the resulting hexagonal image the size parameter  $n$  of the SA is chosen so that the original rectangular images fits into its  $7^n$  hexagonal counterpart. Those hexagons outside the area of the rectangular image are filled with black.

The inverse operation `sa_spiral2rect()` uses the same method to evaluate the pixel values of the resulting rectangular image. Minimum and maximum co-ordinates on the spiral architecture are used to determine the number of rows and columns for the rectangular image. Pixels outside the hexagonal structure are set to white.

Figure 3 shows a sample image (Figure 3a) that is being transformed to a hexagonal representation (Figure 3b) and then back to a rectangular one (Figure 3c).



Figure 3a: Original rectangular image



Figure 3b: Image from 3a represented in spiral architecture



Figure 3c: Image from 3b transferred back to rectangular image

### 3.3. Operations on Spiral Addresses

Some of the functions in `saLib` operate directly on spiral addresses and it is these functions that the methods that operate on whole images rely on. `sa_spiralcount()` implements spiral counting as discussed in Section 2.1. `sa_getval()`, `sa_nhood()` and `sa_nhoodval()` can be used to return pixel values of certain addresses and their neighbouring hexagons. `sa_add()` and `sa_multiply()` provide methods for spiral addition and multiplication whereas `sa_modadd()` and `sa_modmultiply()` encapsulate modulus addition and multiplication. Finally, the functions `sa_hex2cart()` and `sa_cart2hex()` allow conversion between spiral addresses and Cartesian co-ordinates.

For further details on these functions the reader is referred to Table 2.

Table 2: Address-based functions in `saLib`

Function	Description
<code>sa_spiralcount()</code>	<b>spiral counting</b> Performs spiral counting for a given number of addresses starting at a given starting address and returns the resulting address.
<code>sa_getval()</code>	<b>get pixel values</b> Returns the values of a set of given addresses.
<code>sa_nhood()</code>	<b>neighbourhood addresses</b> Returns addresses of six surrounding hexagons for a given address.
<code>sa_nhoodval()</code>	<b>neighbourhood values</b> Returns pixel values of six surrounding hexagons for a given address.
<code>sa_add()</code>	<b>spiral addition</b> Performs spiral addition (see Section 2.2) of two given addresses using <code>sa_ADDTABLE</code> and returns the result.
<code>sa_multiply()</code>	<b>spiral multiplication</b> Performs spiral multiplication (see Section 2.3) of two given addresses using <code>sa_MULTTABLE</code> and returns the result.
<code>sa_modadd()</code>	<b>modulus spiral addition</b> Performs modulus spiral addition (see Section 2.2) of two given addresses and returns the result.
<code>sa_modmultiply()</code>	<b>modulus spiral multiplication</b> Performs modulus spiral multiplication (see Section 2.3) of two given addresses and returns the result.
<code>sa_hex2cart()</code>	<b>Cartesian co-ordinates from spiral address</b> Converts a given spiral address to Cartesian co-ordinates.
<code>sa_cart2hex()</code>	<b>spiral address from Cartesian co-ordinates</b> Converts a given set of Cartesian co-ordinates to their corresponding spiral address. 1, 2 or 3 addresses are returned, depending on whether the given point is within a hexagon, on the edge, or in the cusp.

### 3.4. Operations on Spiral Images

Based on the functions discussed above several operations can be performed on complete spiral images. `sa_imadd()` performs spiral addition on a spiral image which results in a shifting of the image contents. `sa_immune()` allows spiral multiplication on an image thus resulting in a rotation and scaling of the original image. Image rotation without simultaneous scaling can be achieved using the `sa_walkjump()` and `sa_walking()` functions. Functionality for log space operations as discussed in Section 2.5 is also provided; images can be translated and rotated/scaled using the `sa_imlogadd()` and `sa_imlogmultiply()` functions. More detailed information on all the functions operating on spiral images is given in Table 3.

Table 3 Image-based functions in saLib

<code>sa_rect2spiral()</code>	<b>rectangular image to spiral architecture</b> Converts a rectangular image to its representation in spiral architecture using the virtual SA algorithm (see Section 2.4).
<code>sa_spiral2rect()</code>	<b>spiral architecture to rectangular image</b> Converts an image on spiral architecture to its rectangular representation.
<code>sa_iminfo()</code>	<b>show image info</b> Returns class, size, and range of a given spiral image.
<code>sa_imview()</code>	<b>show spiral image</b> Displays a given spiral image on screen.
<code>sa_imadd()</code>	<b>spiral addition on image</b> Performs spiral addition on whole spiral image resulting in a translation of the image content (see Section 2.2).
<code>sa_immune()</code>	<b>spiral multiplication on image</b> Performs spiral multiplication on whole spiral image resulting in a rotation and scaling of the image content (see Section 2.3).
<code>sa_spiral2alog()</code>	<b>log space for spiral addition</b> Converts the spiral architecture to log space for spiral addition (see Section 2.5).
<code>sa_spiral2mlog()</code>	<b>log space for spiral multiplication</b> Converts the spiral architecture to log space for spiral multiplication (see Section 2.5).
<code>sa_imlogadd()</code>	<b>spiral addition on image using log space</b> Performs spiral addition on a whole spiral image using a log space from <code>sa_spiral2alog()</code> . Works faster than <code>sa_imadd()</code> .

<code>sa_imlogmultiply()</code>	<b>spiral multiplication on image using log space</b> Performs spiral multiplication on a whole spiral image using a log space obtained from <code>sa_spiral2mlog()</code> . Works faster than <code>sa_immune()</code> .
<code>sa_walkjump()</code>	<b>rotation of spiral image without scaling</b> Allows rotation of a spiral image without scaling according to Section 2.6.
<code>sa_walking()</code>	<b>rotation of spiral image without scaling</b> Performs rotation of a spiral image without scaling. Works similar to <code>sa_walkjump()</code> but does not produce gaps in the resulting output image.

### 4. saLib VISUALISATION TOOL

The toolbox also provides a graphical user interface which allows the user to visualise the results of the functions provided by the library. A screen shot of the visualisation tool is given in Figure 4. The tool is divided into two sections. The top section offers operations that are applied on a complete spiral image whereas the bottom section provides operations on addresses and co-ordinates of their centre.

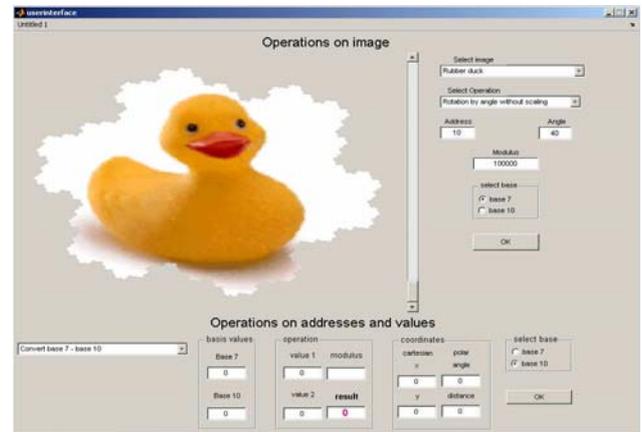


Figure 4: saLib visualisation tool

As indicated, the bottom part of the visualisation tool provides an interface for the functions discussed in Section 3.3. Users have the possibility to enter spiral addresses or co-ordinates and perform the following operations:

- Evaluation of Cartesian and polar co-ordinates for a given address.
- Definition of the nearest hexagon or hexagons for Cartesian co-ordinates.
- Spiral addition of two spiral addresses.
- Spiral multiplication of two spiral addresses.
- Modulus spiral addition of two addresses.

- Modulus spiral multiplication of two addresses.
- Conversion of base 7 to base 10 values and vice versa.

All spiral addresses and modulus can be entered either as base 7 or base 10 values. Results are given in numerical form and displayed. This part of the user interface is indented as an education tool for understanding the basic operations on the spiral architecture.

The top part of the tool allows the visualisation of the effect of spiral operations (discussed in Section 3.4) on whole images.

Operations that can be executed are:

- Shifting of an image by spiral addition (with or without log space transformation).
- Rotation and scaling of an image through spiral multiplication.
- Rotation without scaling of an image, either by a spiral address or by a specific angle, using the `sa_walkjump()` and `sa_walking()` functions.

Images of a sample session of the saLib visualisation tool are given in Figures 5a to 5c. Figure 5a shows the result of spiral addition on the Duck image shown in Figure 4. Figure 5b displays the outcome of spiral multiplication of the image resulting in a rotation and simultaneous scaling of the image. It can be observed that 7 downscaled and rotated instances of the original image are generated. Finally, Figure 5c gives an example of rotation without scaling.

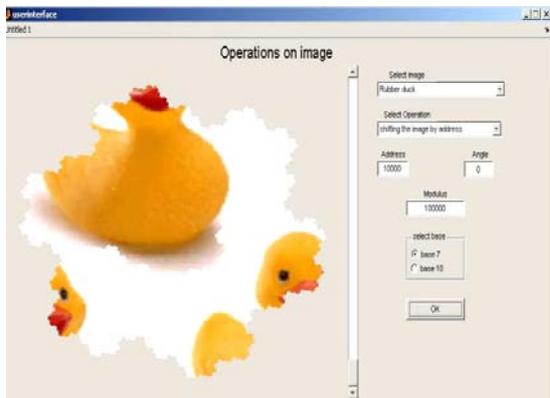


Figure 5a: Duck image after spiral addition of address  $10^4$

## 5. CONCLUSIONS

The spiral architecture offers geometrical as well as algebraic features which makes it a powerful new model for representing and processing images. We have introduced saLib, a Matlab toolbox for image processing on the spiral architecture. saLib offers a library of functions for converting and displaying spiral images as well as performing common operations such as translation and rotation of the image content. In addition, saLib provides a graphical tool which allows the visualisation of the functionality provided and which can hence be also employed as an educational

application. saLib is available to fellow researchers from <http://vision.doc.ntu.ac.uk/>.

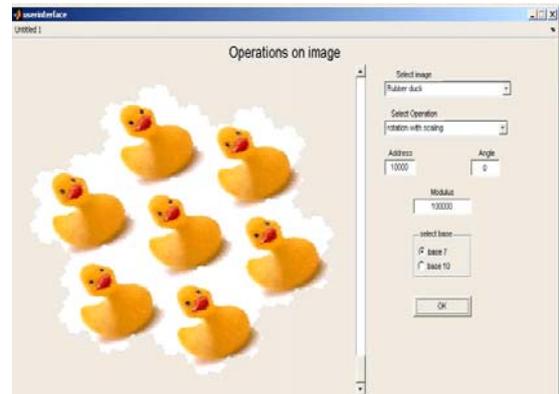


Figure 5b: Duck image spiral multiplied by address  $10^5$

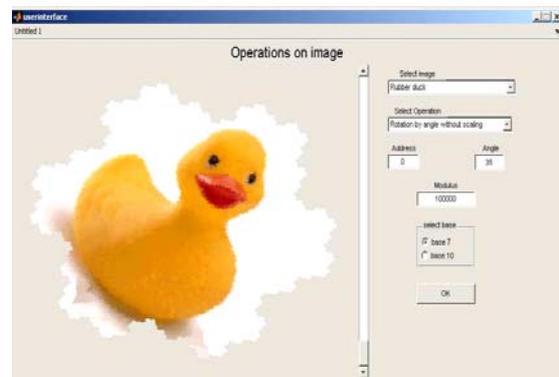


Figure 5c: Duck image rotated (without scaling) by  $35^\circ$

## REFERENCES

- Mathworks, 2005 Website: <http://www.mathworks.com>.
- Sheridan, P., 1996. "Spiral Architecture for Machine Vision", *Ph.D. thesis*, University of Technology, Sydney, Australia.
- Sheridan, P. and Hintz, T., 1999. "Primitive Image Transformations on a Hexagonal Lattice", *Technical Report*, Charles Stuart University, Bathurst, Australia.
- Sheridan, P., Hintz, T. and Alexander, D., 2000. "Pseudo-invariant image transformations on a hexagonal lattice", *Image and Vision Computing*, Vol. 18, pp. 907 – 917.
- Wu, Q., He, X. and Hintz, T., 2004. "Virtual Spiral Architecture", *Proc. Int. Conference on Parallel and Distributed Processing Techniques and Applications*, Vol. 1, pp 399 – 405
- Wu, Q. He, X. Hintz, T., 2002. "Image Rotation without scaling on Spiral Architecture", *Proc. Int. Conference in Central Europe on Computer Graphics, Visualization and Computer Vision*, pp. 515 – 520.

# DEFECT DETECTION USING A DISTRIBUTED BLACKBOARD ARCHITECTURE

Roger J. Tait, Gerald Schaefer, Adrian A. Hopgood and Lars Nolle  
School of Computing and Informatics  
Nottingham Trent University

Burton Street, Nottingham, UK, NG1 4BU

e-mail: roger.tait@students.ntu.ac.uk, {gerald.schaefer,adrian.hopgood,lars.nolle}@ntu.ac.uk

## KEYWORDS

Defect detection, blackboard system, parallel image processing, image registration.

## ABSTRACT

The framework of an automated vision system for the monitoring of quality control is presented. Inspection which is capable of detecting various forms of defects is achieved by combining distributed artificial intelligence and image processing. The blackboard architecture DARBS (Nolle and Wong 2001) manages the processing of image data via an area of shared memory where the current understanding of the problem evolves. Registration into a common co-ordinate system and segmentation of reference and sensed images is performed by intelligent agents which communicate with each other by means of the blackboard. Pixel-level fusion is performed on registered images in order to exploit complementary and redundant data, allowing identification of suspected defects. The difficulties of landmark extraction common to feature-based registration techniques have been replaced by an intensity-based algorithm. Addition or removal of specialised agents is simplified by the blackboard's modular nature.

## 1. INTRODUCTION

In manufacturing industries there is an increasing need for automated detection and characterisation of defects. The motivating factors for the adoption of an automated inspection approach include the reduction of expensive labour costs, reproducibility, and the matching of high-speed inspection with high-speed production. Identification of functional and cosmetic defects in finished products has been achieved using a number of techniques, a general overview of automated visual inspection is provided by (Newman and Jain 1995). The processing techniques described can be grouped into referential comparison (Lee 1978), non-referential modelling (Wen and Tao 1999) and hybrid inspection (Bayro-Corrochano 1993).

A variety of approaches in the printed-circuit-board (PCB) production environment are described by (Moganti et al. 1996). One such technique (Gokturk et al. 1999) uses the differentiating characteristics of a design, stored as a library, to compare and detect defects in a captured image. The detection of defects in

the production of moulded plastic products which employ a Fourier descriptor and differential gradient operator to classify imperfections as being either shape or surface anomalies is described by (Petkovic et al. 2002). Despite the advantages of speed, consistency and accuracy of an automated system over manual inspection, often the objective of these approaches is to ensure electrical connectivity, check form and classify quality using an inspection technique that consists of a rigidly defined set of steps.

The framework presented in this paper will monitor the quality of a sample and be capable of providing information for the removal of a defective product further along a production line. The blackboard architecture supports multiple agents each of which performs an independent task. By exploiting the opportunistic nature of the blackboard architecture the framework is freed from a predefined processing path and allows for the possibility of integrating different inspection techniques.

## 2. THE BLACKBOARD ARCHITECTURE

DARBS is a distributed blackboard system based on a client/server model. The server functions as the blackboard and client modules as knowledge sources (KSs). During execution the blackboard is used to host evolution of a solution to a problem. A KS in contrast is a structure in which rules and algorithms can be embodied. Reading from and writing to the blackboard is implemented as standard KS functionality. When updates to the solution of a problem are made, a broadcasting of messages is used to inform all relevant KSs. A KS's behaviour guides it on a course of action in response to broadcasted messages.

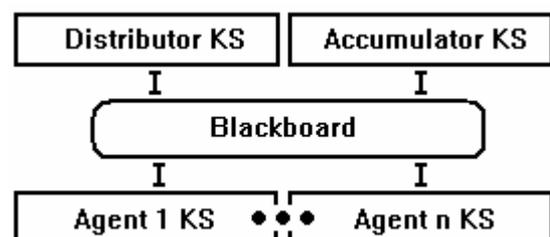


Figure 1: Blackboard components

Figure 1 shows that the initial implementation of the framework presented in this paper consists of a

Distributor KS, several Agent KSs and an Accumulator KS. The Distributor KS splits an image into segments before placing these on the blackboard. The Agent KSs takes image segments from the blackboard, registers them and processes them according to their implemented behaviour. A resulting output image is constructed from processed image segments by the Accumulator KS.

As the current state of a problem is stored on the blackboard, partitioning is used to balance communication and processing workloads. The partitioning of data aids the design of the framework by introducing structure to the blackboard. Due to the exhaustive search required, a drop in performance can be expected with a single partition implementation. Similar inefficiency can be expected when a KS requests information through management and processing of excess partitions. To combat these problems, the chosen partition implementation allows interaction between KSs in a logical and efficient manner.

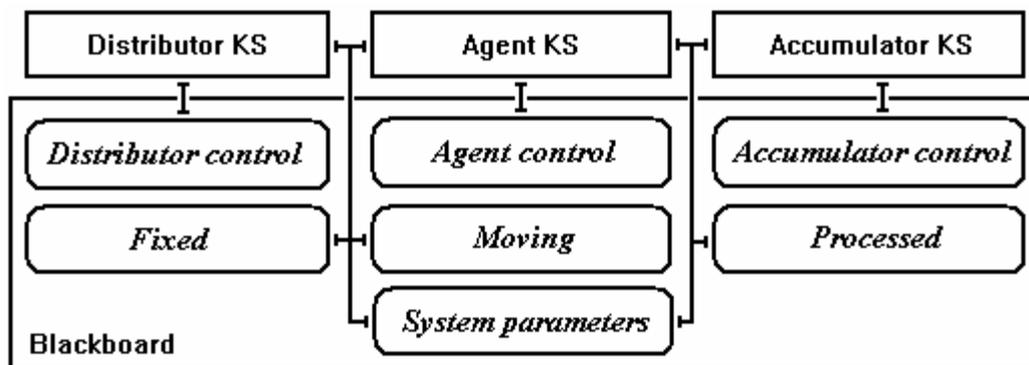


Figure 2: Blackboard partitions

Figure 2 shows that the blackboard is divided into the following partitions:

- A *Distributor control* partition controls division of an image into segments.
- The *Agent control* partition is used to manage processing of image segments.
- Reconstruction of the resulting image is accomplished within an *Accumulator control* partition.
- Variables used by all KSs are maintained by a *System parameters* partition.
- *Fixed*, *moving* and *processed* partitions hold image segments of their respective types.

Segment data is transmitted to and from the blackboard by the KSs. Transmission data is formatted into several parts. The first component corresponds to an image identification number, the next two values represent the image size. A minus symbol marks the end of attributes and the start of pixel data.

### 3. KS IMPLEMENTATION

By changing the structure of a KS's rules and through provision of additional functionality, the behaviour of a KS can be altered. The ITK toolkit (NLM 2004) was used to provide an Agent KS with registration and segmentation functionality through the embedding in rule files of shared library algorithms. Three simple image processing modules which interface ITK with the framework have been created; they encapsulate registration, segmentation and fusion functionality.

A multi-resolution registration algorithm which is capable of aligning images produced by a range of modalities forms the basis of the registration module. The algorithm can be tailored to a specific problem with dynamically selectable transform, interpolation, metric and optimiser components. Global and automatic thresholding functionality to separate an object from its background is provided through the segmentation module. Morphological operators for noise removal are also part of the segmentation module.

The fusion module performs an exclusive-or operation of image data. This XOR operation is used by Agent KSs to identify immediate differences between images and hence potential defects.

#### 3.1 The Distributor KS

Several tasks are performed by the Distributor KS. The first task involves the initialisation of data in the *System parameters* partition. The data includes the size of borders between adjacent segments and the number of segments an image is to be divided into. Registration parameters of transform, interpolation, metric and optimiser are also stored in the *System parameters* partition.

The second task involves selection of reference (fixed) and sensed (moving) images. In order to make sure all segments will be of the same size, both fixed and moving images are resized. If required, pixels from the bottom and right hand sides of an image are removed. Resizing of the images simplifies the production of a

resulting image. Both images are then divided into segments by the Distributor KS and sent to their respective partitions on the blackboard. Before transmission of data, an identification number is assigned to each segment.

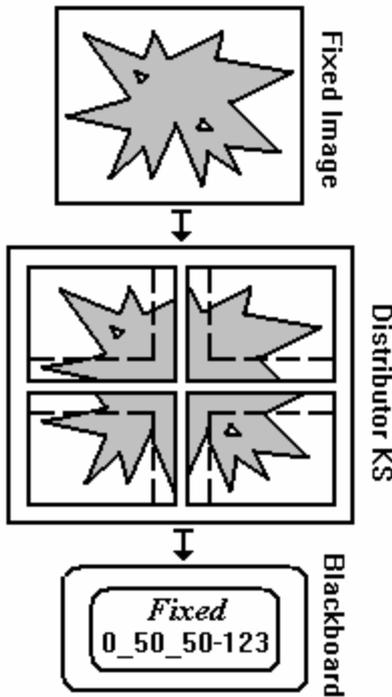


Figure 3: Distributor KS

Figure 3 shows the dividing of an image by the Distributor KS. Since the number of image segments is variable a balanced communication workload can be maintained. An increase in computational overhead however will occur when a large number of segments are allocated. Similarly, the objective of parallel processing will be defeated when the number of segments is small. Figure 3 also shows that only edges which face neighbouring segments have a border. The border is designed to remove non-pixel values which enter at the edges of a segment due to translation during registration. The border also removes the inconsistencies caused by convolution algorithms which require each pixel's neighbourhood.

### 3.2 The Agent KS

Once initialised an Agent KS enters a loop. A trigger mechanism ensures that all active Agent KSs commence processing at approximately the same time. Triggered Agent KSs then process segments in the following way:

- The current image segment waiting to be processed identification variable (CWP) is retrieved from the blackboard.
- Fixed and moving segments with identification numbers matching CWP are retrieved from the blackboard.
- CWP variable is incremented and returned to the blackboard.

- Fixed and moving segments together with registration parameters fetched from the blackboard are passed to the registration module.
- The process repeats itself until no segments are left for processing.

The registration parameters consist of transform, interpolation, metric and optimiser types. Both translation only and affine transform types are available to perform a spatial mapping between points in the fixed and moving segments. In order to evaluate pixel values at non-grid positions bi-linear and nearest neighbour interpolation algorithms have been implemented. Either normalised cross correlation or mutual information metrics can be used to measure the match between segments after they have been transformed. A gradient decent optimiser is used to iteratively search for the transform which best satisfies the chosen metric. According to modality, segmentation and fusion occurs once a segment has been registered.

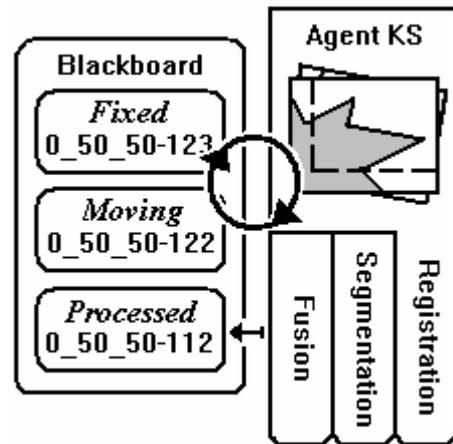


Figure 4: Agent KS

Figure 4 shows the registration of fixed and moving segments by an Agent KS. Once registered a segment is sent to the *Processed* partition on the blackboard. Processing segments in a first-come first-serve fashion allows a fixed number of Agent KSs to process a fixed number of segments. Alternatively a variable number of Agent KSs. This flexibility of processing means when an error is encountered a graceful degradation of the framework can occur. Importantly, memory usage of the blackboard is kept to a minimum by removing segments once they have been retrieved from the blackboard.

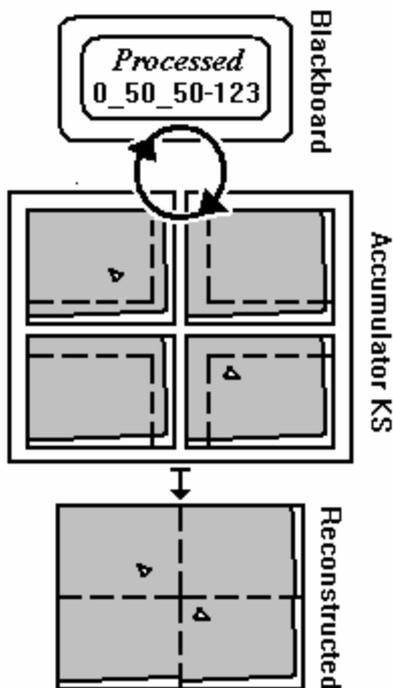
### 3.3 The Accumulator KS

The Accumulator KS is the simplest of all framework components. Once initialised the Accumulator KS stores in the *Accumulator control* partition a segments gathered variable (SG). The Accumulator KS then

enters a loop and process segments in the following way:

- Segments with identification numbers corresponding to SG are fetched from the blackboard.
- SG is incremented each time a segment is successfully retrieved.
- Retrieved segments are added to a list which is locally maintained by the Accumulator KS.
- The process repeats itself until all segments have been processed.

The total number of segments and border size variables are retrieved from the blackboard, as soon as all segments have been gathered. Each segment is then taken from the locally maintained list. Its borders are removed and the segment is inserted into the resulting image. Finally, the constructed resulting output image is automatically displayed by means of an image viewer.



**Figure 5: Accumulator KS**

Figure 5 shows how the Accumulator KS assembles a resulting image by removing segments from the *Processed* partition. The resulting image contains regions of localised rather than global registration. This is because the registration algorithm applies a transformation to each segment. A confidence level based on the magnitude of difference present in a segment can therefore be assigned. To maintain consistency the locally maintained segment list is cleared of all data whenever the Accumulator KS is started and after construction of a resulting image. Again, graceful degradation of the framework in the event of an error is possible by gathering segments in this way.

## 4. EXPERIMENTAL RESULTS

Image subtraction behaviour was added to the Agent KSs using the image processing modules in order to evaluate the initial detection performance of the framework and registration precision. PCB images of approximately 1000x500 pixels were chosen as test samples. The fixed image represents a sample with an acceptable and verified quality of manufacture. In contrast, the moving image is a sample containing a variety of defects. These include a spur and an open circuit both of which can be caused by dirt on a blank board or by air bubbles from electrolysis. Once selected, both images were divided by the Distributor KS into 4 segments and a 10-pixel wide border was assigned.

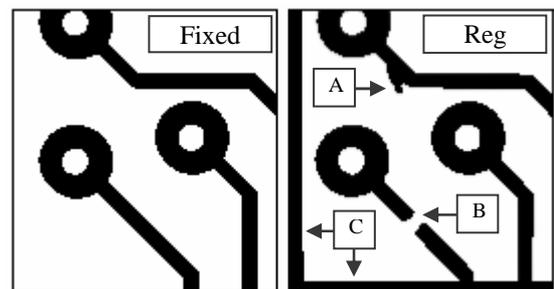
### 4.1 Registration

Spatial alignment into a common co-ordinate system of both fixed and moving segments was performed before the referential comparison could be made. The following components were chosen to perform registration:

- Affine transformation which allows for translation, rotation, and scaling of segments.
- Linear interpolation in order to allow pixel intensities to vary continuously between grid positions.
- Cross correlation metric in order to perform a pixel-wise association between the two segments.
- A regular step gradient descent optimiser because of its compatibility with the other components.

### 4.2 Segmentation

Once registered, threshold levels were automatically determined for fixed and registered segments. In order to segment the conductor from the insulation material, both segments were globally thresholded.



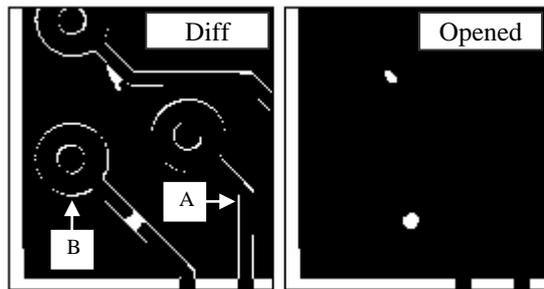
**Figure 6: Thresholded segments**

Figure 6 shows how both spur (A) and open circuit (B) defects have been segmented and are clearly visible in the registered segment. Additionally, translation and rotation caused by the registration process have introduced non-pixel locations which are visible at the bottom and left-hand sides (C) of the segment. All

extraneous pixels at the segment's borders are removed by the Accumulator KS.

### 4.3 Fusion

For the purposes of testing, subtraction was implemented as an exclusive-or operation between fixed and registered segments. Regardless of the sub-pixel accuracy achieved by the registration algorithm, differences appeared after subtraction as phantoms.



**Figure 7: Difference and opened segments**

Figure 7 shows that these phantoms appear along the contours of the conductive material (A,B). As the segments are of a relatively high resolution, phantoms which appear are generally smaller in size than potential defects. In order to achieve better segmentation, morphological opening was applied to the difference segment. Expansion and contraction of the segment caused by the opening operator resulted in the removal of all phantoms. To conserve small defects, the morphological structuring element consisted of a single pixel. The opening operator can be used to eliminate both large and small phantoms through changes in size and shape of the structuring element.

## 5. CONCLUSIONS

The initial implementation of an automated visual inspection framework has been described. As the choice of resolution determines the smallest size of detectable defect, high resolutions have been used to prevent loss of detail during the processing of a difference image. The difference image is produced by subtracting fixed and moving images after registration into a common co-ordinate system. Preliminary testing of the intensity based registration algorithm showed that to obtain reasonable processing speeds, images needed to be small or down sampled in size. Large high resolution images were found to take more than a minute to register. In contrast, smaller images required only a few seconds. To address the problems of resolution and speed the blackboard architecture provides a distributed approach. The blackboard allows for an image to be divided into a number of smaller segments which are distributed between multiple agents. Segments are processed concurrently so that high sensitivity and increased speeds can be maintained. It is intended that accurate detection of

smaller defects will be achieved through experimentation with greater resolutions.

## REFERENCES

- Bayro-Corrochano, E. 1993. "Review of automated visual inspection 1983 to 1993 Part II: Approaches to intelligent systems." *SPIE: Intelligent Robots and Computer Vision*, 159-172.
- Gokturk, S. Akarun, L. and Bozma, I. 1999. "Automated Inspection of Printed Circuit Boards Using a Novel Approach". *IEEE Workshop on Nonlinear Signal and Image Processing*.
- Lee, D.T. 1978. "A Computerized Automatic Inspection System for Complex Printed Thick Film Patterns." *SPIE: Applications of Electronic Imaging Systems*, 143, 172-177.
- Moganti, M. Ercal, F. Dagli, C.H. and Tsunekawa, S. 1996. "Automatic PCB Inspection Algorithms: A Survey." *Computer Vision and Image Understanding*, (63), 287-313.
- Newman, T.S. and Jain, A.K. 1995. "A Survey of Automated Visual Inspection." *Computer Vision and Image Understanding*. 61(2), 231-262.
- NLM Insight Segmentation and Registration Toolkit, <http://www.itk.org>, 2004.
- Nolle, L. Wong, K.C.P. and Hopgood, A.A. 2001. "DARBS: A Distributed Blackboard System." *Research and Development in Intelligent Systems*, XVIII, 161-170.
- Petkovic, T. Krapac, J. Loncaric, S. and Sercer, M. 2002. "Automated visual inspection of plastic products." *Proceedings of ERK 2002*, 283-286.
- Wen, Z. and Tao, Y. 1999. "Building a Rule-based Machine Vision System for Defect Inspection on Apple Sorting and Packing Lines." *Expert Systems with Applications*, (16), 307-313.

# BEHAVIOR VISUALIZATION OF AUTONOMOUS TRADING AGENTS

Tomoharu Nakashima, Hiroko Kitano, Hisao Ishibuchi

College of Engineering  
Osaka Prefecture University  
Gakuen-cho 1-1, Sakai, Osaka 599-8531, Japan  
{nakashi, kitano, hisaoi}@ie.osakafu-u.ac.jp

## KEYWORDS

Fuzzy rule, black-box analysis, futures trading

## ABSTRACT

In this paper, we visualize the behavior of a futures trading agent by using fuzzy if-then rules. The main aim of this paper is to graphically interpret how trading agents make decisions such as to buy or to sell a futures stock. Fuzzy if-then rules are used for this aim because their antecedent part specifies the features of time series. The procedure for visualizing the behavior of trading agent is first to train a trading agent that consists of a set of fuzzy if-then rules. By carefully examining, we select a small number of prominent fuzzy if-then rules that most represent the behavior of trading agents. Finally those selected fuzzy if-then rules are represented in a graphically understandable manner.

## INTRODUCTION

Fuzzy rule-based systems have been successfully applied to various problems such as control problems (Sugeno 1985, Lee 1990). The advantage of the fuzzy rule-based systems is its interpretability. Since fuzzy if-then rules in a fuzzy rule-based system are written using linguistic values, human users can linguistically understand the meaning of fuzzy if-then rules. The validity of fuzzy if-then rules can be measured using two well-known criteria in the field of data mining: confidence and support (Agrawal et al. 1996, Agrawal and Srikant 1994). The fuzzy version of these measures are also introduced in (Hong et al. 2001, Ishibuchi et al. 2001).

Although there are numerous researches on extracting patterns or rules from a large data base, the number of researches on how such extracted rules are effective for practical use in a particular domain is not large. The purpose of this paper is to examine the effectiveness of the extracted rules for supporting decision making of human users.

In this paper, we consider a virtual futures market as a problem domain. The virtual market allows a number of autonomous agents to take part in the futures market. Human beings are also allowed to trade in the futures stock index in the virtual market. Autonomous agents and human beings are required to determine whether they buy the futures stock index or sell, the limit price, and the quantity of the futures trade.

We have developed an autonomous agent that trades in futures stock index in a virtual market (Nakashima et al. 2002). An adaptive fuzzy rule-based system was used in the autonomous agent. The adaptive fuzzy rule-based system

consists of a number of fuzzy if-then rules that linguistically provide the decision making on the trading action (i.e., buy the futures index or sell) for different conditions. The evaluation (i.e., successful trade or not) of the trade action is performed after the new spot price and the new futures price are obtained. According to the evaluation, the agent adjusts the weights of fuzzy if-then rules in the adaptive fuzzy rule-based system. That is, we increase the weights of fuzzy if-then rules if the agent's decision making in the previous time step is successful. On the other hand, the weights of fuzzy if-then rules are decreased if the decision making is not successful. Since the weight update can be performed on-line, it is expected that the performance of the autonomous agent is gradually improved during the course of the trade. Thus the resultant fuzzy rule-based system after the enough number of trade can be viewed as a knowledge base for the virtual futures trade.

In this paper, we try to linguistically interpret the behavior of the autonomous agent by examining the weights of fuzzy if-then rules in the adaptive fuzzy rule-based system. We select a small number of fuzzy if-then rules that have a contrast between the weights associated with *Buy* the futures index and *Sell*. After the selected fuzzy if-then rules are graphically transformed into the trading knowledge base, the trading knowledge base is shown to a human trader who participates in the virtual futures trade. The human being can consult the trading knowledge throughout the futures trade. Statistical evidence shows that the trading knowledge base extracted from the adaptive fuzzy rule-based system improves the performance of human traders.

## A VIRTUAL FUTURES MARKET U-MART

Recently, virtual economic markets have attracted a great deal of attention for analyzing economic systems and developing autonomous agents. From the view point of the economics, the advantages of the virtual economic markets are as follows. First, one can analyze patterns of humans' trading behavior with respect to the trading action. Secondly, one can examine how to avoid the speculative action such as violent fluctuations of stocks. On the other hand, from the engineering point of view, we can examine the actual effectiveness of the use of learning methods, evolutionary methods, and multi-agent techniques in economic systems. The U-MART (Unreal Market as Artificial Research Test-bed) project is one of such virtual markets where multiple players including human beings can simultaneously trade in a futures stock index (Fig. 1).

In the U-MART, a machine (i.e., an autonomous agent) or a human being is called a U-MART client and is given market information such as time series data of spot prices and futures prices. Clients also have its own current information such as its position (i.e., a balancing amount of



Fig. 1 Snapshot of U-Mart

the futures index trade), remaining cash, and time to the final settlement. Based on the above information, each client has to make a decision on whether it buys or sells the futures index, the limit price of the futures trading, and the quantity of the futures trade. Thus, a client in U-MART can be viewed as an input-output system  $A$  as follows:

$$A(\mathbf{S}, \mathbf{F}, Pos, Cash, t) = (BS, P, Q), \quad (1)$$

where  $\mathbf{S}$  and  $\mathbf{F}$  is the time series of the spot prices and the futures prices (called the U-MART prices), respectively,  $Pos$  is the position of a client,  $Cash$  is the remaining cash,  $t$  is the remaining time to the final settlement,  $BS$  represents the client's trading decision on the futures stock index,  $P$  is the limit price, and  $Q$  is the quantity of the trade. Each U-MART client interacts with the U-MART server for trading in a futures stock index through the TCP/IP protocol. Among the input variables to the U-MART client, the two time series  $\mathbf{S}$  and  $\mathbf{F}$  are externally provided by the U-MART server and the other input variables are internally held by the U-MART client. In Fig. 2, we show a general view of the trade between a U-MART client and the U-MART server.

The U-MART server determines the futures index price by a method called *Itayose*. In *Itayose*, the U-MART server first collects an order from each U-MART client such as buy or sell of the futures index, the limit price, and the quantity of the trade. Then it compares the buy orders with the sell orders. The futures price is determined at the point where the price and quantity of buy orders are matched by those of sell orders.

The goal of the U-MART clients is to maximize the profit caused by the difference between the selling futures prices and the buying futures prices.

## ADAPTIVE FUZZY RULE-BASED SYSTEM

We have already developed a learning U-MART client for the virtual futures market (Nakashima et al. 2002). In (Nakashima et al. 2002), the U-MART client maintains an adaptive fuzzy rule-based system that determines whether the client should buy or sell a futures stock index. The weights of fuzzy if-then rules correspond to the support for buying or selling the futures stock index. During the course of the futures trades, weights of fuzzy if-then rules are updated on-line according to the evaluation of the trade at the previous time step. The trade is evaluated at each time

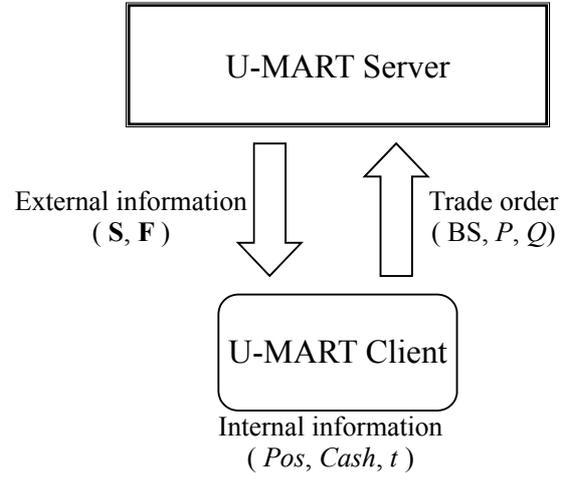


Fig. 2 The general view of the trade in U-MART

step by the high-and-low relation between the spot prices and the futures prices (i.e., U-MART prices). The following subsections describe the adaptive fuzzy rule-based system used for the autonomous agent in detail.

## Problem Formulation and Fuzzy If-Then Rules

In this subsection, we explain the fuzzy rule-based system that was applied to the learning U-MART client in (Nakashima et al. 2002). In the futures market, time series data of both the spot and the futures prices are available to the adaptive fuzzy rule-based system. Let us assume that  $n$  pieces of information are used by the agent for determining whether to buy or sell the futures stock index. In this case, the problem of the futures market for our futures trade agent can be viewed as a two class pattern classification problem with  $n$ -dimensional inputs. A fuzzy rule-based system is applied to this  $n$ -dimensional two-class pattern classification problem. The adaptive fuzzy rule-based system in the learning U-MART client consists of fuzzy if-then rules of the following type:

$$R_j: \text{ If } x_1 \text{ is } A_{j1} \text{ and } \dots \text{ and } x_n \text{ is } A_{jn} \\ \text{ then Buy with } b_j \text{ and Sell with } s_j, \quad j = 1, \dots, N, \quad (2)$$

where  $R_j$  is a label of  $j$ -th fuzzy if-then rule,  $\mathbf{x} = (x_1, \dots, x_n)$  is an input vector to the fuzzy rule-based system,  $A_{j1}, \dots, A_{jn}$  are antecedent fuzzy sets, and  $b_j$  and  $s_j$  are real values of the fuzzy if-then rule  $R_j$  corresponding to buying and selling the futures stock index, respectively.

In the implementation of the learning U-MART client in this paper, we use the difference between the spot price at the current time step and those at the three different time steps as three input values  $x_1, x_2, x_3$  to the fuzzy rule-based system (Fig. 3). That is, the learning client determines whether it buys or sells the futures stock index from an input vector  $\mathbf{x} = (x_1, x_2, x_3)$ . Thus, in this paper we deal with the decision making problem as an three-dimensional two-class pattern classification problem. The fuzzy if-then rules can be written as follows:

$$R_j: \text{ If } x_1 \text{ is } A_{j1} \text{ and } x_2 \text{ is } A_{j2} \text{ and } x_3 \text{ is } A_{j3} \\ \text{ then Buy with } b_j \text{ and Sell with } s_j, \quad j = 1, \dots, N, \quad (3)$$

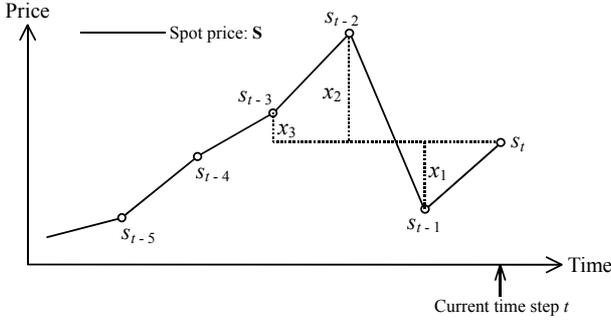


Fig. 3 Input variables in our fuzzy rule-based system

In Fig. 3,  $\mathbf{S} = (s_1, \dots, s_t)$  is the time series of spot prices from the beginning of the trade (i.e.,  $s_1$ ) until time step  $t$  (i.e.,  $s_t$ ) where  $s_k$  is a spot price at time step  $k$ . We use the following three pieces of information as input variables for the fuzzy rule-based system:

$$x_1 = s_t - s_{t-1}, \quad (4)$$

$$x_2 = s_t - s_{t-2}, \quad (5)$$

$$x_3 = s_t - s_{t-3}. \quad (6)$$

From the above explanation, we can see that the fuzzy rule-based system performs a mapping from a three dimensional state vector  $\mathbf{x} = (x_1, x_2, x_3)$  to a single binary value corresponding to either *Buy* or *Sell*.

### Fuzzy Inference and Decision Making

Let us consider that at a particular time we have already calculated three input variables  $x_1$ ,  $x_2$ , and  $x_3$  for the fuzzy rule-based system. In this subsection, we show how an agent makes a decision on whether it buys or sells the futures stock index. Assume that there are  $N$  fuzzy if-then rules in the fuzzy rule-based system. In this paper, we divide each axis of input variables into three fuzzy sets as in Fig. 4. In Fig. 4, N, Z, and P represent linguistic terms *negative*, *zero*, and *positive*, respectively. Since there are three input variables in our fuzzy rule-based system and three fuzzy sets for each input variable, the total number  $N$  of fuzzy if-then rules involved in the fuzzy rule-based system is  $N = 3^3 = 27$ .

Note that each fuzzy if-then rule has two weight values associated with buying and selling the futures index, respectively. After the calculation of three input values  $x_1$ ,

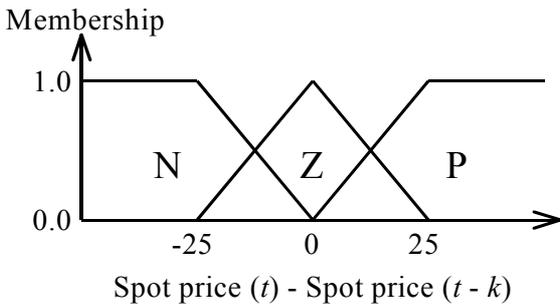


Fig. 4 Membership functions

$x_2$ , and  $x_3$ ,  $Q_{Buy}$  and  $Q_{Sell}$  are calculated using fuzzy inference as follows:

$$Q_{Buy} = \frac{\sum_{j=1}^N \mu_j(\mathbf{x}) \cdot b_j}{\sum_{j=1}^N \mu_j(\mathbf{x})}, \quad (7)$$

$$Q_{Sell} = \frac{\sum_{j=1}^N \mu_j(\mathbf{x}) \cdot s_j}{\sum_{j=1}^N \mu_j(\mathbf{x})}, \quad (8)$$

where  $\mathbf{x} = (x_1, x_2, x_3)$  is an input vector to the fuzzy rule-based system,  $\mu_j(\cdot)$  is the compatibility of the input vector  $\mathbf{x}$  with the fuzzy if-then rule  $R_j$ . The compatibility  $\mu_j(\cdot)$  of the input vector  $\mathbf{x}$  with the fuzzy if-then rule  $R_j$  is calculated by the following product operator:

$$\mu_j(\mathbf{x}) = \mu_{j1}(x_1) \cdot \mu_{j2}(x_2) \cdot \mu_{j3}(x_3), \quad (9)$$

where  $\mu_{ji}(\cdot)$  is the membership function of an antecedent fuzzy set  $A_{ji}$  in the  $j$ -th fuzzy if-then rule  $R_j$  (see Fig. 4).

After calculating  $Q_{Buy}$  and  $Q_{Sell}$ , the agent makes a decision on whether the agent buys or sells the futures index based on the following decision rule:

### [Decision Rule]

If  $Q_{Buy} > Q_{Sell}$ , the agent buys the futures index,

Else if  $Q_{Buy} < Q_{Sell}$ , the agent sells the futures index,

Otherwise, the agent's trade is the same as the decision at the previous time step.

Note that the autonomous agent does not make a decision for the first three time steps since there are not enough information to calculate an input vector for the first three time steps. That is, the agent only collects the information for the decision making for the first three time steps.

In this paper, we assume that the agent tries to optimize the decision making only on whether the agent buys or sells the futures index under various conditions of the time series data of spot prices. Thus fuzzy if-then rules in the adaptive fuzzy rule-based system have only two weights corresponding to *Buy* and *Sell*. There are, however, two more things to be determined in order to trade the futures index. One is the limit price and the other is the quantity of the trade.

The other two pieces of information such as the limit price and the quantity of the trade order are determined as follows. First, the limit price  $P$  in (1) is determined as follows:

$$P = \begin{cases} s_m - 5, & \text{if decision is Buy,} \\ s_m + 5, & \text{otherwise.} \end{cases} \quad (10)$$

That is, the limit price is deterministically decided based on the spot price at the current time step. On the other hand, the quantity  $Q$  of the trade in (1) is specified as  $Q = 200$ . That is, we don't adaptively determine the quantity of the trade according to any information but fixed to a prespecified value.

### On-Line Learning of Fuzzy If-Then Rules

In this subsection, we show how weights of fuzzy if-then rules in our fuzzy rule-based system are adjusted so that the agent can maximize the profit through the trading.

Let us assume that the agent has already made a decision on whether the agent buys or sells the futures index. At the next time step, the agent is given another information on the time series data of spot prices (i.e.,  $s_{m+1}$ ) and the futures prices (i.e.,  $f_{m+1}$ ). We evaluate the agent's trade decision (*Buy* or *Sell*) according to the high-and-low relation between  $s_{m+1}$  and  $f_{m+1}$  as follows:

#### [Evaluation Criterion]

If  $Q_{Buy} < Q_{Sell}$  and  $s_{m+1} > f_{m+1}$ , then the decision is evaluated as successful,

Else If  $Q_{Buy} < Q_{Sell}$  and  $s_{m+1} < f_{m+1}$ , then the decision is evaluated as successful,

Otherwise the decision is evaluated as unsuccessful.

That is, we evaluate the decision making based on the absolute price difference between the spot price and the futures price at the following time step. If the agent's decision is *Buy*, the evaluation for the decision is successful only if the spot price is higher than the futures price. On the other hand, if *Sell* is chosen, the evaluation for the decision is successful only if the spot price is lower than the futures prices. This criterion is derived from the observation that the spot price and the futures price must be coincide at the final settlement (see Fig. 5). This evaluation is used for updating the weights of fuzzy if-then rules.

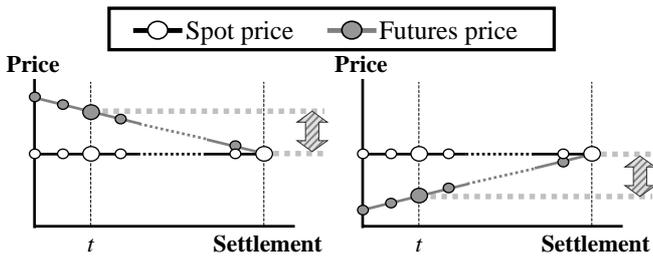


Fig. 5 Final settlement.

The main idea of the weight update is that the weights of the fuzzy if-then rules that contribute to the successful decision making are increased while we decrease the weights of those fuzzy if-then rules that are responsible for unsuccessful decision making. Thus the update rule of the weights is described as follows:

$$b_j^{new} = \begin{cases} b_j^{old} + \alpha \cdot (1 - b_j^{old}) \cdot \mu_j(\mathbf{x}), & \text{if successful,} \\ b_j^{old} - \alpha \cdot b_j^{old} \cdot \mu_j(\mathbf{x}), & \text{otherwise,} \end{cases} \quad (11)$$

$$s_j^{new} = \begin{cases} s_j^{old} + \alpha \cdot (1 - s_j^{old}) \cdot \mu_j(\mathbf{x}), & \text{if successful,} \\ s_j^{old} - \alpha \cdot s_j^{old} \cdot \mu_j(\mathbf{x}), & \text{otherwise,} \end{cases} \quad (12)$$

where  $\alpha$  is a positive learning rate and  $b_j$  and  $s_j$  are weight values of the  $j$ -th fuzzy if-then rule  $R_j$  that are associated with buying and selling the futures stock index, respectively. Note that only the weights corresponding to the selected action are updated by the above equations. We do not update the weights corresponding to the action that is not selected. For example, when we select to *Buy* the futures stock index, the weights  $b_j$ ,  $j = 1, 2, \dots, N$ , are updated and we do not modify the weight  $s_j$  that are associated with the action *Sell*.

To summarize, the procedure of our learning client for the futures trade is described as follows:

#### [Procedure of on-line learning for futures trading]

- Step 1: *Initialization*. Set initial weights of the fuzzy if-then rules to either some prespecified values or random values.
- Step 2: *Decision making*. Using the time series of spot prices, calculate the value corresponding to *Buy* or *Sell* as in (7) and (8). Make a decision of *Buy* or *Sell* according to the decision rule described in the previous subsection. The limit price and the quantity of the trade are also determined.
- Step 3: *Evaluation*. Given the futures price and the spot price at the following time step, evaluate the decision making of *Buy* or *Sell* as successful or not according to the evaluation criterion described in this subsection.
- Step 4: *Weight update*. Update the weights of fuzzy if-then rules involved in the fuzzy rule-based system. Note that only those weights corresponding to the decision making (i.e., *Buy* or *Sell*) at the current time step are updated.

These above steps are iterated until the contract month (i.e., the final settlement) is reached in the futures trading.

## VISUALIZATION

### Procedure of Knowledge Extraction

The learning U-MART client in the last section can be used as a knowledge acquisition tool since the adaptive fuzzy rule-based system in the U-MART client can be seen as a knowledge base for the virtual futures trade. In this section, we examine such possibility through laboratory experiments. The knowledge extraction procedure consists of two phases: tuning of fuzzy if-then rules in the adaptive fuzzy rule-based system and selecting a small number of fuzzy if-then rules with a large contrast between consequent weights. In the following subsections, each phase of the knowledge extraction is explained.

## Tuning and Interpreting the Fuzzy Rule-Based System

First, the learning U-MART client with the adaptive fuzzy rule-based system is iteratively applied to the virtual futures market. Since the learning client needs a number of iterations for learning the weights of fuzzy if-then rules, we repeated the virtual futures trade several times. After the futures trade, it is expected that those fuzzy if-then rules that are related to the critical input states have a contrast between weights for *Buy* and *Sell*. For example, the weight of *Buy* is larger than *Sell* for a fuzzy if-then rule if the U-MART client has made a number of successful decision making of *Buy* in a situation compatible to the antecedent part of the fuzzy if-then rule. Such a fuzzy if-then rule is likely to suggest *Buy* in the corresponding situation. Another example is that if the U-MART client has made a number of unsuccessful decision making in a situation compatible to the antecedent part of a fuzzy if-then rule, the weight corresponding to the decision making becomes smaller than the weight corresponding to the other decision making. In this case, the suggestion by such a fuzzy if-then rules is not to perform the trade action (either *Buy* or *Sell*) with the smaller weight.

### Selecting a Small Number of Fuzzy If-Then Rules

We examined the weights of each fuzzy if-then rule to select a small number of fuzzy if-then rules with a strong contrast between weights for *Buy* and *Sell*. From the total number of 27 ( $=3^3$ ) fuzzy if-then rules, we manually selected five such fuzzy if-then rules. Table 1 shows the selected fuzzy if-then rules with a strong contrast in the weights.

Table 1 Selected Fuzzy If-Then Rules

No	$x_3$	$x_2$	$x_1$	$q_{j1}$	$q_{j2}$
1	N	N	N	0.598	0.086
2	N	Z	P	0.778	0.318
3	N	P	N	0.546	0.870
4	P	Z	Z	0.573	0.800
5	P	P	P	0.924	0.141

Note that these fuzzy if-then rules were selected manually and subjectively according to the difference in the weights of fuzzy if-then rules. Although it is possible to systematically select a small number of fuzzy if-then rules using some statistical technique, it is beyond our scope of this paper. It will be investigated in our future research.

### Experiments with Human Users

In this subsection, we show the experimental results where human users are provided with a small number of the selected fuzzy if-then rules when they participate in the virtual futures trade. In order to make the selected fuzzy if-then rule more understandable, we visualized the selected fuzzy if-then rules as shown in Fig. 6. In Fig. 6, graphs (a)-

(e) correspond to the selected rules No. 1-5 in Table 1, respectively. The visualization is done such that the antecedent linguistic values (N, Z, and P) are interpreted as the relative position between the present spot price  $s_t$  and the previous spot prices  $s_{t-3}$ ,  $s_{t-2}$ , and  $s_{t-1}$ , and the recommended action is determined according to the extreme value of the consequent value (0 or 1) for each selected fuzzy if-then rule (Note that the value 0 means that the corresponding action is not recommended and it is recommended when the value is 1).

In our experiments, six human users separately participated in the virtual futures trade twice. One experiment is done with the presentation of the selected fuzzy if-then rules, and the other without the presentation of the selected fuzzy if-then rules. We performed this experiment for six different human users. For three human users, the first experiment was done with the presentation of the selected fuzzy if-then rules and the second experiment without the presentation. On the other hand, the experiments for the other three human users were done with the selected fuzzy if-then rules presented in the first experiment and without the presentation in the second experiment. This is because we need to minimize the effect of the ordering condition in the experiments. That is, the bias of presenting the selected fuzzy if-then rules in the first experiment for the first three human users is offset by the bias of presenting them in the second experiment for the other three.

We show the experimental results in Table 2. Table 2 shows the remaining assets after the final settlement.

Table 2 Experimental results

Human ID	Without guide	With guide
A	258,687,000	1,449,355,000
B	878,280,000	2,093,095,000
C	961,393,000	2,335,901,000
D	*	1,675,616,000
E	926,983,000	1,220,221,000
F	740,654,000	508,363,000

\* shows the human user went bankruptcy.

From Table 2, we can see that almost all the human users could performed better with the presentation of the selected fuzzy if-then rules than without them. Thus we can expect that our learning client could become a human decision support tool. In order to confirm this observation statistically, we perform the Wilcoxon's rank-sum test. The Wilcoxon's rank-sum test is a nonparametric test that is a sample  $t$ -test based solely on the order in which the observations from the two samples fall. In the Wilcoxon's test, we order the results of human users in the descending

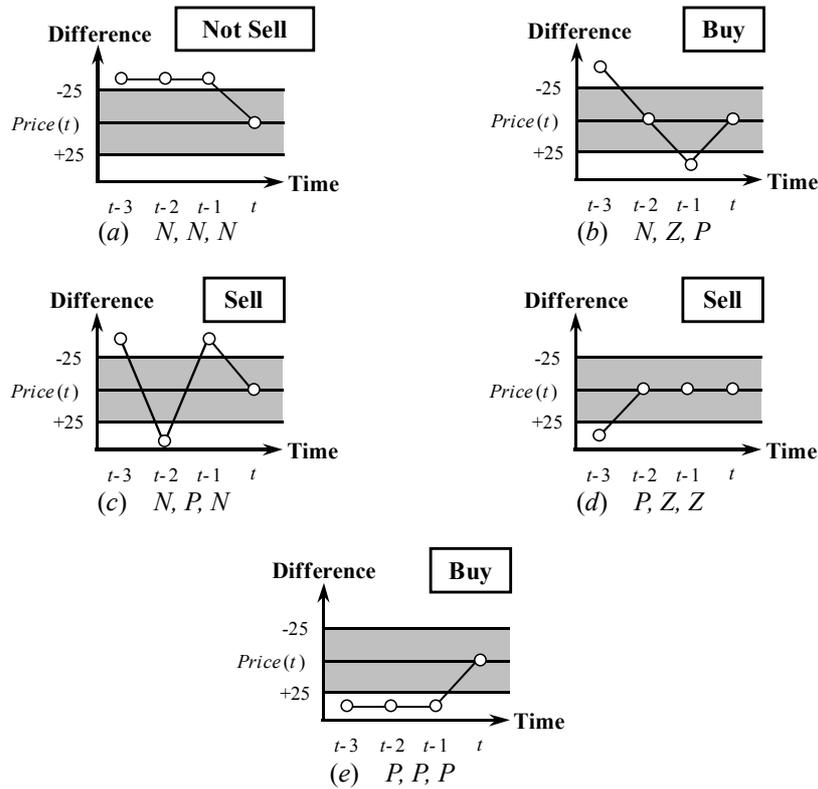


Fig. 6 Visualization of the selected fuzzy if-then rules

order of the final remaining assets in Table 2. The order is used as a rank for each human user. The sum of the rank is used as the statistic for one-sided test. In the test that compares the null hypothesis (there is no difference between the result with and without the presentation of the selected rules) against the alternative hypothesis (there is difference), the null hypothesis is rejected with a 0.05 level. Thus, we can statistically say that the human users can perform better with the help of the selected fuzzy if-then rules.

## CONCLUSIONS

In this paper, we presented how an autonomous agent in a virtual futures market is graphically analyzed. The autonomous trading agent has a learning mechanism during the course of the trade. A set of fuzzy if-then rules were used whose antecedent part represents the ups-and-downs of time series data. The consequent part of fuzzy if-then rules is the trading decision of the agent, i.e., *Buy* or *Sell*. First we applied a learning method to a virtual futures market in order to obtain the possible futures trading knowledge. Then we manually selected a small number of fuzzy if-then rules with a strong contrast in weights for decision making options. The selected fuzzy if-then rules were presented to human users after we visualized those fuzzy if-then rules in order for human users to easily understand them. Experimental results showed that there was a positive effect of presenting the selected fuzzy if-then rules. That is, human users could achieve better remaining

assets by using the extracted knowledge for the futures trade.

## REFERENCES

- M. Sugeno. 1985, "An Introductory Survey of Fuzzy Control", *Information Science*, Vol. 30, No. 1/2:59-83.
- C. C. Lee. 1990, "Fuzzy logic in control systems: fuzzy logic controller Part I and Part II", *IEEE Trans. Syst., Man, Cybern.*, Vol. 20: 404—435.
- R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, and A. I. Verkamo, 1996, "Fast discovery of association rules", *Advances in Knowledge Discovery & Data Mining*, 307-328, AAAI Press, Menlo Park, CA.
- R. Agrawal and R. Srikant, 1994, "Fast algorithms for mining association rules", *Proc. of 20th International Conference on Very Large Data Bases*, 487-499. Expanded version is available as IBM Research Report RJ9839, 1994.
- T.-P. Hong, C.-S. Kuo, and S.-C. Chi, 2001, "Trade-off between computation time and number of rules for fuzzy mining from quantitative data", *International Journal of Uncertainty Fuzziness and Knowledge-Based Systems*, vol. 9, no. 5: 587-604.
- H. Ishibuchi, T. Yamamoto, and T. Nakashima, 2001, "Fuzzy data mining: Effect of fuzzy discretization", *Proc. of 1st IEEE International Conference on Data Mining*, 241-248.
- T. Nakashima, T. Ariyama, and H. Ishibuchi, 2002, "On-line learning of a fuzzy system for a futures market", *2002 International Conference on Fuzzy Systems and Knowledge Discovery*, 54-58.

# APPARATUS AND COMPUTER X-RAY TOMOGRAPHY: VISUALIZATION OF INTRINSIC STRUCTURE, EVALUATION OF PERFORMANCE AND LIMITATIONS

Marina Chukalina  
Sergey Zaitsev  
Maxim Knyazev  
C. J. Vanegas  
Institute of Microelectronics  
Technology RAS  
142432 Chernogolovka  
Moscow District, Russia  
[marina@ipmt-hpm.ac.ru](mailto:marina@ipmt-hpm.ac.ru)  
[zaitsev@ipmt-hpm.ac.ru](mailto:zaitsev@ipmt-hpm.ac.ru)

Dmitry Nikolaev

Institute for Information  
Transmission Problems, RAS  
Bol'shoj Karetnyj lane, 19  
101447, Moscow, Russia  
[dimonstr@iitp.ru](mailto:dimonstr@iitp.ru)

Alexandre Simionovici

Laboratoire des Sciences  
de la Terre  
ENS, 46 allée d'Italie  
Lyon, 69007 Lyon, France  
[asimiono@ens-lyon.fr](mailto:asimiono@ens-lyon.fr)

## KEYWORDS

Confocal apparatus and computer tomography in medicine, image visualization, dose deposition.

## ABSTRACT

In this paper we compare confocal apparatus tomography and computer tomography for medical use. Two aspects are illustrated: the process of image visualization and the dose deposition. The latter aspect is a damage factor of the object under investigation; it is why the data collection time is evaluated. Two principal tomography schemes are described and the expressions connecting the collected signal value with the object's internal structure are given. Based on the expression, the process of image visualization is described and the dose deposition is calculated for both cases. The object's classification is introduced to choose the optimal experimental scheme for the object under study.

## INTRODUCTION

The use of X-ray microdiagnostic systems in biology and medicine has grown, covering such areas of investigations as the tomography of small animals (Asadchikov et al. 2004), the tomography of labeled cells (Schneider et al. 2001), the three-dimensional evaluation of biocompatible materials (Muller et al. 2001), the microtomography for the analysis of osteointegration around implants (Bernhardt et al. 2004) etc. Fluorescence microtomography combines fluorescence analysis with X-ray tomographic techniques and enables multielemental observation (Golosio et al. 2003).

The terms "apparatus" and "computer" describe two modes of the fluorescence collection during the experiment. In the first case a parallel (Takeda et al. 1995) or a confocal (Zaitsev et al. 2002) collimator is

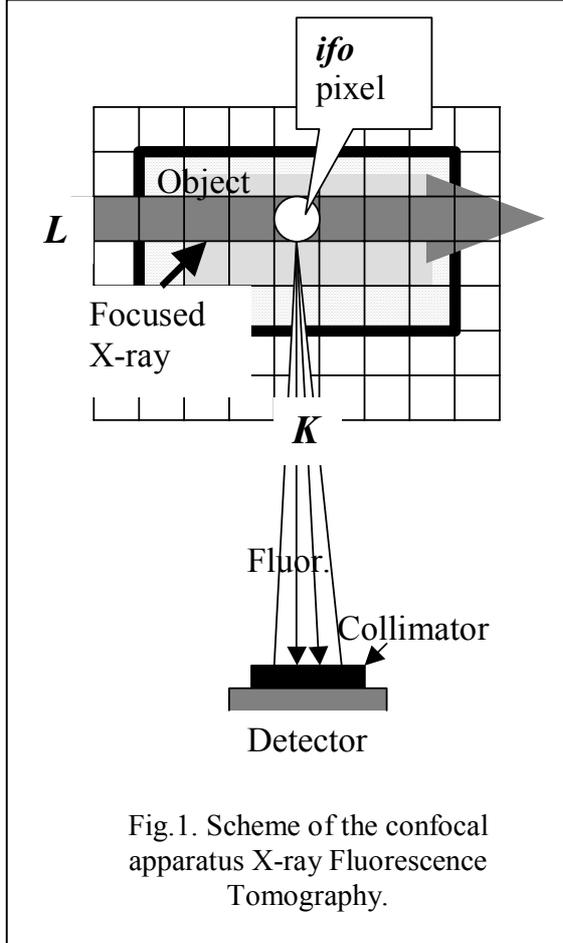
placed in front of the energy-dispersive detector window to decrease the detector solid angle and to localize the fluorescence generation zone from where the signal is collected. The registered signal is a function of the detector focal spot position. In the second case there is no space localization. The experimental set-up repeats the X-ray computer tomography parallel scheme but the fluorescence is registered in addition to the absorption signal. The registered signal is now a function of the X-ray microbeam position and the rotation angle (Simionovici et al. 1999). To compare the advantages and limitations of apparatus tomography and computer tomography we have analysed the image visualization process. The data manipulation procedures and the signal collection time (radiation dose) are evaluated for both cases.

## CONFOCAL APPARATUS TOMOGRAPHY WITH X-RAY MICROBEAM

The scheme for confocal apparatus X-ray fluorescence tomography is presented in Fig.1. An X-ray microfocus beam (Snigireva et al. 2003) illuminates a small volume of the sample. That part of the sample generates fluorescence which is collected by an energy-dispersive detector through the confocal collimator (Fig. 2) placed in front of the detector. The collimator (Fig.2) manufactured by Microelectronics Technology (Zaitsev et al. 2004) features a microfocus spot size. The intersection of the microbeam and the collimator focus localizes a zone from where the fluorescence is collected. The sample mounted on the sample-stage is translated in X-Y-Z directions with an applicable offset. Lateral resolution of the technique is determined by the scanning step, the X-ray beam size and the collimator focal spot size, appropriately. Usually the scanning step is equal to the collimator focus size and a

pixel has the same size (Fig.1). Below we consider pixels rather than voxels to simplify the considerations. For each "focal pixel" (*ifo* pixel on Fig.1) a fluorescence spectrum is registered. Each spectrum is mathematically processed to obtain a discrete set of values.

The spectrum is mathematically processed to obtain a discrete set of values, and the value given to this pixel is the number of one of the fluorescence lines. Let us say that  $M$  elements ( $M$  fluorescence lines) are measured and  $N^2-1$  sample translations are made during an experiment. The image size is  $N$  by  $N$  pixels. Each image corresponds to one element. But the image



is not the distribution of the element until an additional process including an attenuation correction has been done. To describe the correction needed to visualize the real image of the  $i$ th element distribution, the mathematical model of the signal  $S_{ifo}^i$  should be written

$$S_{ifo}^i = I_0 C_{ifo}^i \mu^i \Delta \eta^i \Phi_{ifo}^i \sum_{j=1}^J F_{ifo}^{ij} \quad (1)$$

Here  $I_0$  is the flux of the X-ray beam,  $C_{ifo}^i$  is the  $i$ th element concentration,  $\mu^i$  is the linear attenuation coefficient of the X-ray by  $i$ th element,  $\Delta$  is the linear size of the pixel,  $\eta^i$  is the fluorescence yield of  $i$ th element line,  $J$  is the number of collimator pinholes.

$$\Phi_{ifo}^i = \exp\left(-\sum_{k=0}^{K-1} \mu[ifo - K + k] \Delta\right) \quad (2)$$

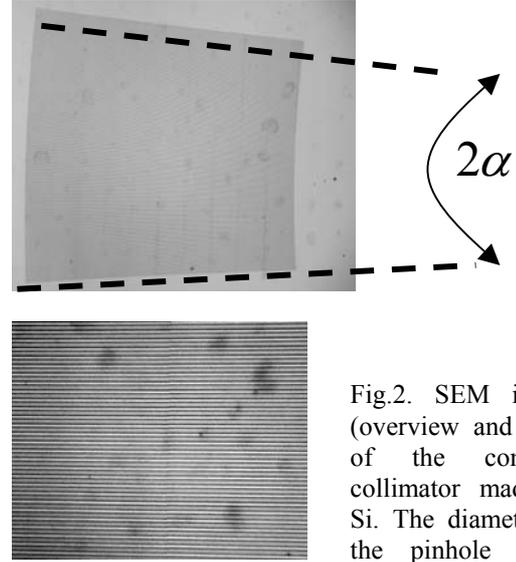


Fig.2. SEM image (overview and part) of the confocal collimator made in Si. The diameter of the pinhole is 2 microns, the number of pinholes is 200, the length of the pinhole is 1 mm, the focal distance is 5 mm.

Here  $\mu[ifo] = \sum_{i=1}^M C_{ifo}^i \mu^i$ .  $\Phi_{ifo}^i$  describes the X-ray attenuation of the X-ray by the sample along the X-ray's path (Fig.1). To calculate it for all pixels to the left of *ifo* pixel, the element concentrations should be known. This shows that the scanning procedure should be started from the left.

$$F_{ifo}^{ij} = \int_{\Omega_{pin}} d\Omega \exp\left(-\sum_{l=1}^{l-1} v^l [ifo - lN - \text{int}(\Delta l \alpha_j(\Omega))] \Delta\right) \quad (3)$$

Here  $j$  is the collimator pinhole number (Fig.2),  $v^i[ifo] = \sum_{j=1}^M C_{ifo}^i v_j^i$  is the linear attenuation coefficient of

the  $i$ th fluorescence line by the sample,  $\Omega_{pin}$  is the solid angle of the collimator pinhole (Fig.2),

$$\alpha_j = -\alpha + \frac{2\alpha}{N_{pin} - 1} (j - 1)$$

is the axis of the  $j$ th

pinhole position.  $F_{ifo}^{ij}$  describes the attenuation of the  $i$ th fluorescence line by the sample on the route to the detector (Fig.1). To calculate the corresponding values it is necessary to know the elements concentrations of the pixels laying on the fluorescence way. In Fig.1 it is seen that that the scanning procedure should be organized from bottom to top. The numbering of the pixels in Fig.1 corresponds to the scanning procedure. The first pixel is the bottom left pixel and the last pixel is the top right pixel.

Now it is possible to write the algorithm for the image visualization by confocal apparatus tomography Fig.1. To start with it is necessary to organize the links to the tables containing the required linear attenuation coefficients of the  $M$  elements for the X-ray and all fluorescence lines; then to start the tour over the pixels according to the scanning scheme; for each pixel measured spectrum to describe to the vector with  $M$  components; to calculate  $\Phi_{if_0}^i$  and  $F_{if_0}^{ij}$  for all collimator pinholes; to calculate  $C_{if_0}^i$  and to move to the next pixel. The procedure of the image visualization can be performed in-situ (pixel by pixel). It is very important because the data collection time (dose deposition) can be varied from pixel to pixel to achieve the required density resolution.

### CONFOCAL APPARATUS TOMOGRAPHY WITHOUT X-RAY MICROBEAM

Despite very promising prospects, to organize the intersection of the X-ray microbeam and the collimator microfocus is a very difficult task. The first experiment without X-ray microbeam was carried out on February 2005 at the ESRF (Zaitsev et al. 2005). The scheme of the data collection is presented in Fig. 3.

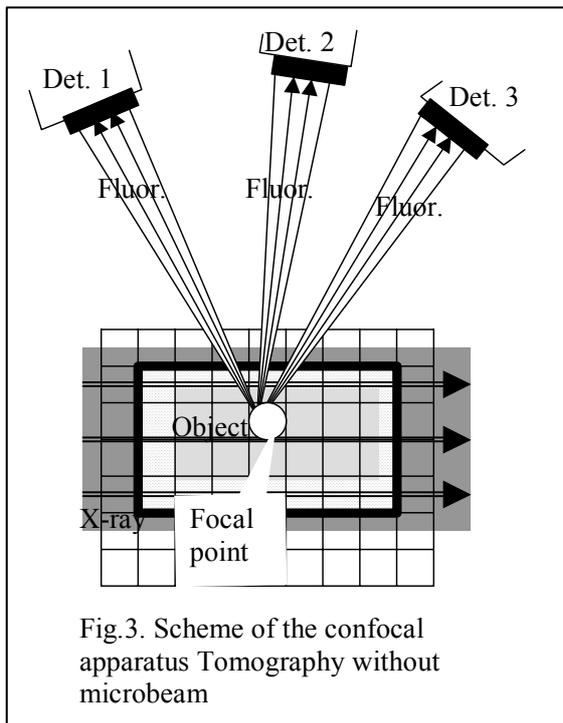


Fig.3. Scheme of the confocal apparatus Tomography without microbeam

Simulation results are presented in Fig.4. Parameters for the simulation: the sample is 5 microns' thickness Al strip in PMMA, X-ray beam energy is 10 keV, no backscattering effects are taking into account. The scanning procedure is the same as the procedure described above (Fig.1). The apparatus function of three collimators system is not so simple as in the previous case, however, we can calculate it before the

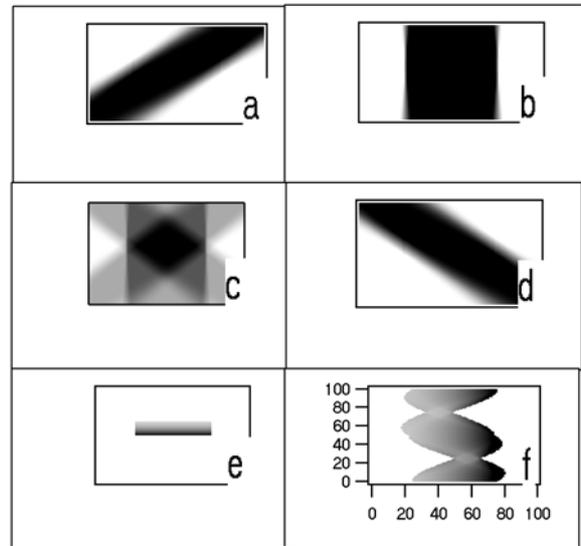


Fig.4. Simulation results. Signals collected with the different tomography set-ups. a) Signal from Det. 3 (Fig.3); b) Signal from Det.2 (Fig.3); c) Sum of Signals  $a,b,d$ ; d) Signal from Det.1. (Fig.3); e) Signal collected on Fig.1; f) Set of Projections (Fig.5).

experiment. The registered signal is the convolution of the AI image, which we are interested in visualizing, with the apparatus function. After the implementation of the deconvolution procedure it will be possible to see the result of the AI strip visualization. This result cannot give quantitative information about the sample (if attenuation is not negligible) but the boundaries of the object will be found. And again the data collection time from pixel to pixel is controlled and can be varied.

### COMPUTER X-RAY FLUORESCENCE TOMOGRAPHY

The last scheme is the computer tomography scheme. The sample is mounted on the sample stage which is moved in X-Z directions and is rotated with some angle step over  $360^\circ$ . For X-ray fluorescence tomography it is important to have the total angle because the attenuation effects for the characteristic lines are different for and rotation angles if the sample is not homogeneous and has no central symmetry. The detector collects all quanta generated on the route of the X-ray beam grabbing by the detector's solid angle. The registered signal now is a function of the X-ray microbeam position and the rotation angle (Fig.4f). The Y-direction on the image is the rotation angle axis and X-direction is the microbeam position axis). For the image visualization it is necessary to move from the signal space to the image space. Write the mathematical model of the signal formation

$$S^i(\rho, \varphi) = \Omega_{\text{det}} I_0 \mu^i \eta^i \int_{L_1} C^i(l) \Phi(l) F^i(l) dl \quad (4).$$

$L_1$  is the direction of the X-ray microbeam (contrary to the  $y'$ -axis direction (Fig.5)) and  $\Omega_{\text{det}}$  is the detector's solid angle.

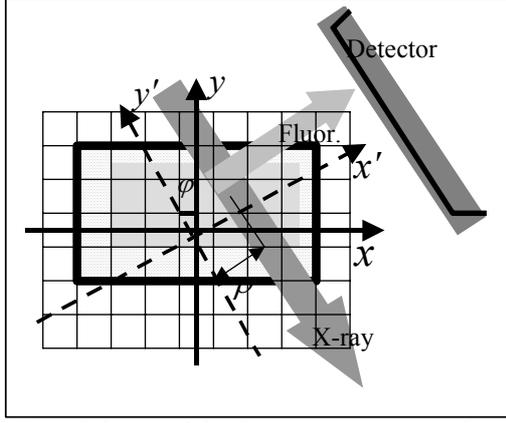


Fig.5. Scheme of the Computer Tomography. The microbeam position is described by  $\rho$ .  $\varphi$  is the rotation angle.

Let us suppose  $N$  translation steps and  $N$  rotations are done during the experiment. Then  $N^2$  spectra are collected and each spectrum is transformed to the  $M$ -component vector. The modified algebraic technique (Chukalina et al. 2002) is used to visualize  $M$  images. The integral equation (4) is presented by the nonlinear algebraic system which is solved by Kaczmarz method.

## DOSE DEPOSITION

Let us try to answer the question – how many incident photons  $TI_0$  (here  $T$  is the data collection time) are needed to resolve a fractional element concentration fluctuation  $\Delta(x, y, elem) = \frac{C_{min}(x, y)}{C_{max}(x, y)}$ .

To answer this question let us introduce the other value  $\delta = \frac{\sqrt{I}}{I}$ . Here  $I$  is the number of quanta collected by

the detector (from one pixel in the apparatus case and for one microbeam position in computer one). For computer tomography if  $N$  translations and  $N$  rotation are done during experiment, then the total time for the computed tomography is equal

$$T_{total}^{comput} = \frac{\Phi}{\delta^2} \frac{1}{\Omega_{det}} \frac{1}{C_{min}} N^2 \quad (5)$$

Now  $\Phi$  includes all attenuation terms. The situation is not the same for apparatus tomography, where it is possible to control the number of quanta coming from each pixel. The data collection time can be varied in agreement with the concentration fluctuations. Then for the pixels with the  $C_{min}$  concentration value the data collection time is equal to:

$$T_{min} = \frac{\Phi}{\delta^2} \frac{1}{\Omega_{pin} J} \frac{1}{C_{min}}$$

and for the pixels with the  $C_{max}$  concentration value it is equal to:

$$T_{max} = \frac{\Phi}{\delta^2} \frac{1}{\Omega_{pin} J} \frac{1}{C_{max}} = \frac{\Phi}{\delta^2} \frac{1}{\Omega_{pin} J} \frac{\Delta}{C_{min}}$$

Let us introduce the value  $N^{min}$  which is the number of pixels with the  $C_{min}$  concentration value. Then the number of pixels with concentration  $C_{max}$  is  $N^2 - N^{min}$  and the total time for the apparatus tomography can be calculated by

$$T_{total}^{app} = T_{min} N^{min} + T_{max} (N^2 - N^{min}) = \frac{\Phi}{\delta^2} \frac{1}{\Omega_{pin} J} \frac{1}{C_{min}} (N^{min} + \Delta [N^2 - N^{min}]) \quad (6)$$

Now we can compare the two tomography experiments (computed and apparatus) from the data collection time point of view:

$$\frac{T_{total}^{comp}}{T_{total}^{app}} = \frac{N^2}{N^{min} + \Delta [N^2 - N^{min}]} \frac{\Omega_{pin} J}{\Omega_{det}} \quad (7)$$

and can analyse the conditions when one or the other technique will be preferable.

## OPTIMAL CONDITIONS FOR SCHEME CHOICE

Let us consider three models of the sample description. The first model is built on the hypothesis that  $\Delta \rightarrow 1$ . There is a very small variation between concentrations from pixel to pixel. Then

$$\frac{N^2}{N^{min} + \Delta [N^2 - N^{min}]} \approx 1$$

and now we rewrite Expression (7)

$$\frac{T_{total}^{comp}}{T_{total}^{app}} \approx \frac{\Omega_{pin} J}{\Omega_{det}} \quad (8)$$

Let us make an estimation of this ratio based on Fig.1. The solid angle of a detector placed at 5 cm distance

from a sample is  $\Omega_{det} = \frac{r_{det}^2}{R^2} = 0.01$ . The surface of the detector window is 25 mm<sup>2</sup>. The diameter of the collimator channel is about 2  $\mu$ m. Solid angle of the

channel is  $\Omega^{col} = \frac{\pi r_{channel}^2}{R^2} = 5 \times 10^{-11}$  Compact packing of the channels by microelectronics techniques (Zaitsev et al. 2004) on a 25 mm<sup>2</sup> square chip allows the

placing of  $10^8$  channels. We have  $\frac{T_{total}^{comp}}{T_{total}^{app}} \approx 0,5$  which

means that the time needed to make the apparatus tomography experiment is twice as much as the computed tomography time.

The second model describes the situation when  $\Delta = \frac{1}{2}$ .

Then we rewrite Expression (7) in agreement with the model

$$\frac{N^2}{N^{\min} + \Delta[N^2 - N^{\min}]} = \frac{2N^2}{N^2 + N^{\min}}$$

and consider two extreme cases  $N^{\min} \rightarrow 0$  and  $N^{\min} \rightarrow N^2$ . The expression (8) is rewritten as

$$\frac{1}{2} < \frac{T_{total}^{comp}}{T_{total}^{appar}} < 1$$

and the times for both cases start to

be commensurable.

The last model is based on the condition  $\Delta \rightarrow 0$ . Then expression (7) is modified to

$$\frac{N^2}{N^{\min} + \Delta[N^2 - N^{\min}]} \approx \frac{N^2}{N^{\min}}$$

and the comparison of the data collection times is described by:

$$\frac{T_{total}^{comp}}{T_{total}^{app}} \approx \frac{\Omega_{pin} J}{\Omega_{det}} \frac{N^2}{N^{\min}}$$

This result shows that if  $\Delta \rightarrow 0$  (a sample has regions with very low concentration ratios of an element) then apparatus tomography has big advantages versus the computed one.

## CONCLUSION

In this paper confocal apparatus tomography and computer tomography are compared from an image visualization point of view. The image visualization procedure is faster and simpler for the apparatus case, however, an additional parameter (dose deposition) should be taken into account if the tomography is used in medicine. It was shown that the data collection time (or dose deposition) depends on the sample type. As a conclusion, it should be mentioned that the sequential application of both techniques can give additional advantages. The computed tomography implemented with a 'big' scanning step size and a 'big' rotation angle value gives a preliminary picture about the sample volume and then using apparatus tomography it is possible to investigate the chosen small volumes with a good lateral resolution and good sensitivity. Confocal apparatus tomography has the extra advantage of being able to perform "partial volume tomography" that is analyzing a small volume inside a bigger one, by partial scanning.

This work was partly financially supported by the NATO Scientific Affairs Division through grant nr. EST-CLG-979530, by the PICS 22-10 and RFBR Programs for scientific investigations. In addition the work of Dr. M. Chukalina was partly supported by the Institute of Crystallography RAS, Russia .

## REFERENCES

Asadchikov, V., A. Buzmakov, Yu. Zanevskii, V. Zryuev, R. Senin, S. Saveliev, L. Smikov, D. Tudosy, G.

- Cheremuhina, E. Cheremuhin, S. Chernenko and A. Chulichkov. 2004. "X-ray transmission tomography with 0.7-1.54 Å in laboratory conditions". In *Proceedings of X-ray Optics Workshop* (Nizhni Novgorod, May 2-7). IPM RAS, 123-130.
- Bernhardt, R., D. Scharnweber, B. Müller, P. Thurner, H. Schpiephake, P. Wyss, F. Beckmann, J. Goebbels and H. Worch. 2004. "Comparison of microfocus and synchrotron X-ray tomography for the analysis of osteointegration around T16AL4V-implantants". *European Cells and Materials* 7, 42-51.
- Chukalina, M, A. Simionovici, A. Snigirev and T. Jeffries. 2002. "Quantitative characterization of microsamples by X-ray Fluorescence Tomography" *X-Ray spectroscopy* 31, No. 6, 448-450.
- Golosio, B., A. Simionovici, A. Somogyi, L. Lemelle, M. Chukalina and A. Brunetti. "Internal elemental microanalysis combining X-ray fluorescence, Compton and transmission tomography". 2003. *Journal of Applied Physics* 94, No. 1, 145-156.
- Müller, B, P. Thurner, F. Beckmann, T. Weitkamp, C. Rau, R. Bernhardt, E. Karamuk, L. Eckert, S. Buchloh, E. Wintermantel, D. Scharnweber and H. Worch. 2001. "Non-destructive three-dimensional evaluation of biocompatible materials by microtomography using synchrotron radiation". In *Proceedings of the SPIE Meeting* (San Diego, Aug. 2-3), SPIE 4503, 178-188.
- Schneider, G., D. Weiss; M. A. LeGros; S. Vogt, C. Knöchel and E.H. Anderson. 2001. "Nanotomography of labeled cryogenic cells". In *Proceedings of the SPIE Meeting* (San Diego, Aug. 2-3). SPIE 4503, 156-165.
- Simionovoci, A., M. Chukalina, M. Drakopoulos, I. Snigireva, A, Snigirev, Ch. Schroer, B. Lengeler and F. Adams. 1999. "X-ray fluorescence microtomography: experiment and reconstruction". *SPIE* 3772, 304-310.
- Snigireva, I., V. Yunkin, S. Kuznetsov, M. Grigoriev, M. Chukalina, L. Shabel'nikov, A. Snigirev, M. Hoffmann and E. Voges. 2003. "Focusing properties of silicon refractive lenses: comparison experimental results with the computer simulation". *SPIE* 5195, 32-39.
- Takeda, T., T. Maeda, T. Yuasa, T. Akatsuka, T. Ito, K. Kishi, J. Wu, M. Kazama, K. Hyodo and Y. Itai. 1995. "Fluorescent scanning x-ray tomography with synchrotron radiation". *Rev. Scien. Instrum.* 66, No. 2, 1471-1473.
- Zaitsev, S., M. Chukalina and P. Mastrippolito. 2002. "X-ray fluorescence diagnostics with confocal collimator for the data collecting". In *Proceeding of the Workshop on X-ray Optics*. (Nizhni Novgorod, Russia, March 14-17). IPM RAS, 293-298 (in Russian).
- Zaitsev, S., M. Knyazev, S. Dubonos, and A. Bazhenov. 2004. "Fabrication of 3D photonic structures". *Microelectronic Engineering* 73-74, 383-387.
- Zaitsev, S., M. Chukalina, A. Simionovici and E. Ziegler. 2005. "Pilot testing method of fluorescent microtomography based on confocal collimators" *Experiment at the ESRF Proposal MI724* (February 17-20).
- simulation.  
al that can be found in the paper!

## AUTHOR BIOGRAPHIES

**MARINA V. CHUKALINA** received her PhD in Physics at the Institute of Microelectronics Technology RAS where she has been working since 1988. Her

interests include the development of signal and image processing tools for X-ray Microscopy and Tomography. Her Web-page can be found at <http://www.ipmt-hpm.ac.ru/english/labs/lcd/>

**SERGEY I. ZAITSEV** is Dr. of Science at the Institute of Microelectronics Technology RAS, Russia. His research interests concentrate on the theory of solid state kinetics.

**MAXIM KNYAZEV** is PhD student at the Institute of Microelectronics Technology from 2001. His research area is microelectronics technology.

**ALEXANDRE SIMIONOVICI** received his PhD in nuclear physics, worked at the ESR as a beamline lider and now carries his investigations in the Laboratoire des sciences de la terre, ENS, Lyon.

**DMITRY P. NIKOLAEV** received PhD degree at Moscow State University in 2004 in Physics. He has been a research scientist at the Institute for Information Transmission Problems RAS since 2000. His research activities are in the areas of computer vision with primary application to colour image understanding.

# COMPARATIVE ANALYSIS OF GAUSSIAN AND LINEAR SPECTRAL MODELS FOR COLOUR CONSTANCY

Dmitry P. Nikolaev and Petr P. Nikolayev  
Institute for Information Transmission Problems, RAS  
Bol'shoj Karetnyj lane, 19  
101447, Moscow, Russia  
E-mail: nikol@iitp.ru

## KEYWORDS

Colour constancy, spectral models, Gaussian spectral model.

## ABSTRACT

The present work considers the features and advantages of a Gaussian model of spectral functions approximation for solving the colour constancy problem. The peculiarities and advantages of the Gaussian model, as compared with traditional linear approximation models, are examined. The possibility of stable numerical estimation of the reflectance features of object surfaces on the basis of the spectral stimulus captured by a camera at different conditions of scene illumination is demonstrated. The efficiency of colour constancy algorithms that use the Gaussian model is estimated.

## INTRODUCTION

In machine vision, it is often required not only to compute the object segmentation map of an image (Nikolaev and Nikolayev 2004), but also to estimate the coloration of object surfaces. Such a problem arises, for example, when processing videos or indexing image databases in the case when a target object should be successively detected, regardless of the illumination and observation conditions, which aren't known *a priori*.

The ability of the vision system of the man and animals to estimate the reflective properties of surfaces in the case when the illumination chromaticity changes (Helmholtz 1910; Nyberg et al. 1971a), that is, the phenomenon of colour constancy (CC), has been discussed quite explicitly. CC algorithms suitable for machine vision systems are known too (Forsyth, 1990; Finlayson et al., 2001; Finlayson and Schaefer, 2001; Barnard et al., 2002). But as a rule they don't involve the spectral models of scenes and so they aren't suitable for processing of the images captured by some different sensors.

The elaboration of a solution method for such CC tasks requires introducing some *a priori* restrictions on the optical properties of the medium in which the sensor system operates. Due to considerable irreversible information loss during the transformation of a radiation signal into a sensor system response, it is necessary to match the language of spectral descriptions with the

colour discrimination capabilities of the system. For this purpose, various models are introduced for approximating the spectral characteristics of the whole vision process, from the emission properties of the illumination sources to the photosensitivities of the colour channels of the system. Such restrictions and approximations turn the CC problem into a solvable one, making it possible to build a mathematical model of the CC (Nikolayev 1985).

The CC problem allows various definitions that differ in the level of the requirements set for the final result. If it is possible to assign an etalon vector stimulus (e. g., 3-stimulus from "white" illumination) to each uniformly coloured object contained in the scene and if this stimulus is invariant to the illumination and observation conditions, then the problem of the constant estimation of the object coloration can be considered solved for this particular type of a sensor. As an etalon stimulus, it is convenient to use the stimulus that would be produced by the object observed in a hypothetical situation when it is illuminated by a diffuse equal-energy source of a given brightness. In such a definition, it is possible to describe the CC mechanism as a "correction of illumination chromaticity" (Helmholtz 1910). Such a solution of the CC problem does not allow one to predict what vector stimulus will be obtained at the output of a different sensor observing the same object. In this sense, the above problem definition can be called "weak". In the framework of this definition, the constant estimation of the source chromaticity is reduced to finding the locus of the vector stimuli obtained when directly observing diffuse sources of the same chromaticity and an arbitrary brightness. As for the problem of the constant estimation of the brightness of the scene illumination sources, it is in principle insoluble if there is no information on the scene geometry (on the arrangement of its objects and their shapes).

The estimates obtained in the framework of the "weak" CC definition are described by the magnitudes of the vectors of sensor response to incident radiation, that is, they are the projections of some spectral stimuli onto the colour space (CSp) of the particular sensor. To obtain constant estimates of the object coloration and the illumination chromaticity, which are also invariant to the choice of a sensor, it is necessary to retrieve a spectral stimulus from its CSp projection (from the

vector of response to it). Obviously, this problem is, in general, insoluble and becomes soluble only when a unique response corresponds to each of the possible etalon spectral stimuli. The restriction on the whole variety of colorations and illuminations to a set of the spectra meeting the above condition is usually performed by introducing a spectral model with a limited number of parameters. We will refer to the retrieval (in the framework of the spectral model accepted) of the spectral composition of illumination and the reflection spectra of objects as a “strong” definition of the CC problem.

Most researchers dealing with the CC problem use so-called linear spectral models (LSM). In a LSM, the space of spectral functions is confined to a 3D linear subspace of the function space. Interesting particular cases of LSM are models in which the bases are step-wise functions (so-called “banded spectral model” (Stiles and Wyszecki 1962; Land and McCann, 1971; Nyberg et al. 1971b)) or the functions of spectral sensitivity of the sensor (Lee et al. 1995). However, all these models have the same drawback: they cannot adequately describe stimuli of a high saturation. This fact is quite obvious if taking into account that spectrally separated stimuli are linearly independent.

One of few nonlinear spectral models suggested so far is a Geusebroek model (Geusebroek et al. 2001), in which the spectral functions are approximated with second-order polynomials, and the functions of spectral sensitivity of the sensor are considered decomposable

on some basis. As it will be demonstrated below, such decomposition is quite far from the ideal, both for human receptors and for an RGB-camera.

These drawbacks are no longer valid for the Gaussian spectral model (GSM) we suggested earlier (Nikolaev 1985). In this model, the spectral stimuli, the reflective properties of scene surfaces, and the sensor sensitivity functions are approximated with a 3-parameter set of various Gaussians. Let us consider some features of the GSM, trying to demonstrate its advantages.

### SENSOR MODELING IN LSM AND GSM

The Geusebroek model is accurate only for those sensors whose sensitivity spectra are decomposable on a basis that consist of a Gaussian function and its first and second derivatives. Figure 1 shows the results of approximating the sensitivity spectra of the human eye (a, c, e (Dartnall et al. 1983, Table 2)) and an RGB camera (b, d, f – a FillFactory camera with an infrared filter,  $\lambda_0 \approx 630nm$ ). The solid marks correspond to the real experimental data, while the curves correspond to best data fit. Figures 1a,b show the best fit between the Geusebroek model and the real data for the human eye and the RGB camera (the root-mean-square errors are  $s_a = 0.05$  and  $s_b = 0.01$ , respectively). As it can be seen from the plots, such an approximation cannot be considered good.

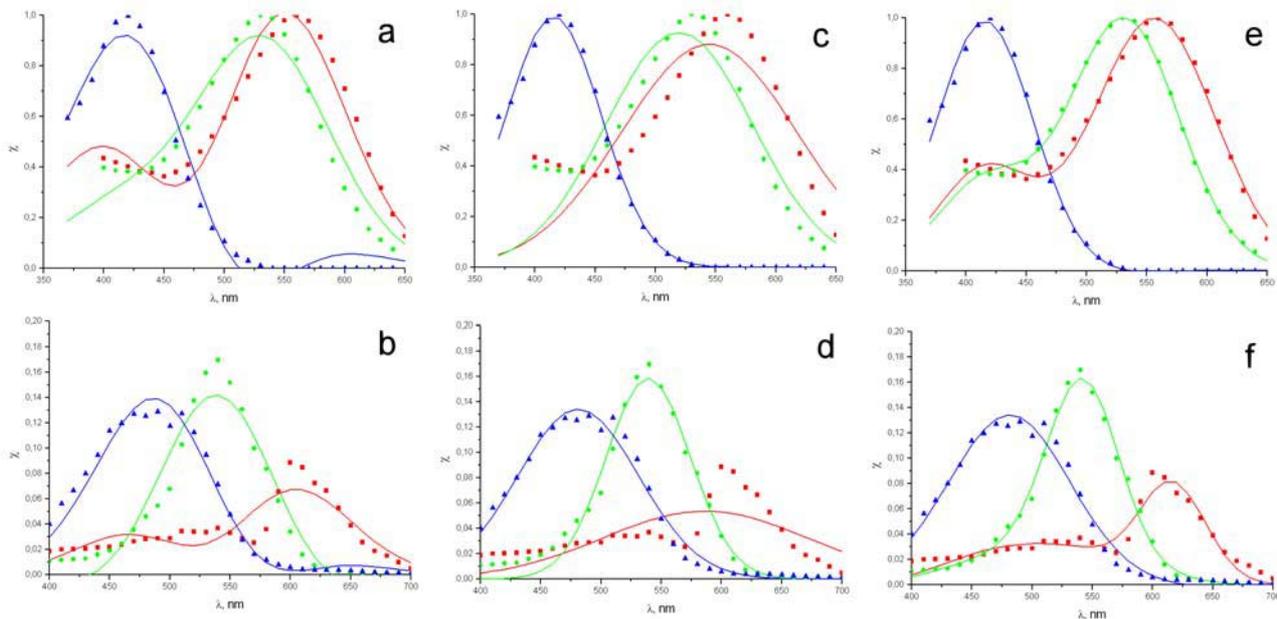


Figure 1: Results of Fitting the Sensitivity Spectra of the Human Eye and an RGB Camera

For the GSM, it is known (Nikolayev 1985) that the model is accurate for a sensor whose each sensitivity spectrum has a Gaussian profile. Unfortunately, for real sensors, such an approximation is even less adequate than the Geusebroek one (Figures 1c,d, the root-mean-square errors are  $s_c = 0.09$  and  $s_d = 0.01$ ,

respectively). However this result can be considerably improved if taking into account that the response vectors of sensors with linearly dependent sensitivities are also linearly dependent. For example, if the sensitivity spectra of the sensor have the following form:

$$\chi_{li}(\lambda) = \sum_{j=1}^3 l_{ij} \cdot G(\lambda, h_j, s_j), \quad 1 \leq i \leq 3, \quad (1)$$

where  $G(\lambda, h, s) = \exp(-s \cdot (\lambda - h)^2)$ , then there exists a linear transform of vectors from the CSp of this sensor to the CSp of a sensor with the sensitivity  $\chi_{2i}(\lambda) = G(\lambda, h_i, s_i)$ , for which the GSM is adequate. Figures 1e,f show the best fit between the extended GSM (EGSM) and the real curves for the human eye and the RGB camera (the root-mean-square errors are  $s_e = 0.02$  and  $s_f = 0.005$ , respectively). As it can be seen from the plots, among all the models considered, the EGSM describes both sensor types most adequately. For the Lee model, evidently, the approximation error for the sensor sensitivity spectra is definitely zero. To obtain the parameters of the sensor model for the EGSM, we used a shooting method that includes a few stages, in order to decrease the probability of finding a local optimum instead of the global one. At the first stage, shooting was performed for the function  $\chi_{11}(\lambda)$  and the parameters  $l_{11}, s_1, h_1$ . At the second stage – for  $\chi_{11}(\lambda), \chi_{12}(\lambda)$  and  $l_{11}, l_{12}, l_{21}, l_{22}, s_1, h_1, s_2, h_2$ , using the results of the first stage as the initial estimate. At the third stage, shooting concerns all 15 parameters (9 weight coefficients  $l_{ij}$  and 3 pairs of parameters  $h_j$  and  $s_j$ , which define the three sensitivity functions in the form (1)), starting from the values obtained at the second stage.

## RETRIEVAL OF A SPECTRAL STIMULUS FROM RESPONSES IN LSM AND GSM

After the spectral model is settled, it becomes possible to link the CSp of the sensor to the spectral stimulus function space and to solve the CC problem in the «strong» definition if it is already solved in the «weak» one. Thus, as applied to an LSM, the following expression can be obtained for the vector of the sensor response to a spectral stimulus:

$$\bar{a} = \int_0^\infty \bar{\chi}(\lambda) \cdot F(\lambda) \cdot d\lambda = \int_0^\infty \bar{\chi}(\lambda) \cdot (\bar{\alpha} \cdot \bar{b}(\lambda)) d\lambda = B\bar{\alpha}, \quad (2)$$

where  $F(\lambda)$  is the spectral stimulus,  $\bar{\chi}(\lambda)$  is the spectral sensitivity vector of the sensor,  $\bar{b}(\lambda)$  is the vector of basis functions of the model, and  $B$  is the matrix of responses to the basis functions:

$$B_{ij} = \int_0^\infty \chi_i(\lambda) \cdot b_j(\lambda) \cdot d\lambda. \quad \text{It is evident that in an LSM a spectral stimulus is retrievable for any sensor with } |B| \neq 0 \text{ and can be expressed as follows:}$$

$$F(\lambda) = \bar{b}(\lambda) \cdot B^{-1} \bar{a}. \quad (3)$$

For a trichromatic sensor that satisfies the GSM, we earlier described an analytic procedure of calculating the parameters of the spectral stimulus  $F(\lambda) = L \cdot G(\lambda, H, S)$  from the corresponding response vector  $\bar{a}$ , provided that the dispersions of the sensor

sensitivity spectra are equal ( $s_1 = s_2 = s_3$ ) (Nikolayev 1985). Let us show how a Gaussian stimulus  $F(\lambda)$  can be retrieved from the responses to it, produced by an arbitrary GSM sensor.

In the general case of a sensor with sensitivities  $\chi_i(\lambda) = G(\lambda, h_i, s_i)$ , it is not difficult to find the  $H$  and  $L$  components if the stimulus saturation,  $S$ , is already found:

$$\left\{ \begin{aligned} H &= \frac{\sum_{i=1}^3 \left( S \cdot s_i \cdot (h_i^2 - 1) + (S + s_i) \cdot \ln \frac{a_i \cdot \sqrt{S + s_i}}{\sqrt{\pi}} \right) \cdot r_i}{-2 \cdot S \cdot \sum_{i=1}^3 h_i \cdot s_i \cdot r_i}, \\ L &= \sum_{i=1}^3 e_i \cdot a_i / \sum_{i=1}^3 e_i^2 \end{aligned} \right. \quad (4)$$

where  $r_i = S_{(i+1) \bmod 3} - S_{(i+2) \bmod 3}$  and  $e_i = \exp(-S \cdot s_j \cdot (H - h_j)^2 / (S + s_j)) \cdot \sqrt{\pi} / (S + s_j)$ . The equation for finding  $S$  is, in general, transcendental, has a complex structure, and, to all appearance, has no analytical solution. Nevertheless, one can solve this equation numerically, using, for example, a shooting method. In this case, it is reasonable to take as the initial approximation of  $S$  the solution obtained for the equal dispersions of the sensitivity spectra:

$$\frac{1}{S} = \frac{\sum_{i=1}^3 h_i \cdot h_{(i+1) \bmod 3} \cdot (h_i - h_{(i+1) \bmod 3})}{(h_1 - h_2) \cdot \ln \frac{a_3}{a_2} + (h_2 - h_3) \cdot \ln \frac{a_1}{a_2}} - \frac{1}{s_2}. \quad (5)$$

Each iteration of the shooting procedure implies calculating the parameters  $\{H, L\}$  from the current value of  $S$ , using Eq.(4), and subsequently finding the response vector for the Gaussian stimulus with the parameters  $\{L, S, H\}$ . Shooting is performed with respect to the parameter  $S$ , the goal being to minimize the discrepancy between the computed and the actual response.

## ESTIMATION OF CC EFFICIENCY FOR LSM AND (E)GSM

Let us build an algorithm for numerically solving the CC problem for an LSM and (E)GSM under equal conditions (identical sets of coloured samples to test, the same set of illumination sources, and a single RGB camera) and compare the solution accuracies. Assume that the scene illumination conditions allow us to use the simplest mechanism of the color constancy – «illumination correction by a white sample» (Land 1977).

Thus, in each experiment (observation of the  $n$ -th sample illuminated with the  $k$ -th source) the algorithm receives two input vector stimuli:

$$\bar{a}_{n,k} = \int_0^\infty \bar{\chi}(\lambda) \cdot \Phi_n(\lambda) \cdot S_k(\lambda) \cdot d\lambda \quad \text{and}$$

$\bar{a}_{0,k} = \int_0^\infty \bar{\chi}(\lambda) \cdot S_k(\lambda) \cdot d\lambda$  (from a white sample). The goal of the algorithm is to estimate  $\Phi_n(\lambda)$ , that is, to solve the CC problem in its «strong» definition. This task can be solved in two steps:

1) Using the methods described in the previous section, retrieve the spectral stimuli  $F_{n,k}(\lambda) = \Phi_n(\lambda) \cdot S_k(\lambda)$  and  $F_{0,k}(\lambda) = S_k(\lambda)$  from the response vectors  $\bar{a}_{n,k}$ .

2) Estimate the reflectance curve of the  $n$ -th sample as the ratio of the stimuli found at the first step:  $\tilde{\Phi}_{n,k}(\lambda) = \tilde{F}_{n,k}(\lambda) / \tilde{F}_{0,k}(\lambda)$ .

This is, in fact, the solution of the CC problem.

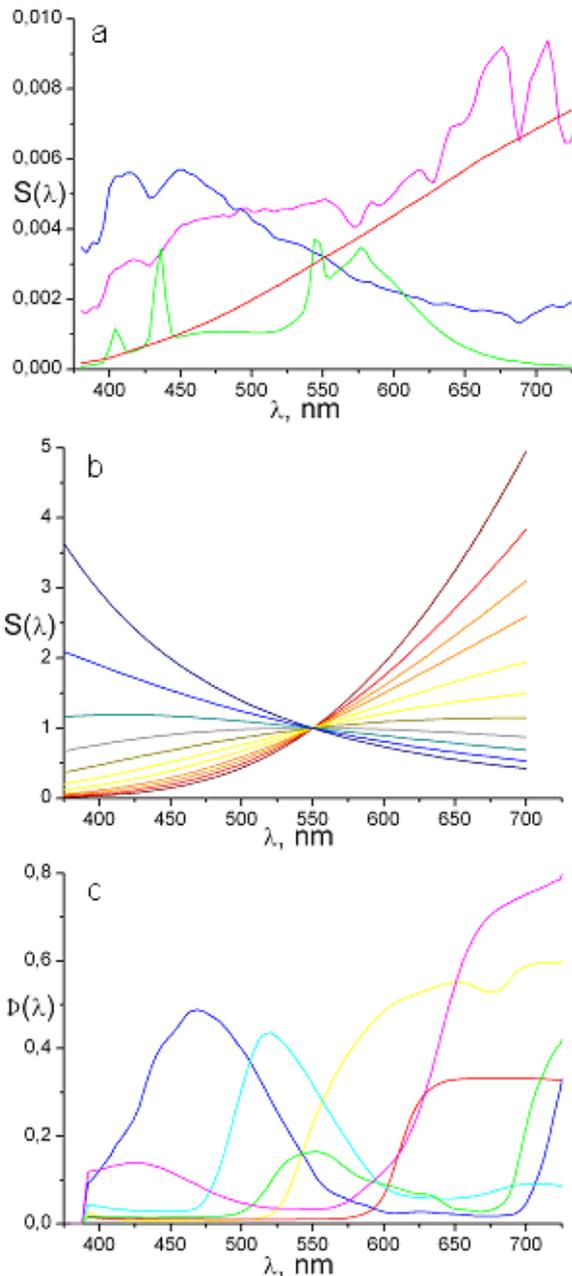


Figure 2: Emission and Reflectance Curves Used in the Experiment

To estimate the efficiency of a spectral model, let us calculate a «virtual vector stimulus»,

$$\tilde{f}_{n,k} = \int_0^\infty \bar{\chi}(\lambda) \cdot \tilde{\Phi}_{n,k}(\lambda) \cdot d\lambda, \text{ for the same } n\text{-th sample}$$

being illuminated with an etalon equal-energy source and compare it with the true stimulus

$$\bar{f}_{n,k} = \int_0^\infty \bar{\chi}(\lambda) \cdot \Phi_{n,k}(\lambda) \cdot d\lambda.$$

To compare LSM and EGSM, we chose among the LSM family the model suggested by Lee (Lee et al. 1995) and banded spectral model. As already mentioned above, Lee model uses the vector of spectral sensitivity of the sensor,  $\bar{\chi}(\lambda)$ , as the linear basis. The two models were compared along the following scheme: for a set of  $N$  samples and a selected sensor, we calculated the estimates  $\tilde{\Phi}_{n,k}(\lambda)$  and fixed on the chromaticity space of the sensor the Euclidean distance,  $r_{n,k}$ , between the projections of the estimation vector  $\tilde{f}_{n,k}$  and the «etalon»  $\bar{f}_n$ . As a measure of the efficiency of each model for the given illumination conditions, we use the root-mean-square (over all  $N$  samples) deviation,  $r_k$ , of the actual estimate from the «absolutely» constant one. 170 reflectance functions of «natural colorants» and 4 emission spectra of «natural non-Planck sources» tabbed with a step of  $4nm$  in the range from  $375nm$  to  $750nm$  were taken from the database of the Computational Vision Lab of S. Fraser University, Canada (<http://www.cs.sfu.ca/~colour>). To better compare our results with results of other groups worldwide, we added to this set 11 Planck sources with colour temperatures ranging from  $2000^0K$  to  $30000^0K$ .

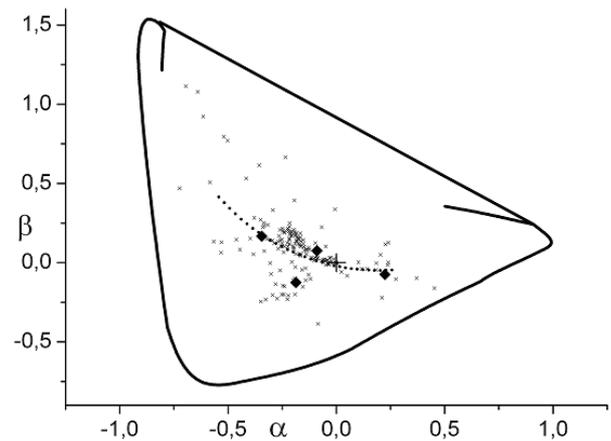


Figure 3: Positions of Stimuli on the Chromaticity Plane

Figure 2 shows normalized emission curves for 4 non-Planck sources (a) and 11 Planck sources (b); and 6 reflectance curves for the most saturated samples (c) (from those 170), which were used in the experiment with CC models. Figure 3 displays the positions of the vector stimuli from 15 sources and 170 samples on the chromaticity plane of the sensor (the latter were calculated for the case of an equal-energy illumination source).

Figure 4 maps onto the sensor chromaticity plane the composite diagrams for the LSM (left column – Lee model, right column – banded model) and EGSM (central column) under ideal Planck (bottom row) and «natural» (top row) illumination. The hollow circles in the central parts of the diagrams correspond to shifts of the *constant estimates* of sample chromaticities with respect to the «ideal CC» positions, while the solid dots correspond to shifts of the initial *constant* values.

The more compact the hollow circle cluster is relative to the solid dot cluster, the better the CC algorithm operates. The overall statistics of this experiment is the following. Under «natural» illumination (K=4), the efficiency is  $q=7.9$  for the EGSM, while it is  $q=3.6$  for the Lee LSM and  $q=6.2$  for banded LSM. Planck illumination (K=11) increases  $q$  to 8.0 for the EGSM, while keeping it at 3.6 for the Lee LSM and increasing  $q$  to 7.0 for banded LSM.

The efficiency  $q$  was calculated by averaging (over the K sources) the ratio  $r_k/r_{0,k}$ , where  $r_{0,k}$  is the root-mean-square deviation from the «ideal» estimate for the

*constant* (i.e., input) stimulus, and  $r_k$  is the same deviation for the *constant* estimate of each sample in the CC model chosen. Thus, over the set of 170 types of functions  $\Phi_n(\lambda)$  and for two sets of illumination sources, the  $q$ -criterion judges the EGSM to be more advantageous in the sense of the accuracy of estimating these functions.

Furthermore, the experiment allowed us to qualitatively conclude that the EGSM estimate accuracy increases with the proximity of  $\Phi_n(\lambda)$  to a symmetric single-extremum function within the sensor sensitivity range.

In conclusion, we would like to note that the triad of parameters,  $h$ ,  $s$ , and  $L$ , of the Gaussian curve, which fits in the GSM all the spectral curves of the vision process, substantially correlates with the three components of the colour perception of a trichromate: hue, saturation, and brightness (or lightness, which is exactly the same), respectively.

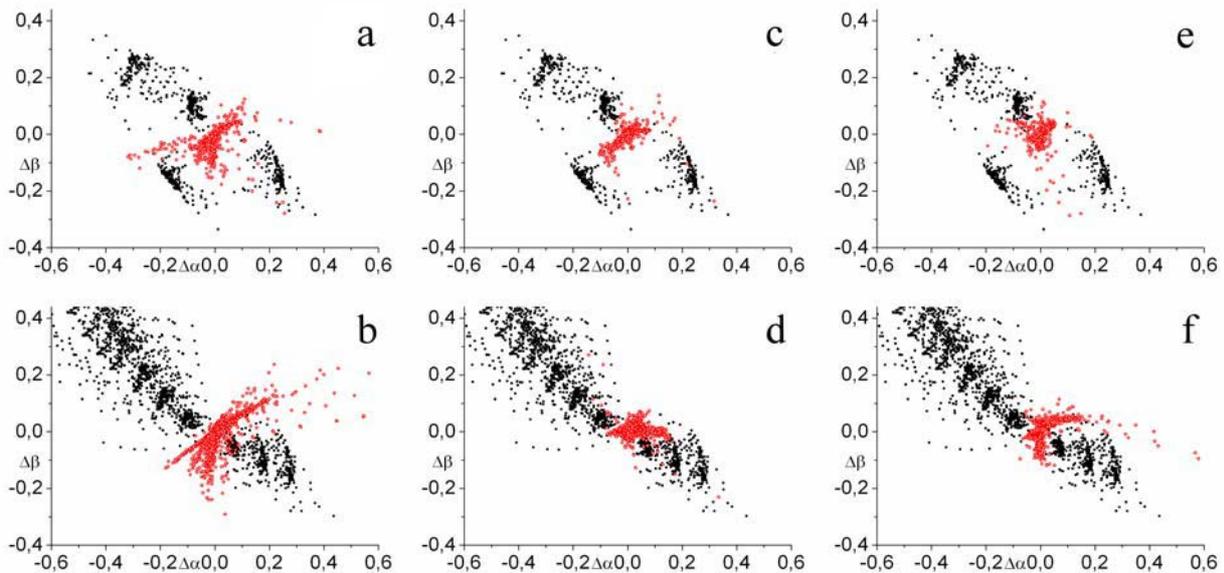


Figure 4: Colour Constancy Efficiency of EGSM and LSM

## CONCLUSION

The present work suggests to use a Gaussian model of spectral functions approximation for solving the CC problem. This model demonstrates good performance in numeric modelling with the extensive set of natural pigments illuminated both ideal Planck and real light sources. Comparative analysis indicates that a Gaussian model is potentially more effective at estimating the chromaticity of the scene illuminant and the scene objects than the linear spectral models while using the same CC algorithm. This was demonstrated for the “banded spectral model” (that underlies scale-by-max Retinex algorithm) and Lee’s linear model that uses

functions of spectral sensitivity of the sensor as a basis. In the following we plan to proceed the comparison of the performance capabilities of our spectral model for CC with other existing linear and non-linear spectral models.

## REFERENCES

- Barnard K.; L. Martin, A. Coath and B. Funt. 2002. "A comparison of color constancy algorithms. Part two. Experiments with image data". *IEEE Trans. in Image Processing*. 11. No. 9, 985-996.
- Dartnall, H.J.; J.K. Bowmaker and J.D. Mollon. 1983. "Human visual pigments: microspectrophotometric results

- from the eyes of seven persons". *Proc. of Royal Soc. of London. B* 220, 115-130.
- Finlayson, G.D.; S. Hordley and P. Hubel. 2001. "Color by correlation: a simple, unifying framework for color constancy". *IEEE Trans. Pattern Anal. Mash. Intel.* 23, No. 12, 1209-1221.
- Finlayson, G.D. and G. Schaefer. 2001. "Solving for colour constancy using a constrained dichromatic reflection model". *Int. J. Computer Vision.* 42, No. 3, 127-144.
- Forsyth D.A. 1990. "A novel algorithm for colour constancy". *Int. J. Computer Vision.* 5, No. 1, 5-36.
- Geusebroek, J.-M.; R. van den Boomgaard and A. Smeulders. "Color invariance". 2001. *IEEE Trans. Pattern Anal. Mash. Intel.* 23, No. 12, 1338-1350.
- Helmholtz, H.L.F. 1910. *Theatise on Physiological Optics*. English translation in 1925. Dover, N.J.
- Land, E.H. and J. J. McCann. 1971. "Lightness and retinex theory". *J. Opt. Soc. Amer.* 61, 1-11.
- Land, E.H. 1977. "The retinex theory of color vision". *Sci. Am.* 237, No. 6, 108-128.
- Lee, S.D; C.Y. Kim and Y.S. Seo. 1995. "Linear model of surface and scanner characterization method". *Proceedings of SPIE.* 2414, 84-93, Device-Independent Color Imaging II, Eric Walowitz, Ed.
- Nikolayev, P.P. 1985. "Model of colour constancy for the case of continuous spectral functions". *Biophysics* 30, No. 1, 112-117 (in Russian and English translation).
- Nikolaev, D.P. and P.P. Nikolayev. 2004. "Linear colour segmentation and its implementation". *Comp. Vision and Image Understanding.* 94, 115-139.
- Nyuberg, N.D.; M.M. Bongard and P.P. Nikolayev. 1971a. "I. On the constancy of perception of coloration". *Biophysics* 16, No. 2, 285-293 (in Russian).
- Nyuberg, N.D.; P.P. Nikolayev and M.M. Bongard. 1971b. "II. On the constancy of perception of coloration". *Biophysics* 16, No. 6, 1052-1063 (in Russian).
- Stiles, W.S. and G.W. Wyszecki 1962. "Counting metameric object colors". *J. Opt. Soc. Amer* 52, No. 2, 313-328.

## AUTHOR BIOGRAPHIES

**Dmitry P. Nikolaev** was born in Moscow, Russia. He studied physics, obtained master degree in 2000 and PhD degree in 2004 from Moscow State University. Since 2000, he has been a research scientist at the Institute for Information Transmission Problems, RAS. His research activities are in the areas of computer vision with primary application to colour image understanding. His e-mail address is [dimonstr@iitp.ru](mailto:dimonstr@iitp.ru) and his Web-page can be found at <http://www.ddt.ru/chaddt/Lace/pub.html>.

**Petr P. Nikolaev** was born in Moscow, Russia. He studied physics and obtained master degree in 1966 from Moscow State University. He obtained PhD degree in biology in 1975 from Institute for Information Transmission Problems, RAS and Dr. of Science degree in physics in 1993 from Institute for Biophysics, RAS. Since 1966, he has been a research scientist at the Institute for Information Transmission Problems, RAS where he is now leading a research group in the field of computer colour vision. His e-mail address is [nikol@iitp.ru](mailto:nikol@iitp.ru).



# **Simulation in Technology, Processes and Operations Research**



# **SIMULATION IMPROVES STAFFING PROCEDURE AT AN OIL CHANGE CENTER**

Edward J. Williams  
Jory D. Bales, Jr.

Justin A. Clark  
Renee M. Amodeo

Department of Industrial & Manufacturing Systems Engineering  
2330 Engineering Complex, University of Michigan – Dearborn  
4901 Evergreen Road, Dearborn, MI 48128  
U.S.A.

## **KEYWORDS**

Discrete-event process simulation, service industry, personnel staffing, Arena®

## **ABSTRACT**

Discrete-event process simulation has long ago expanded from its initial bailiwick of manufacturing and production usage to benefit businesses across a broad spectrum of service industries such as travel, lodging, restaurants, and health care. The study presented here arose in the context of a senior-level university simulation class as a semester project. In this study, simulation was applied to the day-to-day operations of a drive-in facility to service customers' privately owned motor vehicles. As a result of the simulation study, managers of the business, supported by industrial engineers, became both more aware of improvement opportunities and better able to assess their comparative potential impacts on the profitability of the business.

## **INTRODUCTION**

The client of this simulation study was the management of a vehicle drive-in service center; these managers sought to improve the quality of service, hence the competitiveness, and hence the profitability, of their business. Simulation, with which they were initially unfamiliar, has long been entrenched in the manufacturing industry (Miller and Pegden 2000), and has now moved decisively into the field of productivity improvement in the service industries (Herbst, Junginger, and Kühn 1997). Examples of the wide-ranging uses of simulation within the service sectors include an examination of the information technology support function of a multinational construction firm (Hlupic and Bosilj-Vuksic 2004), improvement of emergency response to floods (Wattanapanich and Tandayya 2004), improvement of urban bus operations (Osmólski, Osmólski, and Kaczalski 2003), and improving the operations of a drive-through credit union (Williams and Zottolo 2001). In this study, the business owner and his two managers identified their primary concern that of achieving a proper and effective staffing configuration, thereby increasing the profitability of the

business by decreasing service times and thus surely increasing throughput and customer satisfaction. Several analytical obstacles and complexities en route to this goal included highly variable customer arrival rates, highly variable service times (attributable both to different menus of services requested and to different vehicles requiring different service times even for the same service), and staff scheduling constraints imposed by different skill mixes and levels among the employee service technicians. As an example of the interrelationships among these considerations, the different skill levels among the technicians were a heavy additional contributor to the high variability in service times.

In this paper, we first present an overview of operations at the service center. Next, we describe collection of the data, and the construction, verification, and validation of the simulation model. We then provide a summary of the conclusions our client and we collaboratively reached.

## **OVERVIEW OF OPERATIONS**

The locus of this simulation study was a drive-in vehicle service center of the type very common in the United States. These centers, a well-known and plausible pathway to entrepreneurship for aspiring businesspersons, sometimes via franchise (Horowitz and Shilling 1989) are certainly fraught with significant challenges, such as vulnerability to thuggish crime (Meyer 2005), employee selection, training, and retention problems, and chronic necessity for “hands-on” management. A layout of the center appears in Figure 1 near the end of this paper. The owner of a vehicle drives into the center from a main trunk road, selects a queue leading to one of the three service bays (bay one, closest to the road, is the default), and waits in that queue (due to constraints of vehicle size relative to maneuvering space, jockeying is well-nigh impossible) until that bay is open. However, balking is occasionally observed, typically when all queue lengths exceed seven vehicles, inasmuch as it is both unsafe and technically illegal to wait in queue while the vehicle is still physically on the busy trunk road. Then, guided by a technician, the customer carefully advances into the bay

(the vehicle's wheels must straddle the "pit" wherein some technicians work, not fall in!) and remains in the vehicle while the vehicle receives routine maintenance services such as a change of engine oil, flush of transmission fluid, and/or flush of coolant fluid. Insofar as possible, these services are performed at least approximately concurrently. The customer then pays for the service(s), again typically without leaving the vehicle, and drives off the premises and returns to the trunk road. Of necessity, inbound and outbound traffic is rigorously separated by rigid enforcement of unidirectional ("One Way") signs and directives. Additional services, although available upon request, constitute less than 1% of the center's total business. About 85% of the business volume is compressed into three high-volume days per calendar week (Mondays, Fridays, and Saturdays); therefore, efficient peak business staffing and employee utilization received high attention during the study. All services are performed as soon as a technician able to perform them is available. The mechanical training the technicians receive is primarily "on the job" or "hands-on;" therefore, the skill level of each technician is approximately proportional to their seniority at this and/or similar service centers. At present, there is neither certification nor formal training required to perform these services.

## DATA COLLECTION

In view of the client's stated objective of improving the business's staffing configuration, data collection began with a careful overview of current operational practices before undertaking collection of numeric data. The observed, and quite informal, work assignments were: managers gather information from the customers on arrival and receive payment from them on exit, whereas technicians, both low- and high-skilled, check and replenish fluids (oil, transmission, and coolant) or drain old oil. The technicians' duties are conspicuously divided into "topside" (work at ground level, such as checking fluid levels) and "pit" (draining old oil); predictably, any technician much prefers "topside" duty to "pit" duty. As a typical workplace privilege, the high-seniority employees gravitate to the "topside" work; since, as stated above, skill level is strongly correlated with seniority, management had drifted, largely unawares, into a situation in which the "topside" tasks were performed by the more experienced, skilled technicians. Rather than have explicitly scheduled meal and break times, both managers and technicians take brief breaks as lulls in customer arrivals permit; therefore, no such breaks were included in the model. Times for various services are related to the size of the customer's vehicle; therefore, when a customer requests more than one service, there exists positive correlation between pairs of service times. Even though services can often be performed at least partly concurrently when a customer requires multiple services, a vehicle of course cannot leave its service bay until the last or

longest service time is complete. During the project team's entire data collection (observational data were collected during periods of several peak hours on several different Mondays, Fridays, and Saturdays), no instances of equipment failure or serious service error were observed. Therefore, neither downtimes nor "same customer immediately returns angry, seeking correction" were included in the model. However, the client was apprised of this omission relative to potential extensions of the model to further uses and/or similar business establishments, in keeping with an important principle of successful simulation projects: Ensure the client understands and agrees to the scope and limitations of the model (Sadowski and Grabau 2004).

More obviously and routinely, data collection included gathering data on arrival times, service times, and the frequency with which each of the three primary services (oil, transmission, or coolant) was requested. Since each of these three services could be requested independent of the other two, there were eight ( $2^3$ ) possible service "menus." Since these data could be collected unobtrusively and checked against the client's recollection and various logbooks, the Hawthorne effect (Dilworth 2000) did not arise despite the fact that all services are inherently "manual."

## MODEL BUILDING, VERIFICATION, AND VALIDATION

In keeping with the university course syllabus within which this project was undertaken, the Arena® simulation software tool (Bapat and Sturrock 2003) was used throughout. Arena® provided many standard features and conveniences for this model-building effort, including *Create*, *Process*, and *Dispose* modules, the ability to construct a *Schedule* specifying the changes in arrival rates as an empirical function of time of workday, and the ability to relegate details of the individual processes (transmission flush, oil change, and coolant flush) to *Submodels*. The use of *Submodels*, in keeping with traditional and thoroughly vetted concepts of "structured program design," (Hoffer, George, and Valacich 2002) eased the subsequent tasks of verification, validation, and high-level, non-technical explanation of the model to the client, and may confidently be expected to ease subsequent modification and expansion of the model. A screen shot of one submodel is shown in Figure 2 at the end of the paper. More specific to this real-world process, the possibility of multiple services being performed approximately concurrently was handled by conceptually cloning the customer's vehicle in a *Separate* module. A downstream *Batch* module later allowed the vehicle to proceed to payment and exit when all service(s) requested by the customer were complete. The ability of the Arena® *Batch* module to batch the clones by the unique entity attribute assigned to them just upstream from the *Separate* module corrected the initial error that vehicle #1's oil change, for example, and vehicle #2's coolant flush were incorrectly batched into an exiting

vehicle (Kelton, Sadowski, and Sturrock 2004). Arena® also provided features easing the task of constructing the model so that no customers could enter after closing time, yet all customers already in the system could continue until their requested services and “cashing out” were complete.

The base model, representing current conditions and operating procedures, was then verified and validated. Verification and validation techniques used included structured walkthroughs, sending only one entity with specified attributes through the model on a step-by-step observational basis, historical data validation, directional tests (e.g., if input rate is increased, queue lengths should either remain the same or increase), and, last, a Turing test in collaboration with the client (Sargent 2004). After correction of errors, the input and performance metrics of this initial model matched system observations to within 4%.

A second model, representing proposed changes of staffing and labor pooling, was then constructed, verified, and validated. In contrast to the initial model, whose construction required several weeks, the second model was constructed in less than a week, since the initial model served as a firm foundation. To construct this model, the additional Arena® feature of *Resource Sets* was used. When using this feature, the modeler first defines *Resources* (as had already been done in the base model), then defines a *Resource Set* comprising *Resources* in descending order of desirability of use, and then assigns the *Resource Set* to completion of a task. In this way, use of the model experimented with possibilities of using less skilled employees during less busy times (allowing them to be trained and coached in unhurried, non-stressful situations), using more skilled employees during less busy times (allowing managers time to attend to administrative functions such as preparing tax returns or quarterly reports), or allowing a trainee to complete a task but immediately having a technician experienced in the same task check the quality of the work.

## RESULTS AND CONCLUSIONS

As often happens in successful simulation studies, the client derived ideas and insights beyond the straightforward examination of predictions and confidence intervals pertaining to standard performance metrics (Profozich 1998). In this case study, quantitative results of the model helped managers reassign technicians to teams to simultaneously meet the training needs of the less experienced workers, avoid unnecessarily annoying the higher-seniority workers with “pit” work assignments, and ameliorate the long-standing line imbalance due to the fact that “topside” processes take much longer than “pit.” Additionally, the possibility of having a single queue for “the next available bay” (a thought which immediately occurs to anyone who has studied queueing theory and/or waited for a teller inside a modern bank) proved very enticing when tried in the model. Indeed, so promising were its

performance metrics that management is considering it anew, even at the cost of restriping the lot, and emplacing signage and plastic cones. Qualitative insights spawned by the project included the realization that customers’ waiting time (during this time, in the manager’s words, “most people read, talk on their cell phones, or just stare into space”) could become an opportunity for gentle marketing (e.g., providing leaflets describing services the customer had not already requested). The first author, in particular, was reminded of how hotels have “improved” elevator service by placing mirrors in the halls near the elevators, thereby allowing customers to preen themselves. More recently, McDonald’s Corporation has similarly systematically added value to customers’ times in queue by displaying menu choices and touting new promotions (Schlosser 2002).

## ACKNOWLEDGMENTS

The authors, particularly the first author, express gratitude to professional friend and colleague Marcelo Zottolo, Production Modeling Corporation, Dearborn, Michigan, for a careful and helpful perusal of this paper.

## REFERENCES

- Bapat, Vivek, and David T. Sturrock. 2003. The Arena Product Family: Enterprise Modeling Solutions. In *Proceedings of the 2003 Winter Simulation Conference*, Volume 1, eds. Stephen E. Chick, Paul J. Sánchez, David Ferrin, and Douglas J. Morrice, 210-217.
- Dilworth, James B. 2000. *Operations Management: Providing Value in Goods and Services*, 3<sup>rd</sup> ed. Orlando, Florida: The Dryden Press.
- Herbst, J., S. Junginger, and H. Kühn. 1997. Simulation in Financial Services with the Business Process Management System ADONIS. In *Proceedings of the 9<sup>th</sup> European Simulation Symposium*, eds. Winfried Hahn and Axel Lehmann, 491-495.
- Hlupic, Vlatka, and Vesna Bosilj-Vuksic. 2004. Business Process Modeling Using SIMUL8®. In *Proceedings of the 16<sup>th</sup> European Simulation Symposium*, eds. György Lipovszki and István Molnár, 191-196.
- Hoffer, Jeffrey A., Joey F. George, and Joseph S. Valacich. 2002. *Modern Systems Analysis and Design*, 3<sup>rd</sup> ed. Upper Saddle River, New Jersey: Prentice-Hall, Incorporated.
- Horowitz, David, and Dana Shilling. 1989. *The Business of Business*. New York, New York: Harper & Row, Publishers.
- Kelton, W. David, Randall P. Sadowski, and David T. Sturrock. 2004. *Simulation with Arena*, 3<sup>rd</sup> ed. New York, New York: The McGraw-Hill Companies, Incorporated.
- Meyer, Zlati. 2005. Robbery Is Fatal at Repair Shop. In *The Detroit News and Free Press*, Volume 131 number 132 (1 January 2005), page 3A.
- Miller, Scott, and Dennis Pegden. 2000. Introduction to Manufacturing Simulation. In *Proceedings of the 2000 Winter Simulation Conference*, Volume 1, eds. Jeffrey A. Joines, Russell R. Barton, Keebom Kang, and Paul A. Fishwick, 63-66.

- Osmólski, Wojciech, Waldemar Olmólski, and Pawel Kaczalski. 2003. Simulation of City Bus Operation Processes as Transportation System Realization. In *Proceedings of the 15<sup>th</sup> European Simulation Symposium*, eds. Alexander Verbraeck and Vlatka Hlupic, 515-522.
- Profozich, David. 1998. *Managing Change with Business Process Simulation*. Upper Saddle River, New Jersey: Prentice-Hall, Incorporated.
- Sadowski, Deborah A., and Mark R. Grabau. 2004. Tips for Successful Practice of Simulation. In *Proceedings of the 2004 Winter Simulation Conference*, Volume 1, eds. Ricki G. Ingalls, Manuel D. Rossetti, Jeffrey S. Smith, and Brett A. Peters, 61-66.
- Sargent, Robert G. 2004. Validation and Verification of Simulation Models. In *Proceedings of the 2004 Winter Simulation Conference*, Volume 1, eds. Ricki G. Ingalls, Manuel D. Rossetti, Jeffrey S. Smith, and Brett A. Peters, 17-28.
- Schlosser, Eric. 2002. *Fast Food Nation: The Dark Side of the All-American Meal*. New York, New York: HarperCollins Publishers Incorporated.
- Wattanapanich, Chirawat, and Pichaya Tandayya. 2004. Distributed Simulation of an Emergency System for the Flood Disaster in Hat Yai, Thailand. In *Proceedings of the 18<sup>th</sup> European Simulation Multiconference*, ed. Graham Horton, 187-192.
- Williams, Edward J., and Marcelo Zottolo. 2001. Upgrading Drive-Through Services of a Credit Union Via Simulation. In *Proceedings of Harbour Maritime & Multimodal Logistics Modelling & Simulation Conference* [on compact disc], eds. Agostino G. Bruzzone, Yuri Merkurjev, Luca Maria Gambardella, and Norbert Giambiasi

## AUTHOR BIOGRAPHIES



**EDWARD J. WILLIAMS** holds bachelor's and master's degrees in mathematics (Michigan State University, 1967; University of Wisconsin, 1968). From 1969 to 1971, he did statistical programming and analysis of biomedical data at Walter Reed Army Hospital, Washington, D.C. He joined Ford Motor Company in 1972, where he worked until retirement in December 2001 as a computer software analyst supporting statistical and simulation software. After retirement from Ford, he joined Production Modeling Corporation, Dearborn, Michigan, as a senior simulation analyst. Also, since 1980, he has taught evening classes at the University of Michigan, including both undergraduate and graduate simulation classes using GPSS/H™, SLAM II™, SIMAN™, ProModel®, SIMUL8®, or Arena®. He is a member of the Institute of Industrial Engineers [IIE], the Society for Computer Simulation International [SCS], and the Michigan Simulation Users' Group [MSUG]. He serves on the editorial board of the *International Journal of Industrial Engineering – Applications and Practice*. During the last several years, he has given invited plenary addresses on simulation and statistics at conferences in Monterrey, México; Istanbul, Turkey; Genova, Italy; and Riga, Latvia. He has just served as Program Chair of the 2004

Summer Computer Simulation Conference, and is serving as Program Chair for the 2005 IIE Simulation Conference and the 2005 Summer Computer Simulation Conference. E-mail address and university web pages: <[williame@umdsun2.umd.umich.edu](mailto:williame@umdsun2.umd.umich.edu)> and <<http://www-personal.umd.umich.edu/~williame>>.

**JUSTIN CLARK** is a senior student at the University of Michigan – Dearborn, studying Industrial Engineering with a business concentration. He is an active member of the Institute of Industrial Engineers (IIE). His professional interests include Simulation, Ergonomics, and Project Management. From 1996 to 2001 he performed machine setups for Draw-Tite Company. From 2001 to 2003 he worked as a line leader for Absopure Water Company. Upon graduation, in August 2005, he plans to pursue a master's degree in Engineering Management.

**RENEE M. AMODEO** is a senior at the University of Michigan – Dearborn studying Industrial and Systems Engineering with a personalized focus of Environmental Science. She is the Vice President of the student Institute of Industrial Engineers (IIE) chapter, and a member of the Michigan Association of Environmental Professionals (MAEP). She most avidly studies the interactions that business, ergonomics, and the environment have with engineering. Since 1995, she has worked in a family-owned restaurant, Sam's Sorrento Pizza of Utica, Michigan, and currently holds a management position. She has also acquired a second part-time job in 2001 at G2 Consulting Group, Troy, Michigan, as an Environmental Technician, where she assesses environmental impacts and writes technical reports. After her expected graduation date of August 2005, she plans on initiating/developing an engineering career, with the intention of continuing her education.

**JORY D. BALES JR.** is a senior at the University of Michigan – Dearborn. He is pursuing a bachelor's degree in Industrial Systems Engineering with a concentration in Manufacturing. He is an active member and president of the University of Michigan – Dearborn chapter of the Institute of Industrial Engineers. He is also a member of the Society of Manufacturing Engineers. His career interests include engineering sales, project management, and simulation. Jory has been an employee of Red Lobster, a seafood restaurant chain, for four years. During this time he has worked as a waiter and bartender. After graduation Jory plans to start his professional career, and pursue a master's degree in a job-related field.

# Map of Service Center Facility

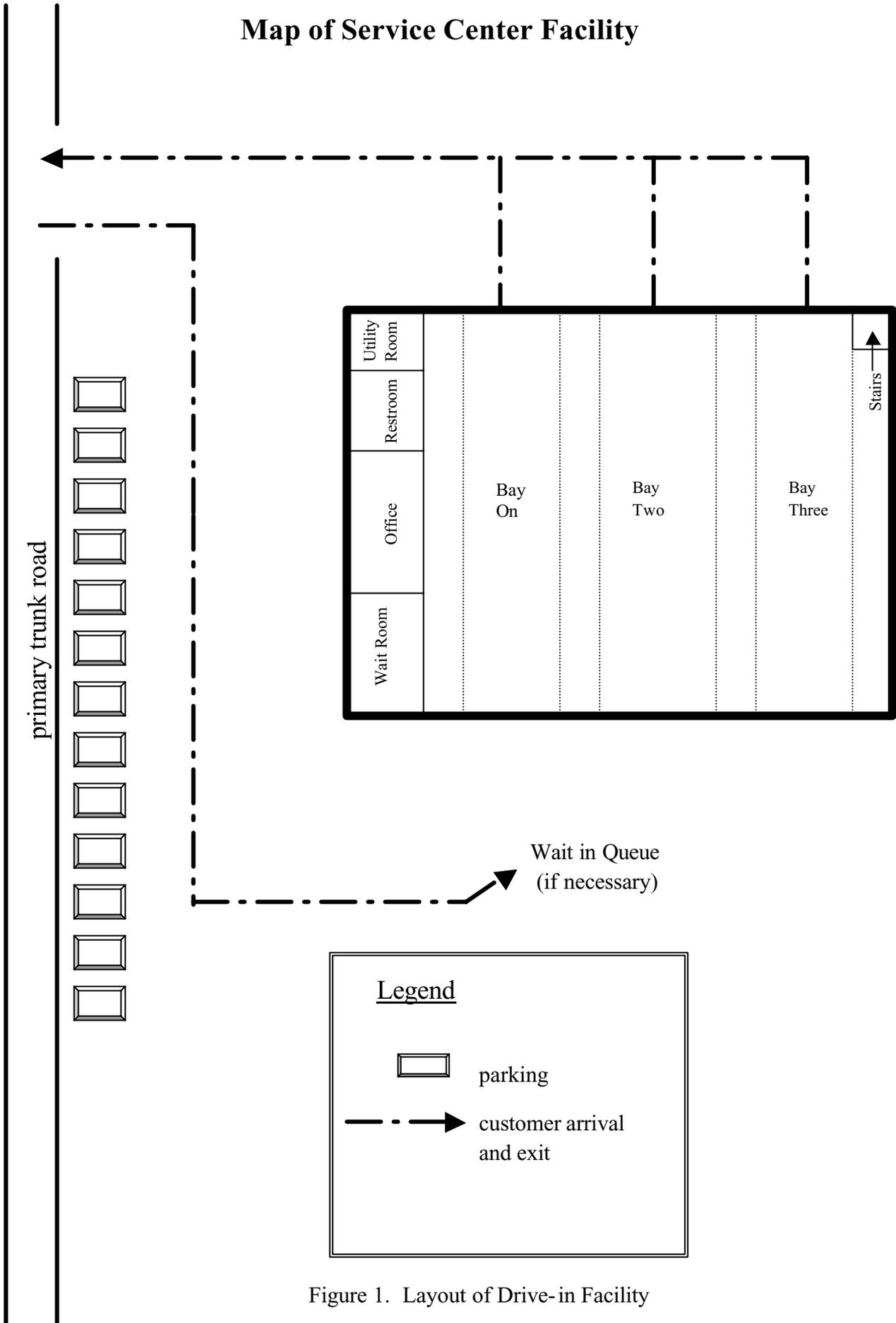


Figure 1. Layout of Drive-in Facility

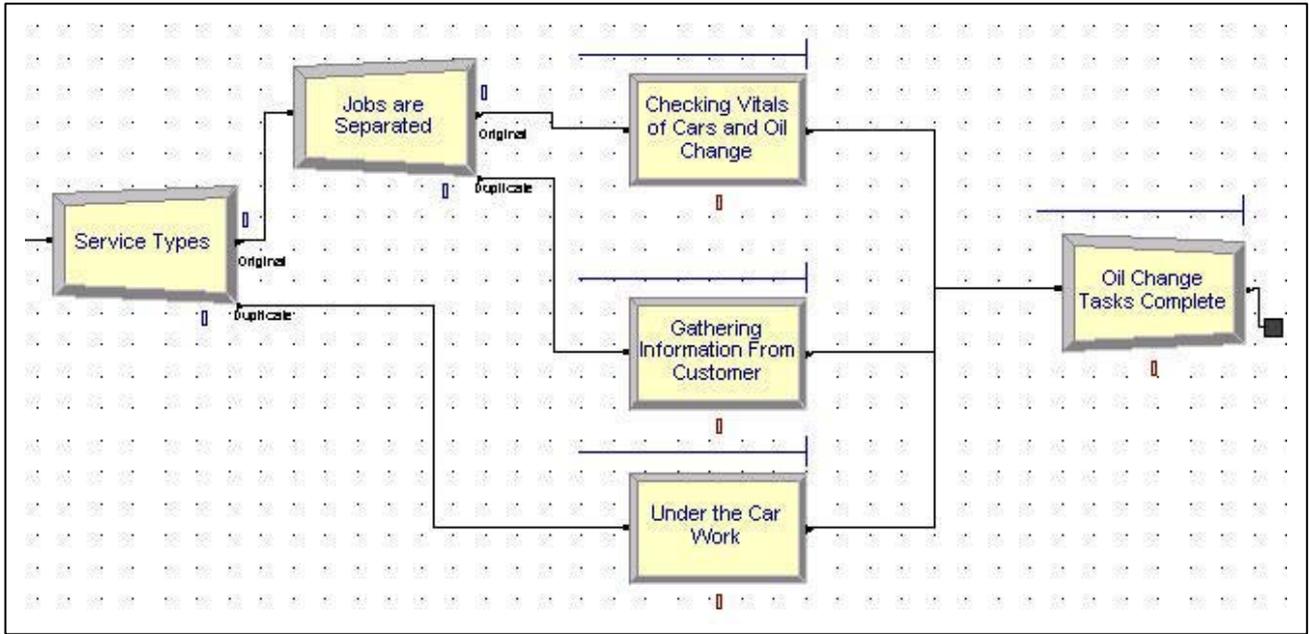


Figure 2. Screen Shot of Submodel

# NONLINEAR FUNCTIONAL APPROACH: PEOPLE BEHAVIOUR DESCRIPTION IN CASE OF EMERGENCY SITUATIONS

Boyko Ranguelov  
Hazard Assessment Unit  
Joint Research Centre, DG-EC  
E. Fermi str. 1, Ispra 21020, Italy  
E-mail: [boyko.ranguelov@jrc.it](mailto:boyko.ranguelov@jrc.it)

## KEYWORDS

Risk Analysis, Knowledge Discovery and Acquisition.

## ABSTRACT

Nonlinear functions are proposed as descriptive expressions of the social behavior of the affected groups of people in case of the extreme critical situations (generated by natural hazards, industrial accidents or terrorist attacks). Five functions are identified up to now, called respectively: "solidarity", "panic", "rumors", "media" and "enthusiasm". The systematic table is created reflected the average time duration, sensitivity of the respected function, its interruption/non interruption behavior and the possibilities of the respective process management. This is a new approach to present the social behavior of the affected groups of people during the extreme events using deterministic functions. The nonlinearity of the functions is clearly expressed. The possibility to use these functions in the risk analysis and the risk management practice of the extreme situations is a strong tool for the investigation of the possible influence of the people's behavior. The increased security and safety policy of the EU coincides with the targets of this research. The only graphic expressions of the functions are presented. The main difficulties obtained are the data collection and verification. The analysis performed shows that frequently the combinations of functions can appear making the decision-making and management processes rather complicated and difficult.

## INTRODUCTION

The development of a strict mathematical approach for the description of the social elements like social behavior and social movements started not very long time ago. Now this is a modern approach used for many social applications, even for the analysis of emotions (Swarts 2004). This approach is much more coherent with the modeling and the different trend analysis. To describe the personal/group behavior and/or to predict the big social groups behavior in case of extreme events is a difficult task (Blaikie et al. 1994). Many mathematical expressions have been built up frequently on different non-linear basis like the theory of chaos, turbulence, attractors, etc. Almost all approaches have been generated using as samples

many and different natural and/or technological disasters.

The target in this study is more practical and humanistic – using both - the technological/the mathematical booms, to try to represent mathematically (which means rather objective) the general human population social behavior in risky situations – e.g. natural hazardous events, terrorist attacks, industrial accidents, etc., which can affect relatively bigger groups of the society (Kovachev et al. 1997).

## COMMON CONSIDERATIONS

If such approach is effective – large possibilities appear using the recent technology and communications for the effective risk management of the crisis events like: human behavior in case of emergency situations in different cases (for example - teracts, industrial accidents, natural catastrophes, etc.). It is known that to find out and to apply (only on the intuitive basis and some personal field observations) the deterministic approach to the social behavior in the risky situation is a really hard task (ISDR 2002). It is important to mention that to collect real relevant data in such situations is really a very difficult, hard, frequently impossible task.

This is due to several reasons (objective and subjective):

- Relatively short time of the influence of the hazardous event and/or the threaten agent (if such exists and could be possibly triggered),(Kovachev et al.1997).
- More important safe-life actions then the data collection. (It is really impossible to "put" in such situations data collectors or interviewers). So, main sources for the relevant information are the investigator's location at the spot of the event and its impressions, occasional photos, pictures, movies, shots, etc. at the moment of the event.
- In the everyday practice the post-event analysis is targeted to discover the generator of the event, and the consequences (mainly the economic part), but almost never to follow the social momentum behavior of the people involved in the hazardous event.

On the basis of the general considerations, personal or shared experience of the “field” observations, data appearance in different sources, lessons learnt publications (NEDIES Reports 2001-2004), etc., to try to define reasonable mathematical functions (sometime even descriptive) of the social behavior of the groups of people involved in such emergency situations is the main task of this research. There are not so many options concerning the group behavior in the risky situations (Sundnes and Birnbaum 2003) – there were identified up to now – several – solidarity and enthusiasm, panic and rumors, mass-media behavior. Probably some others exist, but they could be deal with in the future.

All of the described functions have a common property – clearly expressed nonlinear behavior. This is due to the very complex and complicated factors acting during the emergency situation.

The main reasons considered, are the instability and sensitivity of the group behavior in the extreme situations – very small changes in the input situation could generate big amplitudes of the output function behavior.

## METHODOLOGY

Methodology is based on several assumptions and simple principles:

- Graphical expression of the functions based on the different case studies and general lessons learned.
- Time duration estimations of the functions behavior based on the physical measurements and considerations related to the emergency situations generated by the natural or man-made hazards.
- Estimations of the possibility for the situation management.
- The functions are considered valid only for the groups of people located in the epicenter of risky event (i.e. direct players).
- It is considered that different people react in different way in any risky situation.

## GRAFIC PRESENTATION AND DESCRIPTIONS OF THE FUNCTIONS

All derived functions and their presentations are represented as follows:

### The panic function:

This is a time-intensity function (Figure 1), impulsive, relatively short in time (seconds to minutes), generated always in population (group) society, when a serious threat appears. The initial event ( $e_0$ ) usually generates relatively slower increase of the intensity comparatively the next ( $e_1$ ,  $e_2$ , etc.). Each new event triggered more intensive and faster increasing of the function. Short time plateaus could be observed.

Decreasing phase is longer for each following event. Interruption of the function is possible. If the next event is closer in time domain and the effect of the previous event is still lasting, cumulative effect could be observed (case  $e_i$  on the graphic). Going very fast up to the initial phase, slowing down calmly. The trend analysis shows (dashed line), that trend has a slow increment to a certain moment and then going down with time development, but plateaus also could be observed.

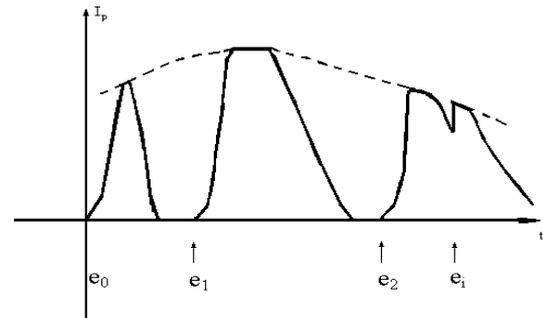


Figure 1: The “Panic” Function

### The solidarity function:

Usually long lasted intensity function (days – weeks). This is a probabilistic function (Figure 2), very sensitive to new threats, and very irregular, depending on the new threats – events ( $e_i$ ), after initial event  $e_0$ . The events  $e_i$  could be of different character – natural or man-made. Function increases very fast after the first event and have relative smooth plateaus. Any new threat leads to very fast decreasing (the “safe-life” syndrome), even interruption. The recovery time is also fast. The trend analysis shows slow decrease in time due to people accommodation to the threat, but this is not significant decrease.

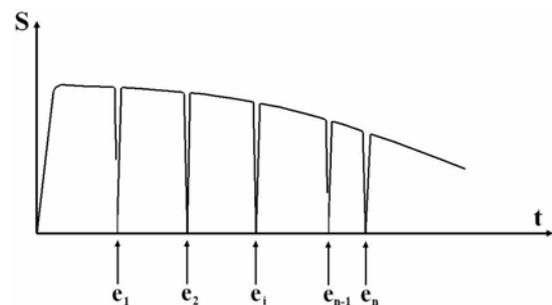


Figure 2: The “Solidarity” Function.

### The media function:

This is a cumulative probabilistic function, which describes media reflectance to the risky events and their consequences (by publications and other different forms and reactions – interviews, direct emissions,

public and/or expert opinions, public presentations, etc.). The function (Figure 3) is a long lasted function (weeks-months). Several similar, but always increasing functions, could be different for different mass-media – Radio (mainly news emissions – fastest and shortest resonance, sometimes (rear) short analytical notes) (curve No1); TV (also the shortest resonance, a little bit longer time coverage) (curve No 2); newspapers (longer stage) with some technological delay (with morning and evening editions) – sometimes analytical articles appear in longer time domain (curve No 3) – but sometimes could be more intensive); Internet – longest stage of reflectance – (curve No 4) with more analysis and opinions, but less primary information. All functions of mass-media behavior is going earlier or later to “saturation” (dashed line). The trend analysis shows faster increasing during initial phase and then slower decreasing. Interim events can trigger “bursts” of increasing number of publications, or fluctuations of the function. These functions usually have “maximum maximorum”, and significant fluctuations mainly in the decreasing part. This is one of few cases, when functions are relatively easily “measurable”. The representative data extract could be possible mainly in country (countries), where the hazardous event has occurred and has significant consequences.

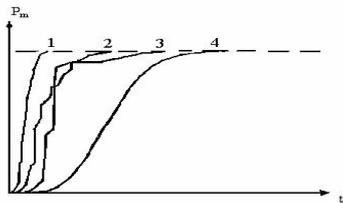


Figure 3: The “Mass-media” Function

**The “rumors” function:**

This is an intensity function, reflecting the generation and spreading of the rumors, which always appear in the society affected by the risk event (Figure 4). Usually this function started with some delay relatively to the initial event. Function is relatively long lasted – weeks to months (even years). The trend (dashed line) is faster increase at the beginning and slower decrease during the latest phase. Intermediate events can trigger the “bursts” of rumors, to create plateaus or local maximums (event  $e_i$  on the Figure 4). If no more events occurred function is going down slowly always. The function usually has the maximum maximorum. Collection of data is possible for this function sometimes. Number of different rumors could be collected, but they are not always reliable due to the collection difficulties and very fast deviations in content of the rumors. The problem of rumors spreading is very complicated and difficult to be

investigated. Generation of the “secondary” rumors, which consist some information of initial rumor, frequently have more “details” and sometimes deviate so much from initial “information” that’s impossible to recognize if it is the “new” or modified rumor. That’s why it is very difficult to calculate the original number of the rumors.

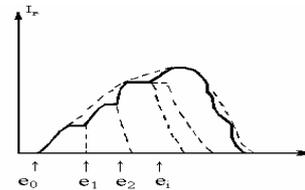


Figure 4: The “Rumors” Function

**The “enthusiasm” function:**

This is a relatively simple time-intensity function –  $I_e$  (Figure 5). Appears in relatively “better” social conditions, then other functions, because this is more or less a “post event” function. The “enthusiasm” itself appears usually amongst not affected by the hazard event groups of people or in some groups of people affected by the event (most frequently – not strongly), but in both cases among people willing to help the strongly affected groups of people. The function is usually long lasted (months even years) with sharper phase of increase, plateau and slow decrease in time. This function is practically not sensitive to the following events, but some fluctuations could be observed. Gives the possibilities for some quantitative measurements (for example by funds or different aids collection).

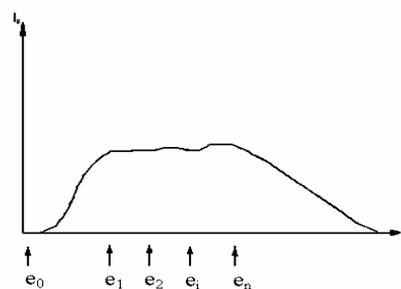


Figure 5: The “Enthusiasm” Function

All functions usually started immediately after the first event ( $e_0$ ), only the rumors and the “enthusiasm” functions could be shifted to the later time.

All described above functions have some common properties:

- Strong nonlinearity – this means that (as it can be seen on the figures) exponential and/or

logarithmic) behavior of the functions are the most frequent cases;

- Strongly event dependence – this means that each next hazardous event more or less affected the group’s behavior. This is the normal psychological reaction;
- Strongly affected group’s dependence – this means that the “soil is fertilized” by the initial event. This fact contains also the normal psychological reactions;
- The time durations are different, which is supported by many field observations and lessons

learnt analysis; sometimes the coincidence of the action of several functions could be expected. This is the most difficult situation for the risk management.

- The bursts of cases are considered, which means, that each case could be described by a separate function

### ANALYSIS

For the analysis following table (Table 1) has been created, summarizing the main properties of the functions. The most important of them are:

Table 1. Main properties of the suggested non-linear functions (in average).

Functions/ Properties	Time duration (minutes)	Sensitivity of the function	Interruptive	Manageable
“Panic” Function	seconds-minutes ( $10^{-1}$ -10)	high	yes	yes
“Solidarity” Function	days-weeks ( $10^3$ - $10^4$ )	high	yes	yes
“Media” Function	weeks-months ( $5 \cdot 10^4$ - $10^5$ )	low	no	not easy
“Rumors” Function	weeks-months ( $5 \cdot 10^4$ - $10^5$ )	middle	no	not easy
“Enthusiasm” Function	months-years ( $5 \cdot 10^4$ - $5 \cdot 10^5$ )	very low	no	yes

- The time duration – it is important to know the average time of the validity of the functions, as they are time dependant. This is an important property for the practical reasons. If in some hypothetical emergency situation there is a mixture of several (or all) functions, the time effectiveness is of a crucial importance for the management practice.
- The sensitivity – means that the function is strongly sensitive to the next dangerous events. An important property from theoretical and practical point of view.
- Interruptive (or non interruptive) – an important element. Any kind of functional analysis could be implemented during the “existence” phase of the function. This is an important property for the mathematical modeling, because any reliable solution could be obtained for the phase of existence of the respective function.
- Manageability - a very important property from practical point of view. It gives the level of the possibility to influence the function, using different measures.

The “sensitivity of the function” means sensitivity to the next following dangerous events with different (but significant) magnitudes.

“Interruptive” – means has (or not) points (phases) of interruption.

“Manageable” – means the possibility of management influence.

The “time duration” as average value is given in minutes.

The analysis shows that some difficulties could be expected in several directions. The main one is the data collection. In such situations to put the interviewers or data collectors using questionnaires, interviews, etc. is practically impossible. That’s why the comparative analysis between the theory verification and the practical realization is a significant difficulty. Always in such cases of the hazards influence the more important activities then the data collection exist (to safe the people lives, to protect the affected population, to perform rescue operations, etc.) Afterwards the influence of the event and the people’s behavior is forgiven and to reconstruct the people group’s behavior is very difficult. That’s why the verification of the approach is really difficult. Another difficulty could appear in case of the multiple risk situations. Most of the factors are acting simultaneously and the exact estimation of the intensity of the functions or their probabilistic behavior is difficult, even sometimes impossible.

## GENERAL RISK MANAGEMENT MEASURES USING THE FUNCTIONAL ANALYSIS

It is very difficult to take real measures for management of the people's social behavior in case of the emergency situations, because of the lack of real data. For our approach we use the analogy with the army and/or police groups for emergency actions, because these teams are frequently acting in similar environment:

- "Panic" function. The practice shows (according to the army, police, emergency teams trainings) that good training decrease the value of this function.
- The "Solidarity" (and the values of the respective function). It could be increased among the trained team, due to exercises in the real environment of similar (as in case of the disaster) conditions.
- The "Rumors" function is not easy to be managed. The commander's responsibility and the "strong hand" of the commander could decrease (but not to stop fully) the values of this function. That's why in the affected regions the organization based on highest single personal power and responsibility is of significant importance. All case studies show that if a good organization is created, the number of rumors decreases. One way to influence this function is also to provide widely the relevant and reliable information without delay.
- The "Media" function is hardly difficult to be managed. It is a very difficult task to manage the media response. Media always likes sensations and frequently do not reflect and interpret the available information in the proper way, creating sometimes panic, rumors or other people reflections, which do not help the proper management.
- The "Enthusiasm" function is the most promising function from the point of view of risk management practice. This is due to the positive circumstances existing in similar environment – people who like to support the affected people, cultural heritage, different structure reconstructions, etc. The response could be managed in long or short time intervals by good motivated and organized persons.

## CONCLUSIONS

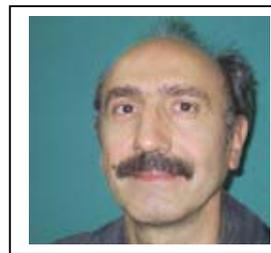
Several nonlinear functions are presented. They are used as formal mathematical descriptive elements about the group's people behavior affected by the emergencies of different origin. Their graphical expressions are suggested together with the analysis of the different expected cases. The functions are suggested on the base of field experience, lessons learnt, case studies and general considerations. The systematic table of main properties of these functions is created helping the analysis and the practical performance. Some general management measures are proposed, which could be useful in everyday management practice or in case of the mathematical modeling of such situations generated by natural hazards, terrorist actions of industrial accidents.

*Acknowledgments: This idea was developed during the sabbatical stay (2004-2005) of the author in the IPSC, JRC-EC, Ispra, Italy.*

## REFERENCES

- Blaikie, P., Cannon, T. Davis, I. and Wisner, B. 1994. At Risk: natural hazards, people's vulnerability, and disasters, London: Routledge., 213 pp.
- ISDR. 2002. Living with risk; a global review of disaster reduction initiatives. Geneva, United Nations., 73 pp.
- Kovachev, V. Rangelov, B. and Tsonchev, R. 1997., Risk Management of Natural disasters and catastrophes., NCDE, Sofia, , 220 pp.
- NEDIES Project Report 2002. Lessons learnt from Earthquakes, EU-JRC, IPSC, Ispra., 37 pp.
- NEDIES Project Report. 2003. Lessons learnt from Landslides, EU-JRC, IPSC, Ispra., 145 pp.
- Schwartz, B. 2004. The tyranny of choice. Scientific American, No 4., pp. 43 – 47.
- Sundnes, K. and Birnbaum, L. 2003. Health disaster management-Guidelines for Evaluation and research in the Utstein Style. Prehospital and Disaster Medicine Volume 17, Suppl. 3., 343 pp.

BOYKO K.RANGULOV was born in Sofia. Studied Geophysics (Mining and Geology University – MGU) and Applied Mathematics (Technical University). Now is leading the research on Seismology and Tsunamis in the Geophysical Institute of Bulgarian Academy of Sciences. Professor in two Universities – MGU and NBU (New Bulgarian University). For the last two years is a visiting scientist on natural hazards and risk assessment in the EC-Joint Research Centre in Ispra, Italy.



# COMPONENT-BASED COMPOSITION OF SYSTEM DYNAMICS MODELS

**Christian Bauer, Freimut Bodendorf**  
**Department of Information Systems**  
**University of Erlangen-Nuremberg**  
**Lange Gasse 20, 90403 Nuremberg, Germany**  
**{christian.bauer|freimut.bodendorf}@wiso.uni-erlangen.de**

## KEYWORDS

System Dynamics, Component Technology, Simulation Framework, Model Composition, Software Tool

## ABSTRACT

An approach using component technology for the development of System Dynamics models is introduced. At first a brief introduction of System Dynamics and its modeling elements is given. The representation of System Dynamics models in Vensim, an interactive software environment for the handling of such models, is described. On this basis the paper provides a concept for component development and model composition. The concept is basic and independent from any problem or domain specific context. An example depicts the appliance of the concept to a given problem domain explaining the usage of domain specific model design patterns and components. Finally a prototype system to support component-based modeling is described.

## PROBLEM

The use of component-based approaches lead to vast improvements in almost every engineering discipline. The list of success stories ranges from classical Industrial production to software development and service manufacturing. The advantages of component technology are well-known. Examples are increased reusability, reduction of complexity, encapsulation of expert knowledge, accelerated production, and the establishment of quality standards.

This paper introduces a concept for the component-based development of System Dynamics models and outlines the potential, that the adoption of component technology can bring to the field of System Dynamics modeling. So far, only few work has been done in this area (Eberlein 1996; Myrtveit 2000; Tignor 2001). Therefore, concepts for modeling components and software environments that support the component-based model composition are rare.

The paper starts with a brief introduction of System Dynamics in general in order to sketch the requirements for the conceptualization of a component-based modeling approach. As the software environment Vensim is used for implementation, its specific representation of System Dynamics is outlined. After that the modeling

concept and a prototype system supporting the composition process are described.

## SYSTEM DYNAMICS

System Dynamics is an approach that focuses on the analysis of the behavior of complex technical and socio-economic systems. By using simulation models it aims at explaining the system structure that causes the observed behavior. Therefore, the system is decomposed into appropriate elements whereby the causal relationships between the identified elements are revealed.

### Modeling Elements

System Dynamics uses mathematical models based on differential equations. Modeling elements are variables and relationships between variables. Relationships are represented by differential equations. Fig. 1 shows the notation that is usually used to visualize the model.

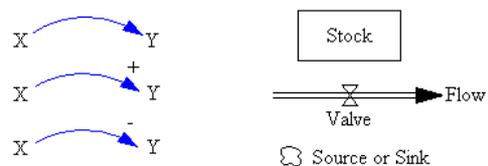


Figure 1: System Dynamics modeling elements

An arrow represents the relationship (causal link) between two variables. The dependent variable is placed at the head of the arrow. Thus, the direction of the relationship is defined. In figure 1 variable Y is influenced by variable X. The formal expression is  $Y = f(X)$ . Link polarities can be assigned to causal relationships. If  $dY / dX > 0$  applies, the link polarity is positive, marked by a plus symbol at the head of the arrow. If the link expresses the relationship  $dY / dX < 0$ , a minus symbol is used (Sterman 2000).

A variable that accumulates the influences it receives over time is referred to as a stock (see figure 2). A stock gives systems inertia and provides them with memory. The change of state that affects a stock at any point in time is described as a flow. The amount flowing in or out of a stock is controlled by a valve. Clouds are used to indicate that the source or the drain of such a flow is outside the model boundaries.

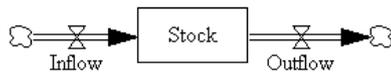


Figure 2: Stock and flow

The mathematical representation is:

$$\int_{t_0}^t \text{Stock}(t) = \text{Stock}(t_0) + \int [\text{Inflow}(s) - \text{Outflow}(s)] ds$$

The value of a stock is computed through integration of the differences between inflow and outflow at any point in time  $s$  between a starting point  $t_0$  and the actual point in time  $t$  under consideration of the initial state of the stock at  $t_0$  (Sterman 2000).

By combining these modeling elements the structure of the system to be analyzed is given. Feedback loops are in the core of the model development. Causal loops are the most important elements to define the behavior of the system. Decisions, intended to govern the system's behavior, are always part of a causal loop with that system (see figure 3).

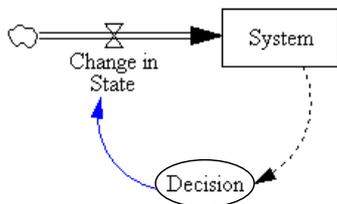


Figure 3: Decision as a feedback process

A decision is made by applying a set of decision rules to the system. Input is information about the state of the system. The decision changes that state of the system. Therefore a decision is part of a causal loop comprising the decision, the initiated change in state and the state of the system. Through simulation it is possible to analyze the behavior of the model over time and thereby approve the appropriateness of the applied decision rule (policy).

### Model Representation in Vensim

Vensim is an interactive software environment for the development, simulation, and exploration of System Dynamics models (Ventana 2003). In Vensim models are created either by a text editor or by a sketch editor. The text editor is a general-purpose ASCII-editor that allows the specification of the model's underlying variables and equations. The sketch editor on the other hand provides a graphical user interface to the modeling elements. No matter in which way a model is created, Vensim always stores the model data in a single file. Two basic file formats are available. The format .vmf stores model data as binary code while in .mdl-files the model data is stored as plain text (Ventana 2003). Figure 4 shows the structure of a simple System Dynamics model created with the sketch editor.

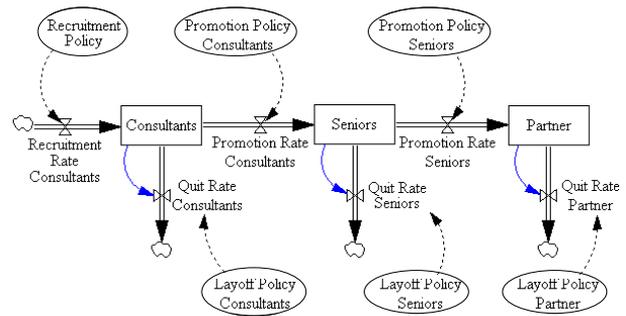


Figure 4: Simplified System Dynamics model capturing the workforce of a consultancy

The model captures the workforce of a consulting company and consists of three stocks, each representing a job level within the organization. The number of staff at each job level at any point in time  $t$  is computed through integration of the differences between inflow and outflow during the period  $t - t_0$ . Quit rates reduce both, the number of staff at the regarded job level as well as the total number of staff. Promotion rates in contrast shift staff from the preliminary to the next job level. Solely the recruitment of new consultants increases the overall number of staff and thereby offers the possibility to compensate fluctuation. The maximum promotion and quit rates depend on the value of the stock they are related to. Recruitment rate, promotion rate as well as part of the quit rate are subject to management policies that define the target values for these rates. In figure 4 variables that represent such policies are marked by a circle. The values of these variables are either subject to user parameterization or have to be determined through additional models that implement the guiding rules for these policies.

Opening the model in the text editor reveals the structure of the underlying differential equations. Figure 5 shows the model variables, equations, and simulation control parameters in the .mdl-file-format.

```

Consultants= INTEG (Recruitment Rate Consultants-Promotion Rate Consultants
- Quit Rate Consultants, 16) ~-|
Seniors= INTEG (Promotion Rate Consultants-Promotion Rate Seniors
- Quit Rate Seniors, 8) ~-|
Partner= INTEG (Promotion Rate Seniors-Quit Rate Partner, 2) ~-|
Promotion Rate Consultants= Promotion Policy Consultants ~-|
Promotion Rate Seniors= Promotion Policy Seniors ~-|
Quit Rate Consultants= Consultants*0.1 + Layoff Policy Consultants ~-|
Quit Rate Seniors= Seniors*0.1 + Layoff Policy Seniors ~-|
Quit Rate Partner= Partner*0.01 + Layoff Policy Partner ~-|
Recruitment Rate Consultants= Recruitment Policy Consultants ~-|
Promotion Policy Consultants = A FUNCTION OF () ~-|
Promotion Policy Seniors = A FUNCTION OF () ~-|
Layoff Policy Consultants = A FUNCTION OF () ~-|
Layoff Policy Seniors = A FUNCTION OF () ~-|
Layoff Policy Partner = A FUNCTION OF () ~-|
Recruitment Policy Consultants = A FUNCTION OF () ~-|
*****
.Control
*****- Simulation Control Parameters |
FINAL TIME = 10 ~ Year ~ The final time for the simulation. |
INITIAL TIME = 0 ~ Year ~ The initial time for the simulation. |
SAVEPER = TIME STEP ~ Year [0,?] ~ The frequency with which
output is stored. |
TIME STEP = 1 ~ Year [0,?] ~ The time step for the simulation. |
...

```

Figure 5: Model variables, equations and simulation control parameters

Each variable is defined by an equation that determines its value. Further it is possible to specify a dimension for the variable and to place some comments. The format has the following structure:

`<equation> ~ <dimension> ~ <comment> |`

The character “~” separates the elements of the definition, while the character “|” terminates the definition as a whole (Ventana 2003a). E. g., the first line of text shown in figure 5 defines an equation to determine the value of the variable “Consultants”. The number of consultants is computed as an integral of the difference between the recruitment rate and the sum of promotion and quit rates at each given point in time. It is defined that the initial value of “Consultants” is 16. The variable has no unit specification and no comments are given. Processing the text line by line is at first straight forward. Several variables are defined that determine the structure of the model. The definitions of variables representing management policies stand out. The right hand sides of these equations all contain the string “A FUNCTION OF ( )”. This string, a keyword of the Vensim Modeling Language, indicates that no equation is defined for the given variable (Ventana 2003a, p. 68). Thus the model of figure 5 is incomplete and therefore not ready for simulation. On the other hand this keyword provides an important starting point for the development of model components.

The definition of the model’s variables and equations is followed by a section that specifies the simulation control parameters “INITIAL TIME”, “FINAL TIME”, “SAVEPER” and “TIMESTEP”.

The last section of the .mdl-file-format contains so-called sketch information, that is the information needed to compute the graphical representation of the model’s structure. Figure 6 shows an excerpt of the sketch information needed to represent the exemplary model.

```

...
\\---// Sketch information - do not modify anything except names
V300 Do not put anything below this section - it will be ignored
*View 1
$192-192-192,0,Times New Roman|12||0-0-0|0-0-255|-1-1-1|-1-1-1|96,96,100
10,1,Consultants,302,230,40,20,3,3,0,0,0,0,0
10,2,Seniors,492,229,40,20,3,3,0,0,0,0,0
10,3,Partner,683,230,40,20,3,3,0,0,0,0,0
1,4,6,2,4,0,0,22,0,0,0,-1-1-1,1|(428,229)|
1,5,6,1,100,0,0,22,0,0,0,-1-1-1,1|(367,229)|
11,6,268,399,229,6,8,34,3,0,0,1,0,0,0
10,7,Promotion Rate Consultants,399,256,50,19,40,3,0,0,-1,0,0,0
1,8,10,3,4,0,0,22,0,0,0,-1-1-1,1|(616,229)|
1,9,10,2,100,0,0,22,0,0,0,-1-1-1,1|(554,229)|
11,10,332,583,229,6,8,34,3,0,0,1,0,0,0
10,11,Promotion Rate Seniors,583,256,50,19,40,3,0,0,-1,0,0,0
12,12,48,166,228,10,8,0,3,0,0,-1,0,0,0
...

```

Figure 6: Sketch data

The beginning of the sketch information is always marked by the string “\\---//” followed by a comment. The second line starts with a version code that indicates the format of the sketch information and is also followed by a comment. The third line names the view of the sketch and is preceded by the char “\*”. By the definition of multiple views it is possible to spread the graphical representation of the model over several

windows. Thus, exploration and handling of large models is facilitated. The line beginning with a “\$” sets the default font and colors of the view. The remaining lines define the objects appearing on the sketch. Each line defines one object following a special format. E. g., the first number sets the object type, the second number is the ID of the object. Objects that represent a variable contain the variable name, whereas an object that represents an arrow contains the ids of the two variables linked together (Ventana 2003a).

## MODEL COMPONENTS

A model component is a piece of model that can be used as a building block. Fully-fledged models are developed by linking several model components together. The concept of a model component corresponds in some way to the concept of a class known from object engineering. Thus, a model component is an artifact that encapsulates a well-defined structure. As it uses specified interfaces to interact with other model components, polymorphism is achieved, allowing the substitution of single components. Thereby a component-based model can easily be adjusted to solve a new or modified problem.

### Model Component Specification

The model component specification is independent from any problem or domain specific context and defines the general structure that has to be followed by every domain specific model component. Figure 7 depicts the definition of a model component using the notation of an UML class diagram.

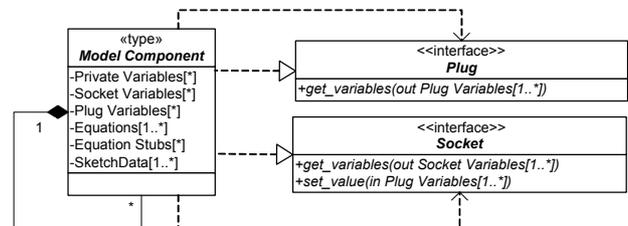


Figure 7: Model component specification

A model component is a piece of model, therefore it contains several model variables, equations and the corresponding sketch data. By aggregation it is possible to build model component hierarchies. The interaction between components is handled by the definition of interfaces ensuring the substitutability of single components. As shown in figure 7 a model component defines two interfaces, a plug interface and a socket interface. At least one of these interfaces has to be implemented, otherwise no interaction is possible.

The plug interface comprises a set of model variables called plug variables. Plug variables are used to connect to the socket variables of another component’s socket interface. In contrast to plug variables, whose values are defined through a corresponding equation within the model component, socket variables have only so-called equation stubs, marked by the keyword “A FUNCTION

OF ()". During assembly the equation stub of the socket variable is replaced by the equation of the corresponding plug variable. User-defined keywords are used to determine whether a model variable is a socket or a plug variable. E. g., for socket variables the string "\$SOS\$" might be defined, while for plug variables the string "\$PLS\$" might be suitable. To determine which plug variable implements a given socket variable, the names of corresponding plug and socket variables have to be identical, except their keyword. E. g., a plug variable intended to implement a socket variable named "Recruitment Policy Consultants \$SOS\$" must be labeled "Recruitment Policy Consultants \$PLS\$". Figure 8 illustrates the assembly process.

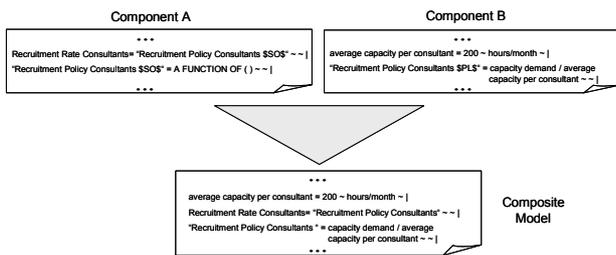


Figure 8: Model composition

Model component A contains the socket variable "Recruitment Policy Consultants \$SOS\$". This socket variable is used to determine the value of the variable "Recruitment Rate Consultants". As the socket variable only possesses an equation stub, no value can be computed. Component B defines the corresponding plug variable "Recruitment Policy Consultants \$PLS\$" and provides an equation to determine the variable's value. Once both components are assembled, the equation stub of the socket variable is replaced by the equation of the plug variable and the keywords "\$SOS\$" and "\$PLS\$" are

deleted to simplify variable names within the composite model.

A similar technique is used to allow multiple instantiation of a component. As variable names have to be unique, it is not possible to use the same variable name in two instances. To solve this problem, the string "#VAR#" is used as keyword in every model variable's name. During component instantiation this keyword is replaced by the ID of the created entity. E. g. the variable "Recruitment Rate Consultants #VAR#" of a given component becomes "Recruitment Rate Consultants 1" for the first instance and "Recruitment Rate Consultants 2" for the second instance. Thereby it is ensured that every variable's name remains unique.

### Derivation of Domain-specific Components and Model Design Patterns

Based on the model component specification, domain-specific components are derived. System Dynamics models are usually built to analyze a well-defined problem or phenomenon and provide insight into the inherent structure of the system under investigation. Within each problem domain certain domain-specific concepts exist and domain-specific components are used to capture this knowledge. Further domain-specific model design patterns can be created through the definition of explicit relationships between components. Thereby it is possible to predefine the structure of the prospective composite model, thus additionally guiding and facilitating the model building process. Figure 9 gives an example of a domain-specific design pattern that depicts the structure of a model capturing the collaborative relationships between consultancyancies.

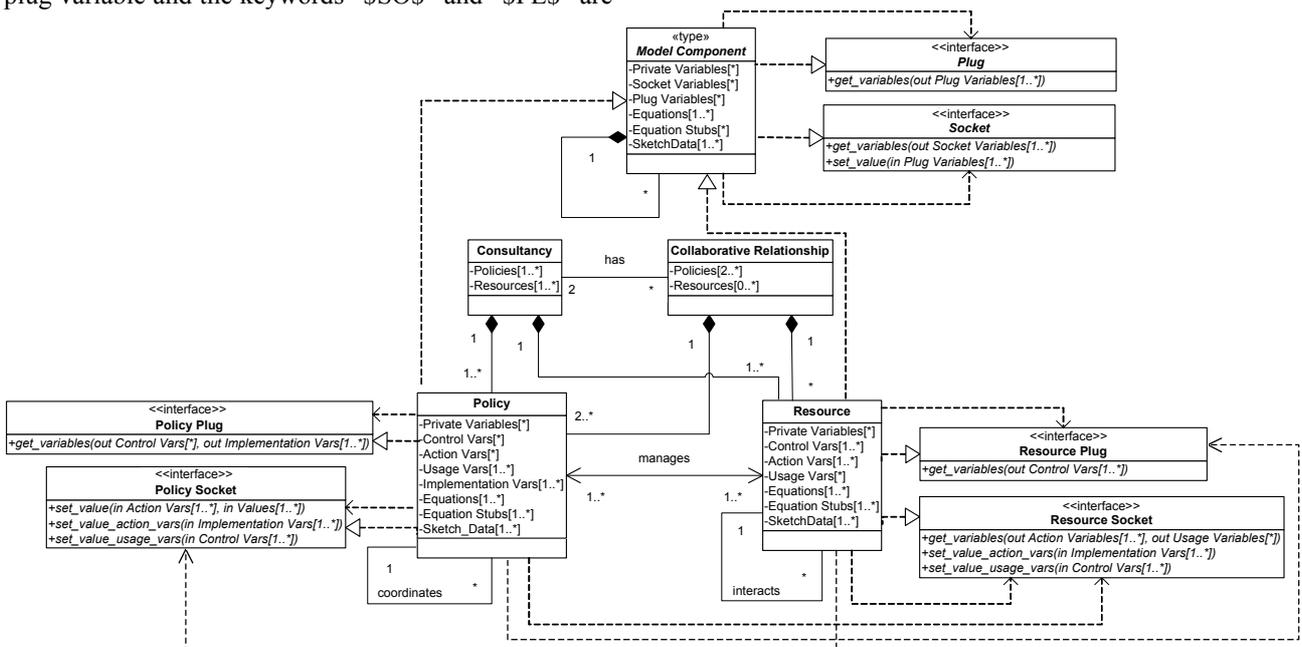


Figure 9: Domain-specific model design pattern including derived components

The design pattern defines two levels of abstraction and four types of components. On the first level the components “Consultancy” and “Collaborative Relationship” are defined. Both are aggregations of the components “Policy” and “Resource” on level two. As they merely act as containers, they don’t need to specify separate interfaces, but draw on the interfaces implemented by the components they include. Nevertheless the design pattern defines a relationship between both components, which specifies that two instances of “Consultancy” are needed for a collaborative relationship to exist. Further it is defined that a consultancy at least consists of one policy component and one resource component, while a collaborative relationship aggregates at least two policy components (one of each consultancy involved).

As figure 9 depicts, the components “Policy” and “Resource” implement the model component specification. Besides the necessary plug and socket interfaces they define several new variable types derived from the abstract model component. The declaration of new variable types is done through the specification of corresponding keywords. E. g. the component “Resource” contains private, control, action, and usage variables. Control variables (keyword “\$CV\$”) are used as plugs and determine which information of the model piece can be used by the sockets of other components. Action variables (keyword “\$AV\$”) and usage variables (keyword “\$UV\$”) in turn are socket variables which provide only equation stubs and thereby indicate where the building block depends on input from other model components. The methods of the resource socket interface specify that the value of action variables is set by implementation variables (keyword “\$IV\$”) while the value of usage variables refers to control variables. The mapping of variable types allows the specification of relationships between components. Thus the relationship “interacts” between resources is implemented by pairs of control and usage variables. The relationship “manages” between a resource and a policy comprises two aspects. First the policy has to define usage variables as part of its socket interface in order to access the resource’s control variables. Second it has to define one or more implementation variables which can be used as a plug to implement the resource’s action variables.

As a policy component may also define control and action variables itself, it is possible to assign the government of the policy to another, superior policy component. Thereby hierarchies of policies coordinating each other can be created (“coordinates” relationship between policies).

## MODEL COMPOSER

While Vensim provides an environment for the development of model components as well as for the simulation and exploration of the composed simulation models, it does not support the process of model composition itself. Therefore a tool has been developed, that on the one hand provides a graphical user interface

to facilitate the composition process and on the other hand helps to administrate the used model components, design patterns, and composed simulation models. The tool has been implemented using the Microsoft .NET Framework and the Dynamic Link Library of Vensim.

## Architecture

Figure 10 shows the architecture of the prototype.

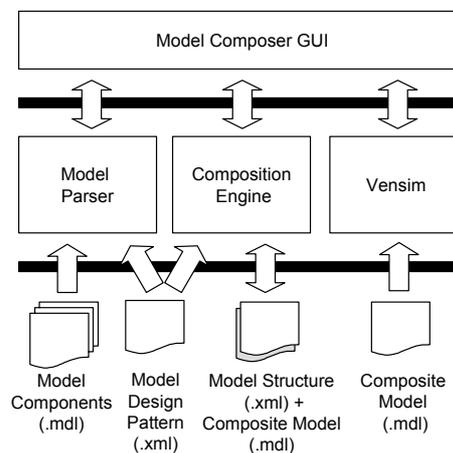


Figure 10: Architecture of the model composer

The application basically consists of three separate modules that are accessible through a common graphical user interface. The model parser is used to analyze the structure of a model component and to create a processable internal representation of it. As a model design pattern specifies the types of valid components and their relationships, it is necessary to load a design pattern into the parser first. After that the .mdl-files of model components are opened and the content is parsed according to the assigned design pattern. The parser uses the specified keywords to access the variable definitions of the pattern and thereby determines the types of the loaded components.

The composition engine implements the routines necessary to automate the assembly of the chosen components. To produce a composite model, the composition engine takes the internal component representations, combines them according to the relationships defined within the design pattern and creates a new .mdl-file for the composite model. Further a .xml-file is created to store the relevant composition information (e. g. the types of components and the number of instantiations). Thereby it is ensured that composite models can be decomposed at any time for further modifications.

In order to facilitate the creation and inspection of model components as well as the exploration and simulation of composed models, Vensim is integrated into the model composer. Thereby frequent manual switch-overs between both applications can be avoided and the usability of the system is enhanced.

## Composition Process

Figure 11 shows the graphical user interface of the prototypical model composer.

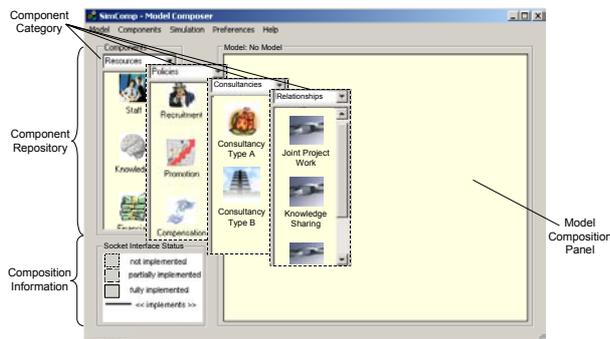


Figure 11: Graphical user interface of the model composer

Available Components are categorized according to their component type. As the exemplary model design pattern shown in figure 9 defines four types of components, the model composer provides four corresponding component categories. Within each category several components exist, that jointly make up the available component repository.

To compose a new model the user simply drags a component from a chosen category and drops it onto the model composition panel. The drag-and-drop-mechanism automatically activates the composition engine, which creates a new instance of the component (replacement of the keyword #VAR#). The composition engine scans the interfaces of all instantiated components on the panel and if a matching pair of socket and plug variables is detected, the involved components are linked together. Figure 12 gives an example of a model being in the composition process.

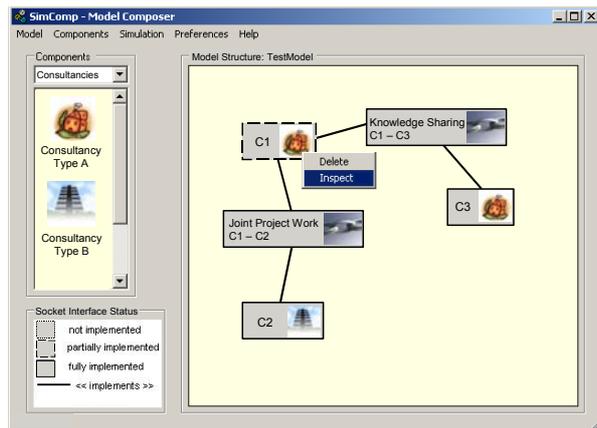


Figure 12: Construction of composite models

The model composition panel comprises two instances of the component “Consultancy Type A”, one instance of “Consultancy Type B” and two instances of relationships. The lines between the instances indicate the links detected by the composition engine. Further the implementation status of each component’s socket interface or, in case of containers, the aggregated status

of the socket interfaces of any child instances, is indicated by the type of border line surrounding the instance on the composition panel. A dotted border signals that no variable of the instance’s socket interface or of the socket interfaces of any children is implemented. This is the case when a new component is placed onto the panel, that has no links to already existing instances. A dashed border indicates that some of the socket variables are implemented through corresponding plugs, while a solid border states that the socket interface is fully implemented. If all instances show a solid border, the composition process is finished and the composite model is ready for simulation.

According to the applied model design pattern, the instances shown in figure 12 are all defined as containers. Thus they don’t define their own interfaces but aggregate the interfaces of the components they include. This means the detected links have their root at a lower level of aggregation. The dashed border of instance “C1” indicates that the container includes at least one component whose socket interface is not fully implemented. By selecting the menu item “Inspect” from the context menu of any instantiated component it is possible to explore its content. Figure 13 shows the content of the instance “C1”.

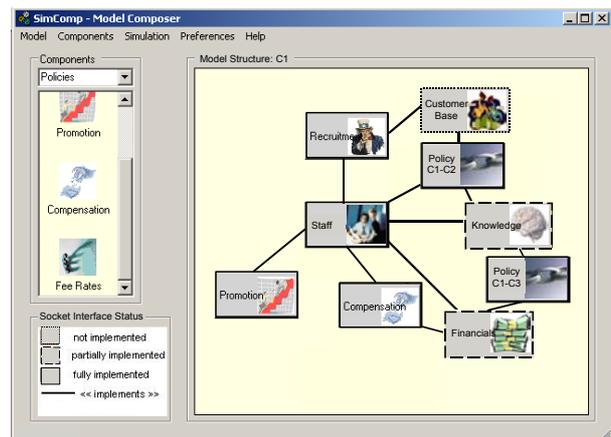


Figure 13: Content of instance “C1”

As defined in the model design pattern, components of the type “Consultancy” are aggregations of policies and resources. Therefore “C1” includes several instances of these two categories. As figure 13 shows, these instances are interrelated. E. g., the instance “Staff”, a component of the type “Resource”, is linked to the instances “Promotion”, “Recruitment” and “Compensation”, which are identified as “Policies” managing this resource. Apparently these three policies fully implement the socket interface of “Staff” (indicated by the solid border of the instance). In contrast to “Staff”, none of the instances on the panel implements the socket variables of “Customer Base” (dotted border). Further the socket variables of the resources “Knowledge” and “Financials” are only partly implemented (dashed border).

The instances “Policy C1-C2” and “Policy C1-C3” establish the links between “C1” and “Joint Project Work C1-C2”, respectively “Knowledge Sharing C1-C3”, which are both instances of the component category “Relationships” and therefore map the component type “Collaborative Relationship”. The applied model design pattern defines this component type as an aggregation of at least two policies, one of each consultancy involved (see section 3.2). This implies that “Policy C1-C2” and “Policy C1-C3” are simultaneously part of two aggregations, “Consultancy” and “Collaborative Relationship”. Nevertheless the inspection of “C1” only reveals the links to components that are embraced by “C1”. To inspect the links of “Policy C1-C2” and “Policy C1-C3” relating to components embraced by “Joint Project Work C1-C2” respectively “Knowledge Sharing C1-C3”, it is necessary to explore the structure of these aggregations.

Although the type of border line of an instance indicates the status of its socket interface and the lines between instances depict their interrelations, it is useful to provide more detailed information about the implementation and usage of the interface variables involved. Selecting “Inspect” from the context menu of a component that is not a container displays its interface variables categorized by type. Further the relationships of these variables to corresponding plug, respectively socket variables of other components are shown. Figure 14 gives an example.

Control Vars	Usage
Maturing Consultants #1	Promotion
Backlog promotable Consultants #1	Promotion
Maturing Seniors #1	Promotion
Backlog promotable Seniors #1	Promotion
Consultants #1	Knowledge, Compensation
Seniors #1	Knowledge, Compensation
Quitting fraction promotable Consultants/y #1	Recruitment
Quitting fraction promotable Seniors/y #1	Recruitment
Layoff Rate promotable Consultants #1	Recruitment
Layoff Rate promotable Seniors #1	Recruitment

Figure 14: Interface inspection of instance “Staff”

The figure shows the inspection of the instance “Staff”. According to the specification of the component type “Resource”, the interface variables are defined as “Action Vars”, “Control Vars” and “Usage Vars”. Selecting the tab “Control Vars” returns a list of all model variables marked as “Control Vars”. Further the column “Usage” shows, which instances define socket variables that use a certain control variable as plug. E. g., the variable “Maturing Consultants #1” is used by the instance “Promotion”, whereas the variable “Consultants #1” is used by “Knowledge” and “Compensation”.

Figure 15 shows the composed model. The solid borders of the component instances indicates that all

interfaces are fully implemented and the model is ready for simulation.

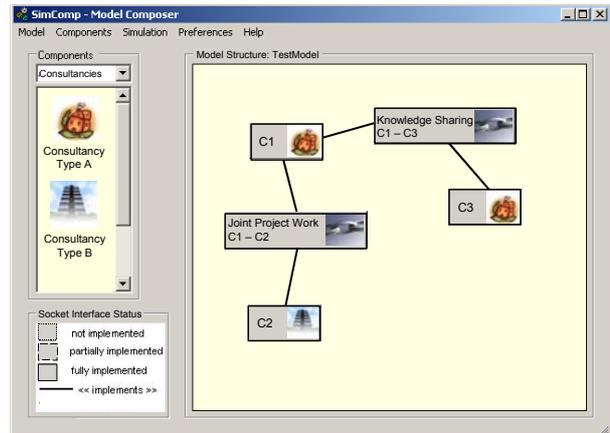


Figure 15: Composed model

Selecting “View Model Structure” from the menu item “Model” calls the Vensim environment to inspect the structure of the composed system dynamics model. The ability of Vensim to handle multiple views within a model is used to keep the graphical representations of the component instances separated. The usage of multiple views reduces complexity and facilitates model navigation. Figure 16 shows the component instance “Joint Project Work C1 - C2” in Vensim.

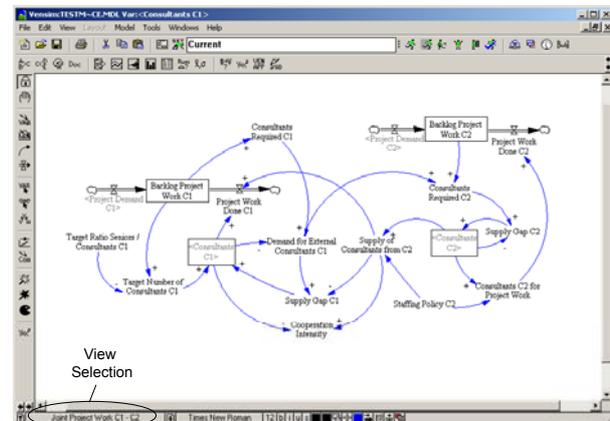


Figure 16: View of “Joint Project Work C1 - C2”

Simulation and analysis of the composed System Dynamics model are done in Vensim. Vensim provides a comprehensive set of tools to analyze the model structure as well as the simulation output. Parameterization allows the setup of different scenarios. Traditionally scenario development is restricted to the structure of the underlying model. The proposed component-based modeling approach facilitates model modification and thereby enhances scenario building to include alternative structures. This aspect is especially important in problem domains that deal with structural changes, such as collaboration management and business networking.

## CONCLUSIONS

A component-based modeling approach to the development of System Dynamics models basically has two major advantages. The first advantage is obvious. The reuse of existing model components facilitates the model building process. As modeling effort is reduced, the building process is accelerated. Further more, the quality of developed models can be enhanced. The expertise of subject matter experts is encapsulated in design patterns and model building blocks that can be easily composed to solve problems within a certain domain. The second advantage is that the component-based modeling approach can be used to improve the structural flexibility of System Dynamics models as such. The multiple instantiation of a component allows the handling of redundant structures, thus paving the way for the distinction and explicit modeling of individual elements, a privilege predominantly unknown to the System Dynamics approach.

## REFERENCES

- Eberlein, R. and J. Hines. 1996. "Molecules for modelers". In *Proceedings of the International System Dynamics Society*. Cambridge: System Dynamics Society.
- Myrtveit, M. 2000. "Object-oriented Extensions to System Dynamics". In *Proceedings of the International System Dynamics Society*. Bergen: System Dynamics Society.
- Sterman, J.D. 2000. *Business Dynamics - Systems Thinking and Modeling for a Complex World*. Boston: McGraw-Hill.
- Tignor, W.W. and M. Myrtveit. 2001. "Object-oriented Design Patterns and System Dynamics Components". In *Proceedings of the International System Dynamics Society*. Atlanta: System Dynamics Society.
- Ventana Systems Inc. 2003. *Vensim 5 User's Guide*. Harvard: Ventana Systems.
- Ventana Systems Inc. 2003. *Vensim 5 Reference Manual*. Harvard: Ventana Systems.

# TOWARDS E-GOVERNMENT: BUSINESS RENOVATION OF PUBLIC SECTOR IN SLOVENIA

Ales Groznik and Dejan Vivic  
University of Ljubljana, Faculty of Economics  
Kardeljeva ploscad 17, SI-1000 Ljubljana, Slovenia  
e-mail: ales.groznik@ef.uni-lj.si

## KEYWORDS

E-government, business renovation, business process simulation.

## ABSTRACT

E-government, as a segment of e-economy, has attained much interest in order to improve communications and level of services between governments and citizens. It is clear that successful e-government implementation requires not only introduction of modern information technology, but also business renovation, business process reengineering and e-business strategy. The main goal of the paper is to present the characteristics of business renovation efforts and readiness for e-government in Slovenia. The case of business renovation project in one of the Slovene Ministries, where the process modelling, analysis and simulation were extensively used is shown. The simulation modelling proved useful since it shows the process as a whole, drawbacks of the existing process, bottlenecks in the process execution, provides critical insight into process execution etc. The results of the business renovation show in reduced execution times, organizational changes, simplified business processes which represent a good foundation for informatization as a next step towards e-government.

## INTRODUCTION

E-government is the execution by electronic means of interactive, inter-organizational processes and represents a shift in business doctrine that is changing traditional organizational models, business processes, relationships and operational models that have been dominant in the public sector in the past decades. The new doctrine of e-government requires organizations to integrate and synchronize the strategic vision and tactical delivery of services to its clients with the information technology and service infrastructure needed to meet that vision and process execution. In the next few years, successful countries will restructure their public sector, process and technology infrastructure for successful e-government execution.

Past experience in introducing e-government in the most developed countries (Singapore, Canada, Australia, New Zealand...) in this field has shown us that the root of the problems, which have to be solved in introducing e-services, has moved from the technological into the organizational and process domain. The essence of e-

government is to radically change the ways and mechanisms of operating administration and, as a result, also basic principles, on which these mechanisms have been developing in the last decades or even centuries. Therefore, the business renovation (BR) or business process renovation methods should be used in the framework of e-services introduction. BR integrates radical strategic method of Business Process Reengineering (BPR) and more progressive methods of Continuous Process Improvement (CPI) with adequate Information Technology (IT) and e-business infrastructure strategies. Process renovation is a re-engineering strategy that critically examines current business policies, practices and procedures, rethinks them through and then redesigns the mission-critical products, processes, and services (Prasad 1999). Beside business processes reengineering business renovation also includes changes in human resources, culture, technology and organizational structure.

BPR as a segment of BR is a fundamental redesign of a companies business processes and organisational structures in order to achieve dramatic improvements in its critical success factors – quality, productivity, customer satisfaction and time to market etc. (Tapscott and Caston 1993). Because of its great intervention in the organization, BPR demands full co-operation of management team and workers. BPR is based on critical evaluation of existing processes in the organization (AS-IS business models), from which alternative (TO-BE business models) are made. Understanding of the existing processes is the key to successful modeling of renovated processes.

After the processes are optimized and renovated, the suggestions for their informatization need to be prepared. Informatization presents general and holistic implementation process and use of informatization technology which can be compared by analogy to industrialization process of industrial society. One of the main purpose of informatization is economic competitiveness achievement or automatisisation and optimization of their business processes.

In the article is presented business renovation process at the Ministry of education, science and sport. In Section 2 the e-government strategy in Slovenia is explained, which is recently actual topic in Slovenian political place. Then, in the Section 3, the theoretical starting-point of business renovation is described and it includes section about Business Process Reengineering and Informatization. Section 4 is the main part of the paper

and includes practical example of business renovation project at the Ministry of education, science and sport. It describes the main phases of the project and the key processes, which were identified. Then the modeling, business renovation and guidelines for informatization are discussed of the process Promotion of the employees in education to a higher professional title are presented.

## THE E-GOVERNMENT STRATEGY IN SLOVENIA

By adopting the "Strategy of E-commerce in Public Administration for the Period 2001-2004, SEP-2004" (Government Centre for Informatics, 2001), in February 2001, the Government of Slovenia has defined the primary strategic orientations for the next essential phase of informatization of public administration, which is the development of e-government. As a result, Slovenia is following a number of most developed European countries, which are approaching the accelerated development of e-government in a similar way.

Although Slovenia has started a new developmental cycle of technological modernization of administration and has launched a number of new projects, the conclusion was that development is not progressing as planned and expected. This is not only a problem in Slovenia, but based on analyses carried out in EU, also a problem in mostly all other countries. Due to the lack of experience in most cases, plans and deadlines for introducing e-government were in all places too optimistic. After a year or two, it can be seen that in most countries it was relatively easy to achieve the first (information) stage, which refers to the introduction of information services, as this step does not require specific changes in internal operations of administration and in business processes and procedures (Government Centre for Informatics 2001). Much more complex is the introduction of more demanding, so-called transaction services, which enable all phases of a selected administrative procedure or process to be executed electronically. As a rule, this requires a complete renovation of administrative operations, internal business processes and procedures, the integration of registers and public databases, the alteration and completion of material legislation and the development of new organizational regulations, classifications, and standards. At this point, the development of e-government in most developed countries has come to a standstill, which is evident from viewing web portals of these countries where it can be found very little transaction services. The same has also occurred in Slovenian public administration.

Problems, which need to be solved as soon as possible, are, in a minor sense, of technological nature (Government Centre for Informatics 2001). They predominantly extend to the internal renovation of administration operations, its reorganization, greater process orientation and close coordination and cooperation among various departments, and even branches of power (executive, legislative, and also

judicial). It has to do with deep structural changes in the operation of administration, which will be successfully and quickly implemented only with a total and well-considered approach, as used in the modernization and reformation of administration up to the present. BR projects should be focused on all related key business elements: business processes, people and finally the technology. E-government is not only enabling the redesign of internal organizational processes, but is extended into inter-organizational processes.

Within the framework of development of a new "organizational paradigm", which will be based on the operation of e-government, all State Bodies and other institutions from the public sector will have to analyze in detail all (action and other) administrative procedures and processes and renovate them in accordance with defined starting points and principles of development of e-government, and the possibilities that information technology can offer as soon as possible (Government Centre for Informatics 2002).

## BUSINESS RENOVATION

Reengineering a company means tossing aside all systems and starting over. It involves going back to the beginning and inventing a better way of doing work (Hammer and Champy 1993). Towards Leavitt's diamond (Burke and Peppard 1995) BR includes changes in technology, human resources, structure, culture and processes.

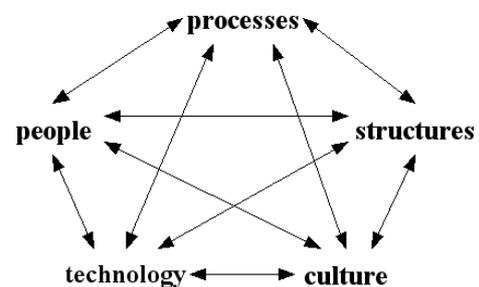


Figure 1: Amended version of Leavitt's 'diamond'

In the paper the orientation is on optimization of business processes and their informatization.

## Business Process Reengineering

BPR is an organizational method demanding radical redesign of business processes in order to achieve better efficiency, quality and more competitive production (Hammer and Champy 1993). It is also a method of improving the operation and therefore the outputs of organization (Kettinger and Grover 1995). It means analyzing and altering the business processes of the organization as a whole. BPR was first introduced in a research program at MIT (Massachusetts Institute of Technology) in the early nineties. BPR was the buzzword of the mid-1990s, and although there were plenty of successes, there were many more failures

(Hammer and Champy 1993). To many, BPR remains a dirty word, bringing back memories of head count reductions, budget cuts, facility closures, expensive consulting engagements and endless reorganizations that destroyed morale and confused employees, partners and customers. By the time it was recognized that successful BPR required careful change management, the damage was done. The BPR craze encouraged organizations to focus on internal process and internal (mostly transactional) ERP applications. Today, the e-business craze has reinvigorated interest in process, this time on a grander scale that spans organizations. The difficulties of formulating and adopting new process, a lack of cooperation between vendors, and the sheer difficulty of interorganizational coordination will likely lead to yet another era: the era of e-business.

Many leading organizations have conducted BPR in order to improve productivity and gain competitive advantage. A study by Dhaliwal (1999) showed that about 50% of firms surveyed in Singapore (in some cases comparable to Slovenia) were engaged in BPR projects, with 37% of the firms indicated their intention to take up BPR projects in next few years. However, regardless of the number of companies involved in re-engineering, the rate of failure in re-engineering projects is over 50% (Hammer and Champy 1993). Some of the frequently mentioned problems related to BPR include the inability to accurately predict the outcome of a radical change, difficulty in capturing existing processes in a structured way, shortage of creativity in process redesign, the level of costs incurred by implementing the new process, or inability to recognize the dynamic nature of the processes.

### **Informatization**

The goal of information engineering is to describe an already-conceptualized process in informational (or, more accurately, data-oriented) terms so that a system can be rapidly and rigorously constructed to support the new process design (Davenport 1993).

Information technology refers to the technological side of an information system. It includes the hardware databases, software network and other devices. It can be viewed as a subsystem of an information system (Turban et al. 2002).

Informatization and information technology are powerful tools for enabling and implementing process innovation. Although it is theoretically possible to bring about widespread process innovation without the use of IT, we know of no such examples (Davenport 1993).

Davenport (1993), in arguing for radical change rather than incremental change of business renovation, suggests that this is the only means of obtaining the order-of-magnitude improvements necessary in today's global marketplace. He is also saying that information technology is both an enabler and an implementer of process change.

The term of electronic business (e-business) presents from the business renovation view for organization new challenge, full of radical changes. E-business presents

radical move and consideration about business doctrine. From organization a new doctrine of e-business demands accommodation and synchronization of its strategic vision and its practical execution with opportunities of contemporary information technology (Groznič and Kovačič 2002).

### **BUSINESS RENOVATION PROJECT AT THE MINISTRY OF EDUCATION, SCIENCE AND SPORT**

The Business renovation project at the Ministry of Education, Science and Sport (Ministry) started due to internal and external factors. Internal factors that caused business renovation were the integration of two ministries, Ministry of Education and Sport and Ministry of Science and Technology into the Ministry of Education, Science and Sport, versified business processes that were not well defined and duplication of activities. Externally, the project has been stimulated by the Slovenian government that started the anti-bureaucratic program on the governmental level. The goal of the program is, according to Action Plan E-government Up to 2004 (Government Centre for Informatics 2001), to remove inefficiencies in business processes, to change organizational structure and to introduce suitable information technology that will support renewed business processes.

The Business renovation project project has three main phases:

- identification of key business processes and their modelling;
- analysis of key business processes on the basis of their models;
- modelling renewed processes and proposing organizational changes.

The project started with formation of project group consisted by members from the Ministry and consultants from Business Informatics Institute (BII), Faculty of Economics, Ljubljana. Then a workshop for Ministry project group was prepared in which they were acquainted with project goals and the methodology. After the workshop, five key business process groups were identified by discussion and brainstorming:

- strategic planning;
- working program preparation;
- laws and provisions preparation;
- financial processes;
- administrative processes.

The processes were modelled by interviewing people from the Ministry who perform the activities. This phase of the project was very difficult and lasted for almost six months and models had to be changed several times. Then, the members of Business informatics institute made analysis of key business processes on the basis of their models. The results of the analysis were the starting-point to renovated business processes, which were made in two months.

Since the scope of the project is too big for the presentation in the paper, only a fragment, subprocess Promotion of the employees in education to a higher

professional title of Administrative processes at General Affairs and Human Resource Service, will be shown in the next section.

### **Modeling and analysing the existing processes**

The Administrative processes group includes some of the most frequently executed processes and are therefore very interesting for a detailed examination and analysis in the BPR and informatization project as significant improvements in efficiency can be expected.

This group consists of more than 30 processes, however some of them are of the same type, but for different areas (e.g. elementary schools, high schools, universities) and therefore their substantial activities are executed in different departments.

In the first phase of the analysis some processes with the highest application frequency were examined in more details. One of them is "Promotion of the employees in education to a higher professional title" (Figure 2) which has about 2500 applications per year. The rate of complete application is 60%, after the completion of incomplete application this rate is 80%. The owner of this process is the General Affairs and Human Resource Service (GAHRS), where the application is professionally executed by four officers. The applications are always accepted only in dispatch centre. The application state is recorded four times, always twice: manually and using a computer program. The Minister signs the decision statement.

The simulation of the process that was carried out showed that the mean execution time for one application is 49 days. The effective work time is less than one day. The rest of the time is the delay in the process (signing,

transfers of documentation, waiting for the completion of the application etc.).

However, the quantitative results of the simulation experiment as presented in the simulation report, regardless of how precise and detailed the simulation may be, are only one aspect of the business process analysis. Business process maps themselves can frequently show many problems that have not previously been observed. In the modelling phase, several problems were observed. Beside the problems with data collection presented in the previous section there are also some difficulties related to the tool as not all the situations from the real world can be directly modeled. Some examples are (Tarumi et al. 2000):

- Process flow can be interrupted by other predominant processes.
- Multiple processes compete for a common resource.
- Many other kinds of exceptions can occur, such as the absence of personnel.
- Human behavior cannot be predicted (e.g. some persons start tasks as late as possible to meet the deadline).

Due to stated problems, the results obtained when using simulation modeling of business processes should be used cautiously, as the figures cannot be considered exact values. As such, its primary use is in analysis and in understanding the process itself, in observing the problems in process operation (e.g. bottlenecks), in evaluating and comparing alternative scenarios, in supporting decisions on process informatization, renovation, and in the introduction of organizational changes, etc. According to (Bellinger 2002) modeling and simulation is a discipline used to promote a deeper and more complete understanding of how things work; it does not provide answers.

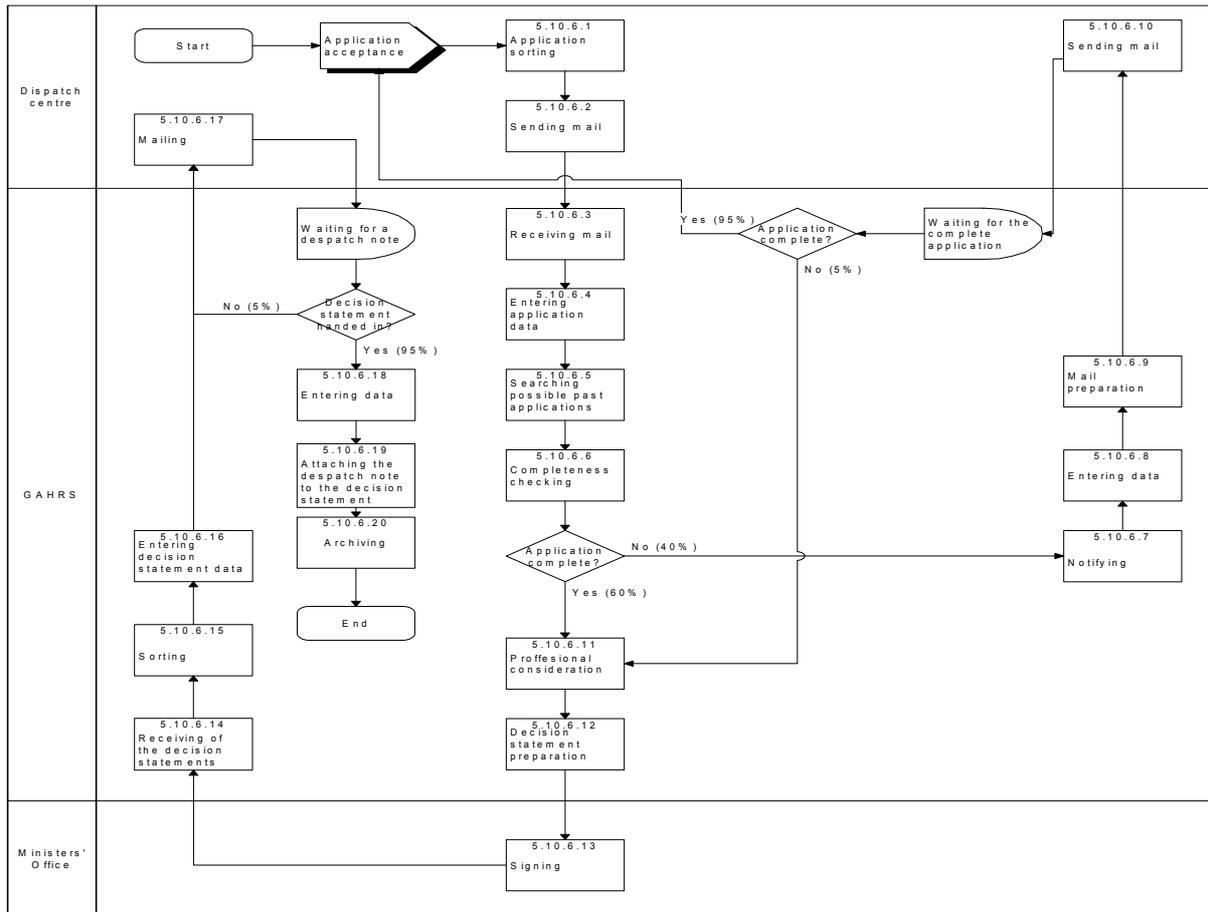


Figure 2: Promotion of the employees in education to a higher professional title

### Business Process Renovation and Informatization

To uniform the administrative processes current similar processes had to be grouped:

- statuses of associations, private workers, professional workers, professional and top sportsmen, organisations, Interscription in evident of sports objects, Fulfilment conditions for financing private kindergartens,
- agreement to the act of establishment of organization from the area of upbringing and education, science, sport,
- nostrification of the certificates acquired abroad,
- doing technical exams of professional workers, which are working on the area of upbringing and education in sport,
- second stage processes.

For each group, TO-BE models have been modelled. As the number of the processes is extensive, the focus will be on Promotion of the employees to a higher professional title (Figure 3).

The main goals of the renovation of the process Promotion of the employees to a higher professional title were:

- the procentage rate of the incomplete applications should be reduced,
- the execution time of the process should be decreased,

- information level of the customers should be improved,
- the applications tracking should be organized more efficient,
- the officers in GAHRS need to be discharged of administrative work,
- improvement of the applications recording.

During the analysis a high rate of incomplete applications was detected. The reason for that were mostly uninformed customers which did not know what should the application contain. The answer to that problem is in initiation of the Acceptance office, where customers could get all the information, needed to correctly fulfil the application (Figure 3 - Application preparation advising). The main tasks of the Acceptance office are formal completeness checking, giving informations about specific Administrative process and advising.

In the case of incomplete application the Officer from GHARS sends a request to complete the application. That would not be always necessary if they had an official database where they could find the missing information. That kind of database would increase customer satisfaction and decrease the execution times. In cases where sending request to complete application is necessary the Officer should use the fastest way. For that reason renovated process includes proposal of electronic mail and electronic signature.

The simulation results of the existing process showed that a lot of time is used for signing prepared decision statement in Ministers' Office. To decrease the execution times of the process and to discharge Minister, the activity signing decision statement should be executed by the Head of the General Affairs and Human Resource Service (GAHRS).

Business renovation is successful only in connection with informatization of business processes. The information support needs to be uniformed and connected for entire Ministry. The most important

information solutions on the process Promotion of the employees to a higher professional title are:

- the accompaniments of the procedure which works as a Workflow management System and enables uniform launching of activity and process execution,
- Document Management System which allows scanning of documents and electronic approach to archives and is connected to Workflow Management System,
- uniformed management of evidents,
- delivering applications through internet which must use digital confirmation and electronic signatures.

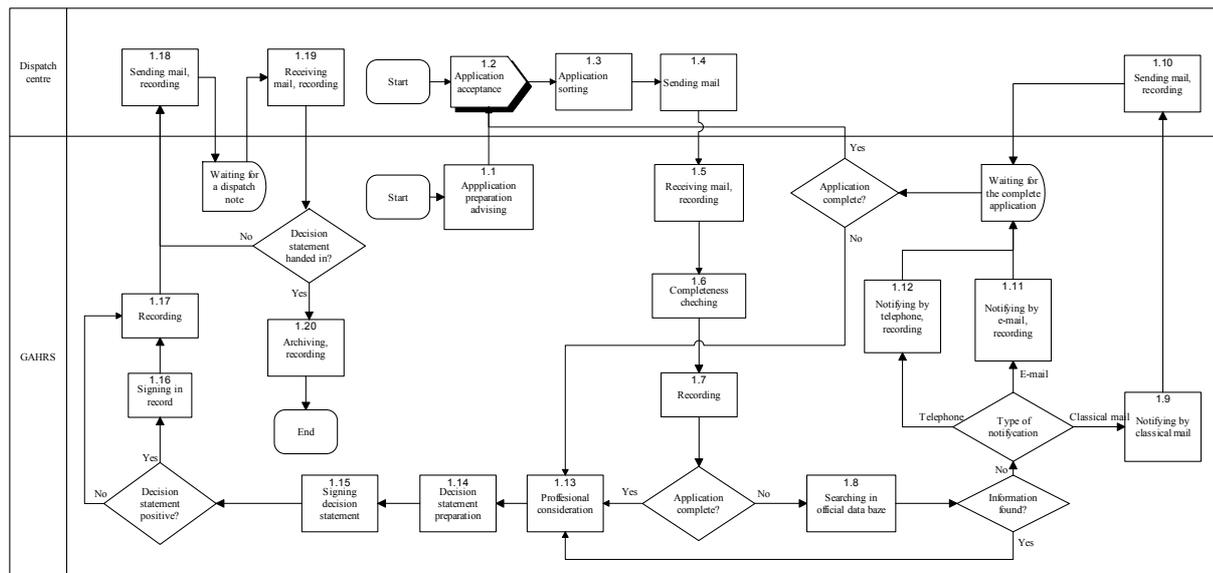


Figure 3: Renovated process Promotion of the employees in education to a higher professional title

## RESULTS

TO-BE model of the process Promotion of the employees in education to a higher professional title includes changes based on the main goals of the renovation.

The first difference is seen in implementation of the new activity Application preparation advising. On the base of appreciation 80% of all incoming applications go through this activity and this is why there are only 10% of incomplete applications instead of previous 40%.

Activity Searching in official database is also new in the process. Its existence affects on the number of demands about missing data, which becomes with renovation reduced. In renovated process, there are three ways of notifying customers about incomplete application. Officers can notify customers by phone, by mail and e-mail. According to implementation of informatization in the renovated process, notifying by e-mail must be used as often as possible and only exceptional cases are executed by the two other ways.

In the AS-IS model the decision statements are signed in Ministers' office which increases time of process execution. With renovation this activity is placed in

GAHRS and the decision statements are signed by state undersecretary.

The main result of the renovation is seen in simulation of the renovated process were the average execution time is 22 days shorter.

## CONCLUSION

In the paper are presented the main characteristics of business renovation efforts in the Ministry of education, science and sport. The results of the business renovation show in uniformed and simplified process, reduced execution times, organizational changes. The project Business renovation plan of the Ministry of education, science and sport is successfully concluded, but it is only a part of whole business renovation project. The Business informatics institute has in this stage made suggestions of new business processes, their informatization and prepared the suggestions for organizational changes. In the next phase, the Ministry of education, science and sport has to implement suggestions in praxis which will, according to experiences, take a great deal of work, time and funds to do it.

## REFERENCES

- Bellinger, G. 2002. "Simulation Is Not The Answer". OutSights, [URL: <http://outsights.com/systems/simulation/simnotta.htm>].
- Burke G., Peppard J. 1995. Examining Business Process Re-engineering: Current Perspectives and Research Directions; Kogan Page: London.
- Davenport, T. H. 1993. *Process Innovation: Reengineering Work through Information Technology*. Harvard Business School Press, Boston.
- Dhaliwal, J. 1999. "An empirical review of the application of business process reengineering". In *Proceedings of the Fifth International Conference Integrating Technology & Human Decisions*, Athens, 1573-1575.
- Government Centre for Informatics. 2001. *Strategy of E-commerce in Public Administration for the Period 2001-2004 (SEP-2004), Version 1.0*. Government Centre for Informatics, Slovenia.
- Government Centre for Informatics. 2002. *Action Plan E-government up to 2004 (AP-2004), Version 1.1*. Government Centre for Informatics, Slovenia.
- Groznik, A. and A. Kovačič. 2002. "E-prenova poslovanja". *Zbornik posvetovanja Dnevi slovenske informatike 2002*, Portorož, 154-158.
- Hammer, M. and J. Champy. 1993. *Reengineering the corporation*. Harper Collins Books, New York.
- Kettinger, W. J. and V. Grover. 1995. "Toward a Theory of Business Process Change Management". *Journal of Management Information Systems* 12, No. 1.
- Prasad, B. 1999. "Hybrid re-engineering strategies for process improvement". *Business Process Management Journal* 5, No. 2, 178-197.
- Tapscot D. and A. Caston. 1993. *Paradigm Shift: The promise of Information Technology*. McGraw-Hil, New York.
- Tarumi, H., et al. 1999. "Evolution of business processes and a process simulation tool". In *Asia-Pacific Software Engineering Conference*. 180-187.
- Turban, E. et al. 2002. *Information technology for management: transforming business in the digital economy*. John Wiley & Sons, New York.

projects as a member of Business Informatics Institute project team since 2002.

## AUTHOR BIOGRAPHIES

**ALES GROZNIK** is assistant professor in the Department of Information Sciences at the Faculty of Economics, University of Ljubljana. He holds a first degree and a M.Sc. in Engineering and a M.Sc. and Ph.D. in Information Sciences from University of Ljubljana. He has extensive industry experience in management and strategic information systems gained working for several multinationals. His research interest is in the areas of information system role within the broader context of corporate objectives, management and strategic information system planning, information technology productivity, information technology management and the role of information systems in ever changing business environments.

**DEJAN VICIC** obtained his degree in 2004 on Faculty of Economics, University of Ljubljana, where he continues studying on Information Science M.Sc. programme. He is working on business renovation

# SIMULATION-BASED DECISION SUPPORT SYSTEM FOR AN ASSEMBLY LINE

Javier Otamendi  
Universidad Rey Juan Carlos, Campus Vicálvaro  
Facultad de Ciencias Jurídicas y Sociales  
Paseo Astilleros s/n  
28032 Madrid, Spain  
E-mail: [jotamendi\\_30@yahoo.com](mailto:jotamendi_30@yahoo.com)

## KEYWORDS

Decision Support Systems, Knowledge Representation, Simulation.

## ABSTRACT

An application of simulation in the ship building industry is presented. The model is part of the decision support system designed and built to help the programmers develop both weekly and long term assembly plans. The DSS is a mixture of spreadsheets and a simulation model where the worksheets act as the interface between the factory database and the model. A simulation optimization methodology that includes expert input to provide a robust initial solution is also incorporated into the system. The paper presents as well how the different knowledge groups (simulation specialists, production planners, in-house system developers, plant and factory managers) interact in the development and the maintenance of the application.

## INTRODUCTION

The competition in the ship building industry is furious, and the structure of the industry in Spain is old both in terms of technology, information management, and personnel.

Therefore, there is a need to update the systems to the new information era. Acquisition of new robots to increase throughput, development of new software to speed the decision making processes, training of personnel to improve the production process... They are all directions of improvement that are necessary to compete and that converge in the representation of knowledge in models to facilitate the decision making processes of the company.

In particular, there is the need in this Spanish factory to incorporate into a decision making tool the knowledge of the production planners to reduce the necessary time to develop the weekly production plans. Moreover, the introduction of new equipment has made the management believe that it was the proper time to incorporate the new plant layout into a simulation model along side the electronic knowledge obtained from the planners. The outcome has been a decision support system (DSS) that proposes production plans

that result in a reduction of the total time to assemble the ship. Shorter lead times mean cheaper ships and the possibility of competing in the international markets.

## THE SYSTEM

Let us start with a brief description of the operations performed in a specific production line to obtain the two parts in which a given unit or block is divided. The unit is called "sandwich" since it has a double layer of panels with support bones in between. The bottom layer (see Figure 1) is a series of panels welded together in which the bones are inserted and assembled. The top layer has nothing attached but is a similar welding combination of individual panels. It should be noted at this point that the final assembly of the ship is performed outside the specific plant of the factory being studied in this analysis.

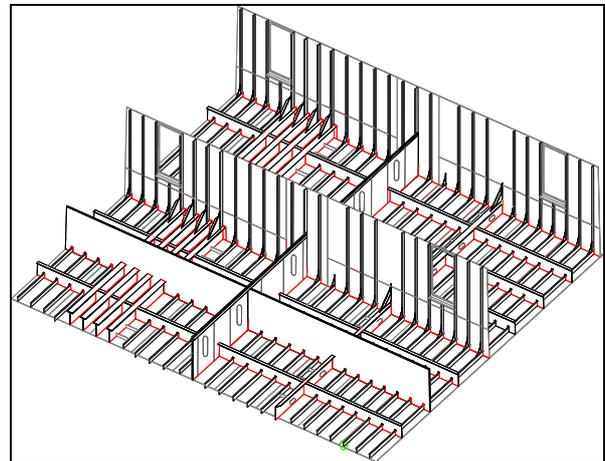


Figure 1. Bottom Part of Block

The plant has a long linear layout in which panels are first prepared before starting the assembly processes, which are divided into two groups (see Figure 2).

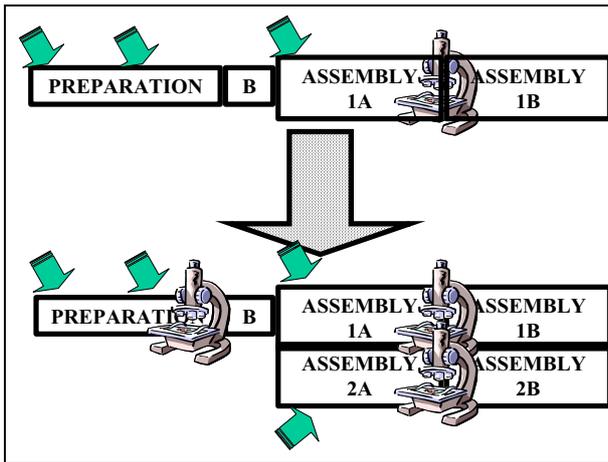


Figure 2. Changes in Layout

In the preparation phase, there is a huge transload that helps move the panels through different cutting and shaving operations, which are performed in the floor over rails. The operations might be manual or automatic, with two points of entry for raw materials. The panels are welded together to the right dimensions and then stored in an intermediate buffer. The exception is that the top layers of the sandwich are at this point removed from the line and stored until the bottom layer is assembled.

In the assembly phase, with four lines, the first part of the line, up until a big welding robot (part A), is used to assemble the bottom layer of the sandwich, inserting the bones (another entry point). Then, in the second part of the line (part B), is where minor welding is manually performed along with the necessary quality assurance checks.

In Figure 2, two different layouts are presented. The first layout includes the old layout whereas the second layout corresponds to the actual distribution. The change is due to the acquisition of a new robot that improves the throughput of the preparation line. Then, to improve the total number of sandwiches assembled, and having enough space and money, it has been possible to duplicate the second part of the process.

### THE DECISION SUPPORT SYSTEM

Due to these layout changes, it was the proper time to come up with a robust tool that helped the weekly planning of the line. The managers then launched the project of developing a simulation-based decision support system for the assembly line of sandwiches as a trial version for developing one for the whole factory.

The implementation of the DSS involved the combined work of four main groups: management, production planners, simulation specialists and system developers. The first and important step was to define the tasks to

be performed by each group and all the possible interactions (see Figure 3).

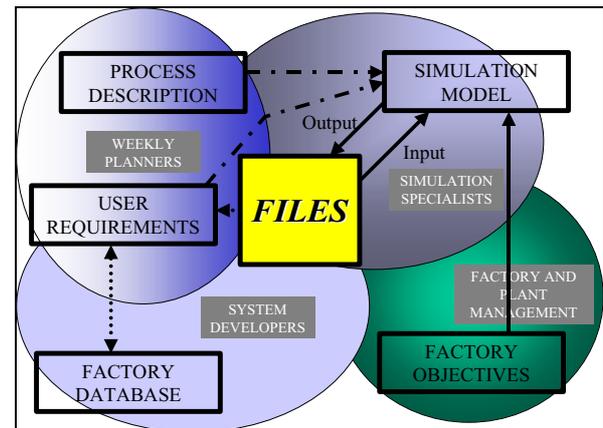


Figure 3. Implementation of the DSS

The production planners had to perform two complementary tasks. In order for the simulation specialists to create a simulation model, they had to put into words all of the knowledge they had not only in terms of the description of the necessary small tasks to assemble the product, but also in terms of the algorithm they use to develop the weekly plan. As it turned out, this last step was critical as it had more assembly restrictions than previously foreseen. Also, as final users of the software application, they had to define the charts and output lists they will be using in the line: Gantt charts, requirements of raw materials...

In terms of the software development, two different groups performed the work. On one side, the factory systems development group was given the task to generate in simple files all the necessary information to manage the line: process times, the production orders, the shifts... They had to combine the information coming from engineering with the one coming from labor unions, and also with the one coming directly from the line. They also had to create the output reports needed by the final users, taking as a primary input the information and the reports that come from the execution of the simulation model.

So, on the other side of the files line, the simulation specialists had to develop the model so it correctly represented the system. Also, the model had to take the necessary values from the files produced by the system developers to generate a set of files with all the necessary output values to produce the plans.

Management, besides overlooking the whole process, also had to set the objectives of the production plan, which are not just the ones of the plant in hand but of the whole ship building factory. The main criterion is obviously the lead time or time to assemble all the orders, but there are some competing criteria like assembling the top part of a sandwich immediately



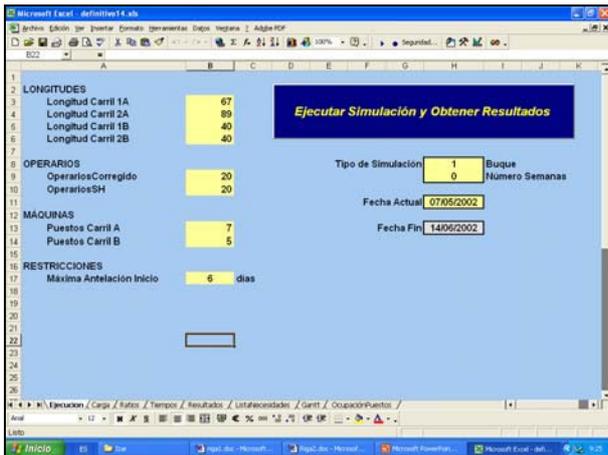


Figure 5. Front End

The information relates to the date in which the program is being executed and the length of the planning period, that could be set for a number of weeks or for the total number of orders to finish the whole ship.

It also includes a set of tactical variables that might be changed if the layout or the staff changes. The length of the lines might be modified as well as the number of its processing stations. The number of operators at each group of stations might also be altered.

#### *Shift*

It includes information about the time in which the operators are available. The information is filled automatically from an in-house application developed by the software developers of the company.

#### *Workload*

This spreadsheet contains the information about both the orders that are pending and those that are already in progress at the beginning of the simulated period. For each one, there is a description of its physical characteristics, like dimensions and weight. The data is also automatically updated by the information system of the company through yet another in-house developed application.

#### *Machines Turnover Ratio*

For each of the stations, this spreadsheet contains a table of velocities that depends on the number of operators that are working on a given unit and the characteristics of the unit. The tables are manually updated since they are constant over time. In case a change must be made the factory management should do it.

#### *Machine Execution Times*

Each of the operations performed at one particular machine have been subdivided into several tasks. This spreadsheet contains the setup and execution times for each task. The values are not likely to change so the

factory management manually includes them. Some of them are not constant but depend on the characteristics of each unit, so they are not readily calculated but are referred to the Machines Turnover Ratio spreadsheet.

#### *WITNESS input*

The simulation specialists developed an additional spreadsheet to reduce the amount of data interchanged with the simulator. The data from the Workload, Execution Times and Machine Turnover Ratio spreadsheets is combined into a single spreadsheet that contains only one line per production order with all its necessary attributes.

This worksheet is not made available to the user because it does not include input values but only formulas that are necessary to prepare the information for the simulator.

#### *Criteria specification*

The management of the factory specified the following important criteria:

- Total Assembly Time: the difference between the end of the last unit and the start of the first one should be minimized.
- Buffer Blockage: the percentage of time that the buffer is blocked should be minimized so the preparation line is not blocked.
- Assembly Line Occupation: the percentage of the time that the assembly stations are occupied is to be maximized.

Besides these three main criteria, another two were included as restrictions. The proposed start and finish times of any unit should not provided a large difference with the predetermined times provided by the factory management, since a large deviation could negatively influence in the rest of the plants of the factory.

#### **Simulation Model**

Built in WITNESS, it has been developed not with the idea of a user-friendly model, but as a mathematical model that needs to be optimized. That is, the model was developed to read data from MSEXCEL files, to automatically change the input parameters and to output the results corresponding to the selected plan back into the spreadsheets. It is then just a black box that can be thought of as a combination of a calculator and an experimenter.

However, it must be mentioned that the verification process has been very thorough. Even hand simulations for 30 units were used to trust the results of the black-box model.

## **The Optimization Routine**

The simulation model includes a search procedure of the appropriate production plan, that is, one that satisfies the given criteria.

There are several Simulation Optimization Methodologies (SOMs) that have been applied to the development of production plans. Among the reported combinations, genetic algorithms seems to be one good possibility (for example, Iyer and Saxena 2004, Yu and Liang 2001)

However, since the area of simulation optimization is too complex to develop universal search methodologies, a different search procedure has to be applied to any new situation. As it has already been mentioned, by including expert knowledge and production characteristics and restrictions, the search area is different for any given situation, calling for a particularized search procedure, especially in terms of the objective function (Hilgers and Boersma 2001).

Therefore, to develop the search procedure, an important amount of the time was devoted to talk with the production planners to understand their doing so it could be correctly represented in the SOM. The two main thrusts at this point were to develop good initial solutions and to provide “psychological validity” (Wager and Nichols 2003). In fact, after this step, in many situations, the first plan already fulfilled the requirements. The need is just for minor adjustments of the initial plan.

At the same time, by using production control theory, several concepts are included in the search algorithm. The more important idea is that of slackness, or difference between the available time until the due date and the total production time. Those units with small slack should be scheduled reasonably soon so the due date is met.

The total production time is also a key factor. If a unit with a large production time is scheduled first, the more blockages will be produced, but the higher the line occupation and the smaller the total assembly time.

The combination then of the plant production planners knowledge with the simulation specialists input in terms of simulation and production control theory provides a search algorithm that tests a satisficing initial solution and then modifies it in an additional small set of tries.

## **Output Files and Worksheets**

They are just raw data tables with a consensus format for the in-house software developers to produce the daily output.

## *Machine Utilization*

For each of the stations, its utilization rate is broken into the time that it has been working, the idle time, the time it has been blocked because the next station is full and the waiting time for labor to appear to perform the necessary operation.

It is worth mentioning at this point that two are the key statistics included in this worksheet. The first one is the blockage time for the buffer or the station, which is the link between the preparation line and the two assembly lines. The buffer acts as a distributor of workload between each of the assembly lines. This block time has been included as one of the main criteria of the study, trying to be minimized.

In fact, the management main policy is that all the stations are occupied. It is more important that the plant looks fully loaded with sandwiches even if many of the stations are blocked. The second key number in the table is therefore the idle time of each station.

A good production plan is perceived then if the idle time of the stations is low and so is the blockage rate of the distribution buffer.

## *Gantt Chart*

One of the spreadsheet is devoted to a Gantt chart that includes the unit number that should be in each station for the whole simulated period, usually a week, in intervals of one hour. This is the main control tool that the production controllers will use in a daily basis once the plan is finally decided.

## *Bill of materials*

Another key worksheet is the one that shows the materials required at each of the entry points of the system. It is used both by the production controllers and by the other plants in the factory that supply parts to this individual plant.

The worksheet shows four lists: the first for the first station, one more for the start of Preparation B and two more for each of the assembly lines entry points.

## *Operation Times*

For verification purposes, a table that shows the start and finish time of each sandwich at each station is created. The difference between the start at one station and the end at the previous one is then calculated to be the blockage time, which will help detect important assignable causes of blockage.

It also shows the start and finish of the whole unit so it can be compared with the planning times, which are provided by the production department of the whole factory.

## CONCLUSION

An application of simulation in the ship building industry is presented. Moreover, a decision support system based on a simulation model has been developed to come up with both weekly and long-term assembly plans. The DSS has been implemented at one of the assembly plants of a huge ship building factory that will keep on developing decision support systems for the rest of the plants.

The developed DSS is a mixture of spreadsheets and a simulation model where the worksheets act as the interface between the factory database and the model. In fact, the success of the project has come not only by the art of the simulation specialists to represent the system in hand, but, more importantly, by the ability to bring the knowledge and expertise of all the people involved in the decision system into a common set of spreadsheet files. Once a consensus has been reached on the format of both the input and output files, each knowledge group is responsible for either inputting data into the common files or reading from them. Each group is to maintain also its own part of the system.

A simulation optimization methodology has been incorporated at the same time into the DSS. Searching for a satisficing solution has been achieved by first including the knowledge of the weekly programmers and the engineers into the development of a good initial solution, which is improved using sound production control theoretical techniques.

The applications software is then for planners to use, but it has been the result of a combined effort of management and developers, which have conveniently defined a set of files to interchange information, in order to represent the expert knowledge in a simulation-based DSS.

In that sense, it is worth mentioning as the final conclusion that maybe more simulation studies could be performed if the experimentation and execution of the models was done from spreadsheets rather than from the model itself. Companies and users of the model many times fear that the simulation software is too complex to use so that they prefer to interact with it from a more standard tool like the MSEXcel spreadsheet.

## REFERENCES

- Bedworth, D.D. and J.E. Bailey. 1987. *Integrated Production Control Systems*. John Wiley and Sons, New York, NY.
- Hilgers A. and B.J. Boersma. 2001. "Optimization of Turbulent Jet Mixing." *Fluid Dynamics Research* 29, No.6, 345-368.
- Iyer S.K. and B. Saxena. 2004. "Improved Genetic Algorithm for the Permutation Flowshop Scheduling Problem." *Computers and Operatins Research* 31, No.4, 539-606.

- Shelokar, P.S.; V.K. Jayaraman; and B.D. Khulkarni. 2004. "An ant colony approach for clustering." *Analytica Chimica Acta* 509, No.2, 187-195.
- Wager, T.D. and T.E. Nichols. 2003. "Optimization of Experimental Design in fMRI: a General Framework Using a Genetic Algorithm." *NeuroImage* 18, No.2, 293-309.
- Yu, H. and W. Liang. 2001. "Neural Network and Genetic Algorithm-Based Hybrid Approach to Expanded Job-Shop Scheduling." *Computers & Industrial Engineering* 39, No. 3-4, 337-356.

## AUTHOR BIOGRAPHY



**JAVIER OTAMENDI** received the B.S. and M.S. degrees in Industrial Engineering at Oklahoma State University, where he developed his interests in Simulation and Total Quality Management. Back in his home country of Spain, he received a B.S. in Business Administration and a Ph.D. in Industrial Engineering. He is currently a simulation and statistics consultant and university professor. His e-mail address is: jotamendi\_30@yahoo.com

# INTEGRATED MODELING OF STRUCTURE-DYNAMICS CONTROL IN COMPLEX TECHNICAL SYSTEMS

Boris V. Sokolov

St.-Petersburg Institute for Informatics and Automation  
39, 14th Line, St. Petersburg, 199178, Russia  
E-mail: sokol@iiias.spb.su

Dmitry N. Verzhilin

Department of Informatics  
St.-Petersburg States University of Economics and Finances  
21, Sadovaya, St. Petersburg, 191023, Russia  
E-mail: verzhilin@SV10100.spb.edu

Eugene M. Zaychik

Public Corporation Vympel- Communication  
R, 3, Konnogvardeiski Bulvar, St. Petersburg, 190000, Russia  
E-mail: EZaychik@beeline.ru

## KEYWORDS

Operations and production management, scheduling, operations research, optimization, stochastic models.

## ABSTRACT

A concept of an active mobile object as a typical element of complex technical systems (CTS) with dynamic structures is introduced. The processes of CTS structure-dynamics control are described via a system of analytical and simulation models. General principles of joint use of these models within a simulation system were worked out. The problems of parametric and structural adaptation of the models are discussed. Theoretical results were implemented in program prototypes of computer-aided structure-dynamics control.

## INTRODUCTION

The main subject of our investigation is complex systems. That is the systems that should be studied through polytypic models and combined methods. In some instances investigations of complex systems require multiple methodological approaches, many theories and disciplines, and carrying out interdisciplinary researches. Different aspects of complexity can be considered to distinguish between a complex system and a simple one, for example: structure complexity, operational complexity, complexity of behavior choice, complexity of development (Klir 1985).

Classic examples of complex systems are: control systems for various classes of moving objects such as surface and air transport, ships, space and launch vehicles, etc, geographically distributed heterogeneous networks, flexible computerized manufacturing (Arkhipov et al. 2004; Sokolov and Yusupov 2002; Sokolov 2003).

One of the main features of modern complex technical systems (CTS) is the changeability of their parameters and structures as caused by objective and subjective reasons at different stages of the CTS life cycle. In other

words we always come across the CTS structure dynamics in practice. Under these conditions to increase (stabilize) CTS potentialities and capacity for work a structure control is to be performed. Reconfiguration is a widely used variant of the CTS structure control. Reconfiguration is a process of the CTS structure alteration with a view to increase, to keep, or to restore the level of CTS operability, or with a view to compensate the loss of CTS efficiency as a result of the degradation of its functions (Ackoff 1978; Athaus and Falb 1966; Napolitano and Swaim 1989a; Napolitano and Swaim 1989b; Sokolov and Yusupov 2004).

The presented considerations led us from a narrow traditional interpretation of CTS reconfiguration to a wide interpretation within a new applied theory of CTS structure-dynamics control. Developing of this theory is one the main aims of our investigations.

## APPROACH

As applied to CTS we distinguish the following main types of structures: the structure of CTS goals, functions and tasks; the organizational structure; the technical structure; the topological structure; the structure of special software and mathematical tools; the technology structure (the structure of CTS control technology). Structure dynamics control provides transition of CTS from the current macro-state to a given one. Figure 1 illustrates possible variants of structure dynamics in a complex technical system. The problem of CTS structure-dynamics control consists of the following groups of tasks: the tasks of structure dynamics analysis of CTS; the tasks of evaluation (observation) of structural states and CTS structural dynamics; the problems of optimal program synthesis for structure dynamics control in different situations. From our point of view, the theory of structure-dynamics control will be interdisciplinary and will accumulate the results of classical control theory, operations research, artificial intelligence, systems theory, and systems analysis. The two last scientific branches will provide a structured definition of the structure-dynamics control problem instead of a weakly structured definition.

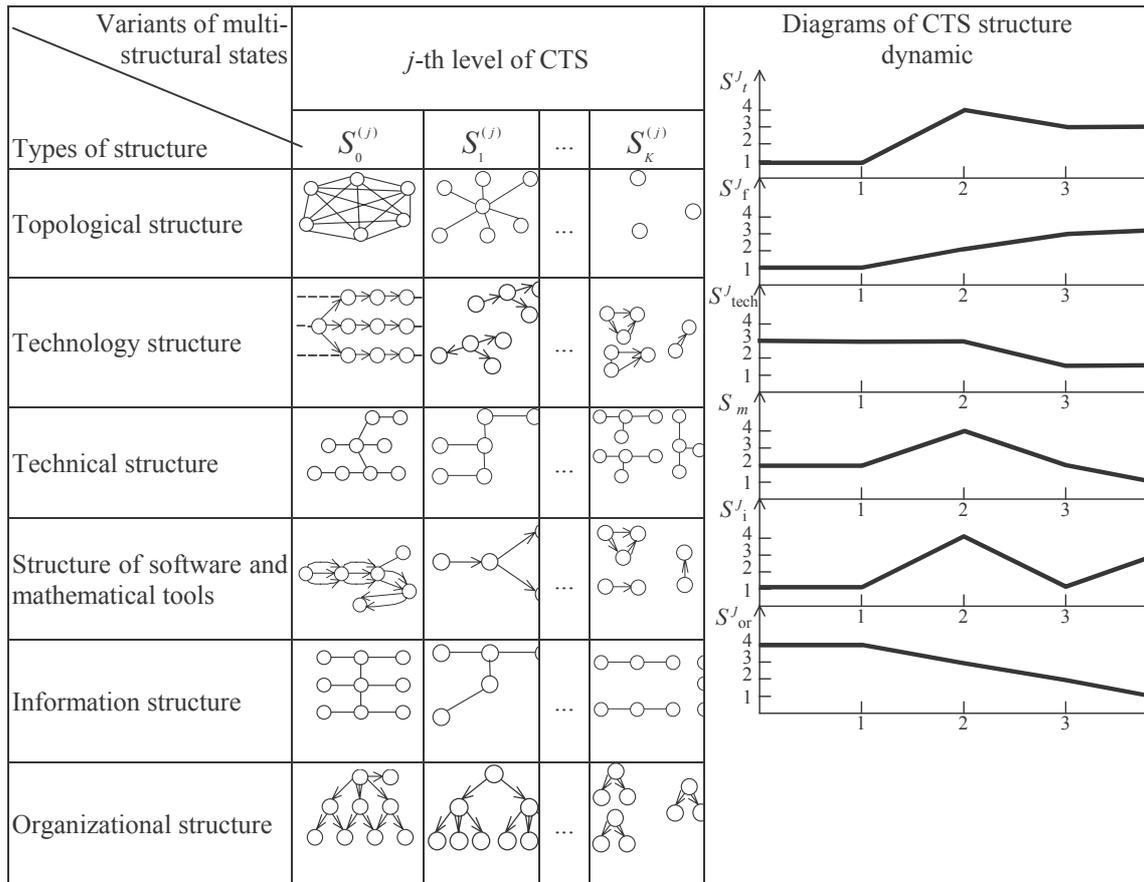


Figure 1: Possible variants of structure dynamics in complex technical systems

Today different methods and models are used for solving the problems of CTS structure-dynamics control. The known approaches to these problems are based on the PERT description of scheduling and control problems and traditional dynamic interpretation. The realization of these dynamic approaches produces algorithmic and computational difficulties caused by high dimensionality, non-linearity, non-stationary, and uncertainty of the models (Klir 1985; Athaus and Falb 1966; Siliak 1990).

We proposed to modify dynamic interpretation of operations control processes. The main idea of model simplification is to implement non-linear technological constraints in sets of allowable control inputs rather than in the right parts of differential equations. In this case, Lagrange coefficients, keeping the information about technical and technological constraints, are defined via the local-sections method. Furthermore, we proposed to use interval constraints instead of relay ones. Nevertheless the control inputs take on Boolean values as a result of the linearity of differential equations and convexity of the set of alternatives (Sokolov 2003). The proposed substitution lets use fundamental scientific results of the modern control theory in various CTS control problems (including scheduling theory problems).

As provided by the concept of CTS multiple-model description the proposed general model includes particular dynamic models: dynamic model of CTS motion control; dynamic model of CTS channel control; dynamic model of CTS operations control; dynamic model of CTS flows control; dynamic model of CTS resource control; dynamic model of CTS operation parameters control; dynamic model of CTS structure dynamics control; dynamic model of CTS auxiliary operation control.

Procedures of structure-dynamics problem solving depend on the variants of transition and output functions (operators) implementation. Various approaches, methods, algorithms and procedures of coordinated choice through complexes of heterogeneous models are developed by now.

CTS structure-dynamic control problem has some specific features in comparison with classic optimal control problems (Ackoff 1978; Athaus and Falb 1966; Okhtilev 2004; Siliak 1990). The first feature is that the right parts of the differential equations undergo discontinuity at the beginning of interaction zones (communication intervals). The considered problems can be regarded as control problems with intermediate conditions. The second feature is the multi-criteria nature of the problems. The third feature is concerned

with the influence of uncertainty factors. The fourth feature is the form of time-spatial, technical, and technological non-linear conditions that are mainly considered in control constraints and boundary conditions. On the whole the constructed model is a non-linear non-stationary finite-dimensional differential system with a re-configurable structure. Different variants of model aggregation were proposed. These variants produce a task of model quality selection that is the task of model complexity reduction. Decision-makers can select an appropriate level of model thoroughness in the interactive mode. The level of thoroughness depends on the input data, external conditions, and required level of solution validity. The proposed interpretation of CTS structure dynamics control processes provides advantages of modern optimal control theory for CTS analysis and synthesis.

## RESULTS

The preliminary investigations confirm that the most convenient concept for the formalization of CTS control processes is the concept of an active mobile object (AMO). In general case, it is an artificial object (a complex of devices) moving in space and interacting (by means of information, energy, or material flows) with other AMO, control system (AMO CS) and objects-in service (OS). The AMO consists of four subsystems relating to four processes (functioning forms): moving, interaction with OS and other AMO, functioning of the main (goal-oriented) and auxiliary facilities, and resources consumption (replenishment). The four functions of AMO are quite different, though the joint execution of these functions, the interaction being the main one, provide for AMO new characteristics. Thus, it becomes a specific object of investigation, and AMO control problems are strictly different than classical problems of mechanical-motion control. In general, AMO functioning includes informational, material, and energy interaction with OS, with other AMO, and with the environment. Along with the interaction, the facilities functioning, resource consumption (replenishment), and AMO motion are to be considered via functioning models (Sokolov 2003). The notion "Active Mobile Object" generalizes features of mobile elements dealing with different CTS types. Depending on the type of CTS the active mobile objects can move and interact in space, in air, on the ground, in water, or on water surface. Active Mobile Object can be also regarded as multi-agent system. The main classes of CTS structure dynamics problems include: AMO structure-dynamics analysis problems; AMO structure-dynamics diagnosis, observation, multi-layer control problems; problems of AMO generalized structural states synthesis and the problems of programs construction for a transition from a given CTS structural state to an allowable (optimal) structural state. Methodological and methodical basics for the theory of structure-dynamics control were developed. Methodological basics include: the methodologies of

the generalized system analysis and the modern optimal control theory for CTS with re-configurable structures. The methodologies find their concrete reflection in the corresponding principles. The main principles are: the principle of goal programmed control; the principle of external complement; the principle of necessary variety; the principles of multiple-model and multi-criteria approaches; the principle of new problems. The dynamic interpretation of structure-dynamics control processes lets apply the results, previously received in the theory of dynamic systems stability and sensitivity, for CTS analysis problems.

The multiple-model description of CTS structure-dynamics control processes is the base of comprehensive simulation technologies and of simulation systems. We assume the simulation system (SIS) to be a specially organized complex. This complex consists of the following elements: simulation models (the hierarchy of models); analytical models (the hierarchy of models) for a simplified (aggregated) description of objects being studied; informational subsystem that is a system of data bases (knowledge bases); and control-and-coordination system for interrelation and joint use of previous elements and interaction with the user.

The components of the simulation system were the main parts of the developed program prototypes during our investigation. The processes of AMO structure-dynamics control are hierarchical, multi-stage and multi-task ones. The structure of simulation system conforms the features of control processes. There are three groups of models in SIS: models of AMO CS and OS functioning (the 1st subsystem of SIS); models of evaluation (observation) and analysis of structural states and AMO CS structure-dynamics (the 2nd subsystem of SIS); and decision-making models for control processes in AMO CS (the 3rd subsystem of SIS).

The simulation system also includes: system of control, coordination and interpretation containing user interface and general control subsystem, local systems of control and coordination, subsystem of data processing, analysis, and interpretation for planning, control and modeling, subsystem of modeling scenarios formalization, subsystem of software parametric and structural adaptation, subsystem of recommendations producing for decision-making and modeling.

The data-ware includes databases for AMO states, for AMO CS states and general situation, for SO states and data bases for analytical and simulation models of decision-making and AMO CS functioning.

Existence of various alternative descriptions for CTS elements and control subsystems gives an opportunity of adaptive models selection (synthesis) for program control under changing environment.

Therefore we considered two general actual problems of the CTS structure-dynamics investigation, namely the problem of selection of optimal CTS structure-dynamics control programs at different states of the environment and the problem of parametric and structural adaptation

of models describing CTS structure-dynamics control. The formal statement and decomposition of structural and parametric adaptation tasks were worked out for models of CTS structure-dynamics control (SDC). Here the adaptive control should include the following main phases: parametric and structural adaptation of structure-dynamics control (SDC) models and algorithms to previous and current states of objects-in-service, of control subsystems, and of the environment; comprehensive scheduling of CTS operation (construction of SDC programs); simulation of CTS operation, according to the schedules, for different variants of control decisions in real situations; and structural and parametric adaptation of the schedule, control inputs, models, algorithms, and SDC programs to possible (predicted via simulation) states of SO, CS, and of the environment.

To implement the proposed concept of adaptive control let us consider two groups of parameters for CTS SDC models and algorithms: parameters that can be evaluated on the basis of real data available in CTS and parameters that can be evaluated via simulation models for different scenarios of future events.

The adaptation procedures can be organized in two blocks (models), namely SIS external adapter and SIS internal adapter.

As regards the CTS SDC models constructed previously, the following parameters belong to the first group and can be evaluated through the external adapter: the values of end conditions of the SDC models; technical and technological characteristics of CTS elements and subsystems [for example, computer speed, amount of random-access memory, maximal (minimal) intensity of resource consumption (regeneration), carrying capacity of data link channels]; and probabilistic characteristics and values of real and observed random processes.

The second group of parameters being evaluated through the internal adapter includes such characteristics as: redundancy rate for reserving of different type (functional, time, hardware/software, and informational reserving); priority of CTS SDC quality measures; and parameters defining the variants of compensation for trajectory deviations (violations of the schedule) in the simulation models.

When the parametric adaptation of SCS does not provide simulation adequacy then the structural transformations can be needed. Two main approaches to structural model adaptation are usually distinguished.

The first approach lies in the selection of a model from a given set. The model must be the most adequate to SO and CS. The second approach stands for CTS SDC model construction of elementary models (modules) in compliance with given requirements.

The second approach provides more flexible adjustment of SO and CS for particular functioning conditions. However, the first one is faster and can be effective if the application knowledge base is sufficiently large.

Both approaches need active participation of system

analysts and decision-makers who interact with SIS and consider hard-formalizing factors and dependences within the general procedure of CTS SDC program selection.

The structural adaptation of SCS takes a certain period of time, when the following main activities should be done: selection or construction (synthesis) of CTS SDC models meeting given requirements; selection or construction (synthesis) of CTS SDC algorithms for given conditions and given control problems; synthesis of software and data-ware for given control problems; and adjustment of SCS parameters for current and predicted states of SO and CS (parametric adaptation). Sometimes it is useful to adjust models and algorithms that are not currently used in CTS control processes, it will provide fast utilization of additional models when they are needed. The considered adaptation should be based on the results of CTS SDC simulation.

During our investigations the main phases and steps of a program-construction procedure for optimal structure-dynamics control in CTS were worked out.

At the first phase generation of allowable multi-structural macro-states is being performed. In other words a structure-functional synthesis of a new CTS pattern should be fulfilled in accordance with an actual or forecasted situation. Here the first-phase problems come to CTS structure-functional synthesis.

The general algorithm of the CTS structure-functional synthesis includes the following main steps.

Step 1. Gathering, analysis, and interrelation of input data for the synthesis of CTS multi-structural macro-states. Construction or correction of the appropriate models.

Step 2. Planning of a solving process for the problem of the CTS macro-states synthesis. Estimation of time and other resources needed for the problem.

Step 3. Construction and approximation of an attainability set for dynamic system. This set contains indirect description of different variants of CTS pattern (variants of CTS multi-structural macro-states).

Step 4. Orthogonal projection of a set defining macro-state requirements to the attainability set.

Step 5. Interpretation of output results and their transformation to a convenient form for future use (for example, the output data can be used for construction of adaptive plans of CTS development).

At the second phase a single multi-structural macro-state is being selected, and adaptive plans (programs) of CTS transition to the selected macro-state are constructed. These plans should specify transition programs, as well as programs of stable CTS operation in intermediate multi-structural macro-states. The second phase of program construction is aimed at a solution of multi-level multi-stage optimization problems.

The general algorithm of problem solving should include the following steps.

Step 1. Input data for the problem are being prepared and analyzed in an interactive mode. During this step a

structural and parametric adaptation of models, algorithms, and special software tools of simulation system is being fulfilled to the past and to the current states of the environment, of object-in-service, of control subsystems embodied in existing and developing CTS. For missed data simulation experiments with SIS models or expert inquest can be used.

Step 2. Planning of comprehensive modeling of adaptive CTS control and development for the current and forecasted situation; planning of simulation experiments in SIS; selection of models, selection of model structure; determination of methods and algorithms for particular modeling problems, selection of models and model structure for these problems; estimation of necessary time.

Step 3. Generating via comprehensive modeling, of feasible variants of CTS functioning in initial, intermediate, and required multi-structural macro-states; introducing of the results to a decision-maker; preliminary interactive structure-functional analysis of modeling results; producing of equivalent classes of CTS multi-structural macro-states.

Step 4. Automatic putting into operation of data of CTS functioning variants; analysis of constraints correctness; final selection of aggregation level for CTS SDC models, and for computation experiments aimed at CTS SDC program construction.

Step 5. Search for optimal CTS SDC programs for transition from a given multi-structural macro-state to a synthesized one and for stable CTS operation in intermediate multi-structural macro-states.

Step 6. Simulation of program execution under perturbation impacts for different variants of compensation control inputs received via methods and algorithms of the real-time control.

Step 7. Structural and parametric adaptation of the plan and SIS software to possible (forecasted through simulation models) states of SO, CS, and of the environment.

Here CTS structural redundancy should be provided for compensation of extra perturbation impacts. After reiterative computation experiments the stability of constructed CTS SDC plan is being estimated.

Step 8. Introducing of comprehensive adaptive planning results to a decision-maker; interpretation and correction of these results.

One of the main opportunities of the proposed method of CTS SDC program construction is that besides the vector of program control we receive a preferable multi-structural macro-state of CTS at final time. This is the state of CTS reliable operation in the current (forecasted) situation.

The combined methods and algorithms of optimal program construction for structure-dynamics control in centralized and non-centralized modes of CTS operation were developed too.

The main combined method was based on joint use of the successive approximations method and the “branch

and bounds” method. A theorem characterizing properties of the relaxed problem of CTS SDC optimal program construction was proved for a theoretical approval of the proposed method.

Classification and analysis of perturbation factors having an influence upon operation of a complex technical system were performed. Variants of perturbation-factors descriptions were considered for CTS SDC models. In our opinion, a comprehensive simulation of uncertainty factors with all adequate models and forms of description should be used during investigation of CTS SDC. Moreover, the abilities of CTS management should be estimated both in normal mode of operation and in emergency situations. It is important to estimate destruction “abilities” of perturbation impacts. In this case the investigation of CTS functioning should include the following phases:

- Determining of scenarios for CTS environment, particularly determining of extremely situations and impacts that can have catastrophic results.

- Analysis of CTS operation in a normal mode with the help of a priori probability information (if any), simulation, and processing of expert information through the theory of subjective probabilities and theory of fuzzy sets.

- Repetition of the previous item for the main extremely situations and estimation of guaranteed results of CTS operation in these situations.

- Computing of general (integral) efficiency measures of CTS structure-dynamics control.

Algorithms of parametric and structural adaptation for CTS SDC models were proposed. The algorithms were based on the methods of fuzzy clusterization, on the methods of hierarchy analysis, and on the methods of a joint use of analytical and simulation models.

The SDC application software for structure-dynamics control in complex technical systems was developed. It included elements for the above-mentioned 1st subsystem of SIS. External simulation software was inevitable at the current stage of our investigations. Specificity of AMO CS can be efficiently described via the terms of the queuing theory. Thus the use of the General Purpose Simulation System (GPSS world by Minuteman Software) was rather reasonable as it includes all necessary elements for such description. The embedded programming language under simulation (PLUS) provided a more flexible adjustment of simulation models to the input data. However, the processes of AMO operation and control differ from the typical servicing processes that are usually simulated via GPSS. The main distinction is that the control operations are executed according to previously obtained plans (SDC programs). Thus, the process of plan implementation under the presence of perturbation impacts should be simulated.

Characteristics of AMO control processes were arranged in a sequence of matrices containing the information about the control subsystems, the interaction zones (intervals of communication with

AMO), and the control operations. Systems of servicing facilities were automatically built via the PLUS language according to the contents of the matrices. The use of the GPSS models helped to evaluate reliability of the SDC programs.

## CONCLUSIONS

Methodological and methodical basics of the theory of CTS structure-dynamics control are developed by now. This theory can be widely used in practice. It has interdisciplinary basis provided by classic control theory, operations research, artificial intelligence, systems theory and systems analysis. The dynamic interpretation of CTS reconfiguration process provides strict mathematical base for complex technical-organizational problems of high importance that were never formalized before and have high practical importance.

The proposed approach to the problem of CTS structure reconfiguration control in the terms of general context of CTS structural dynamics control enables the following: common goals of CTS functioning to be directly linked with those implemented (realized) in CTS control process; a reasonable decision and selection (choice) of adequate consequence of problems solved and operations fulfilled related to structural dynamics to be made; and a compromise distribution (trade-off) of a restricted resources appropriated for a structural dynamics control to be found voluntary.

This work was supported by European Office of Aerospace Research and Development (Project # 1992p), Russian Foundation for Basic Research (grant #02-07-90463, 05-07-90088), Institute for System Analysis RAS (Project 2.5).

## REFERENCES

- Ackoff, R.L. 1978. *The Art of Problem Solving*. Wiley-Interscience, New York.
- Arkhipov A.V., Ivanov D.A., Sokolov B.V. 2004. "Intelligent Supply Chain Planning in "Virtual Organization" In *Proceedings of the 5th IFIP Working Conference on Virtual Enterprises PRO-VE'04* (France, Toulouse), Vol.8, Part 8, 215-224.
- Athaus, M. and Falb P.L. 1966. *Optimal control: An Introduction to the Theory and Its Applications*. McGraw-Hill Book Company. New York, San Francisco, Sidney.
- Klir, G.J. 1985. *Architecture of Systems Problem Solving*. Plenum Press, New York.
- Napolitano M.R., Swaim R.L. 1989 "A New Technique for Aircraft Flight Control Reconfiguration" In *Proceedings of the AIAA Guidance, Navigation and Control Conference*. Pt 1, 1-9.
- Napolitano M.R., Swaim R.L. 1989 "An Aircraft Flight Control Reconfiguration Algorithm" In *Proceedings of the AIAA Guidance, Navigation and Control Conference*. Pt 1, 323-332.
- Siliak, D.D. 1990 *Decentralized Control of Complex Systems*. Academic Press, New York.
- Okhtilev M.Yu. 2004 "The Data Flow and Distributed Calculations Intelligence Information Technology for Decision Support System in Real Time" In *Proceedings of*

*the 6th International Conference on Enterprise Information Systems, ICEIS 2004* (Porto, Portugal). Vol.2, 497-500.

- Sokolov B.V., Yusupov R.M. 2002. "Complex Simulation of Automated Control System of Navigation Spacecraft Operation" *Journal of Automation and Information Sciences* 34, No 10 (Oct), 19-30.
- Sokolov B.V. 2003 "Optimal Structure Reconfiguration in a Complex-Technical Systems (CTS): principles, models, methods and algorithms for the CTS structure dynamics control" In *Proceedings of the 2003 VI ISTC Scientific Advisory Committee Seminar "Science and Computing"* (Moscow, Russia). Vol. 1, 142-149.
- Sokolov B.V. Yusupov R.M. 2004 "Conceptual Foundations of Quality Estimation and Analysis for Models and Multiple-Model Systems" *Journal of Computer and System Sciences International* No 6 (accepted for publication in USA).

## AUTHOR BIOGRAPHIES

**BORIS V. SOKOLOV** was born in Leningrad (now Saint-Petersburg), Russia in 1951. He obtained his main degrees in Mozhaisky Space Engineering Academy, Leningrad. MS in Automation Control Systems of Space Vehicles in 1974. Candidate of Technical Sciences subject the area of planning automation and decision making in 1982. Doctor of Technical Sciences subject the area of military cybernetics, mathematical modeling and methods in military research. Professional Interests: Basic and applied research in mathematical modeling and mathematical methods in scientific research, optimal control theory, mathematical models and methods of support and decision making in complex organization-technical systems under uncertainties and multi- criteria. At present he is a leading researcher of St.-Petersburg Institute for Informatics and Automation. His e-mail address is: sokol@iiias.spb.su and his Web-page can be found at <http://www.spiiras-grom.ru>.

**DMITRY N. VERZILIN** was born in Leningrad (now Saint-Petersburg), Russia in 1960. He graduated from Mathematical faculty of Leningrad States University in 1982. He obtained the degree of Candidate of Technical Sciences in Mozhaisky Space Engineering Academy, 1992 and the degree of Doctor of Economics in St.-Petersburg States University of Economics and Finances, 2004. At present he is a lecture at the department of Informatics in this university. Professional interests lay in operations research, simulation, and statistical analysis. His e-mail address is [verzilin@SV10100.spb.edu](mailto:verzilin@SV10100.spb.edu).

**EUGENE M. ZAYCHIK** was born in 1982 in Rostov-on-Don, Russia. He graduated from Military Academy of Communication in 1994. He is a Candidate of Technical Sciences (1994, Military Academy of Communication). He is a specialist in simulation and control of communication systems. At present he works in the Public Corporation Vympel-Communication. His e-mail address is [EZaychik@beeline.ru](mailto:EZaychik@beeline.ru).

# SUCCESSFUL AUTOMATION OF A LINE OF G.R.C. PANELS USING SIMULATION

Jose M. Pastor

Department of Systems Engineering and Automation  
Polytechnic School of University Carlos III of Madrid  
Avda. Universidad, 30. 28911 Leganes (Madrid) SPAIN  
E-mail: jpastor @ing.uc3m.es

Javier Otamendi

Department of Applied Economy I  
University Rey Juan Carlos, Campus Vicalvaro  
Facultad de Ciencias Jurídicas y Sociales  
Paseo Astilleros s/n 28032 (Madrid) SPAIN  
E-mail: jotamendi\_30@yahoo.com

Carlos Corpas

Department CARACOLA, GRC-Mesas unit  
Construcciones Especiales y Dragados S.A. (DRACE)  
C/ Brújula, 20. 28850 Torrejón de Ardoz (Madrid) SPAIN  
E-mail: ccr-drace-torreon@dragados.com

## KEYWORDS

Model simulation, Discrete Event Systems, Process Optimisation, Decision Support.

## ABSTRACT

This article presents the modeling and simulation study to optimize the manufacturing process of prefabricated G.R.C. (Glass Reinforced Cement) panels for building facades. The process optimisation has changed the tasks organisation, and a proposal to automate the works with more value added has been made to increase panels productivity and quality.

The main characteristic of this products is the flexibility to obtain panels with complex three-dimensional forms. The "DRACE System of GRC Prefabricated Panels for Building Facades" published in (DRACE 2001) has a detailed explanation of characteristics, specifications and manufacturing operations of GRC panels.

## INTRODUCTION

Construction industry automation is still well below the levels of other industries, although an increasing effort has been made in recent years. Applying automation in this important sector is difficult because of the non-repetitive processes and the low level of standardisation. Construction activities can be divided into two main groups: off-site and on-site. On site processes are more relevant and form what is considered typical construction work, i.e. building, civil works, etc. Off-site construction processes are more suitable to be automated, since the work takes place in a structured environment and process variables are under control.

A common off-site process is the manufacturing of prefabricated panels which are later assembled on-site. In recent years one important material used in this kind of industry is the GRC. Thanks to its flexibility, this technology has become very popular. GRC material is prepared mixing cement with small cut glass fiber strips, achieving enough flex-traction strength while maintaining light weight ( $60 \text{ kg/m}^2$  in comparison with

conventional concrete panels  $220 \text{ kg/m}^2$ ). This allows the manufacture of large panels ( $7 \times 3 \text{m}$ ) of any 3D geometry, see Figure 1, with the dual advantage of easy transportation and easy assembly on site.



Figure 1: Panels of GRC with Different Shapes

The Spanish construction company DRACE has been using manually manufactured GRC panels mainly as facade units, see an example in Figure 2, for a long time. The excellent finishing quality of the external parts of GRC panels enables to apply them in a great variety of circumstances. Therefore a project to develop an automated manufacturing factory of prefabricated GRC panels is being launched by DRACE with the financial support of the Spanish Ministry of Industry.



Figure 2: Aspect of Typical GRC Facade

This article presents the use of modelling and simulation (O'kane et al 2000), to find the best process lay-out, see (Potluri and Atkinson 2003) for automation and optimisation of GRC facade panels manufacturing tasks.

## OBJECTIVES

The main objective of the analysis is to develop a flexible manufacturing system that will be monitored with a simulation model that helps decision making both in the short term and in the long run, (Benjamin et al. 1999).

In particular, the main thrusts are:

- Improving panel quality.
- Reducing materials consumption.
- Increasing productivity.

Those ideas might be achieved by:

- Reducing manual labour: due to the high repercussion on the final costs.
- Improving the manufacturing process: tasks reorganisation and elimination, automation....

## THE PRODUCTS

Panels differ according to the type and number of layers to be sprayed. The first layer, which forms the external surface of the resulting panel, is common to all of them.

It consist of mortar without fibre up to a total thickness of 2 mm. Depending on the remaining layers, there are five distinct types of panels showed in Figure 3.

- Plain shell: two more layers of mortar and fiber up to a total thickness of 10 mm.
- Shell with ribs: same as plain shell but with stiffening ribs.
- Stud frame: same as plain shell but with a steel frame structure.
- Shell with insulation: same as plain shell but with insulation sheets.
- Sandwich: same as plain shell with insulation with and additional GRC top layer.

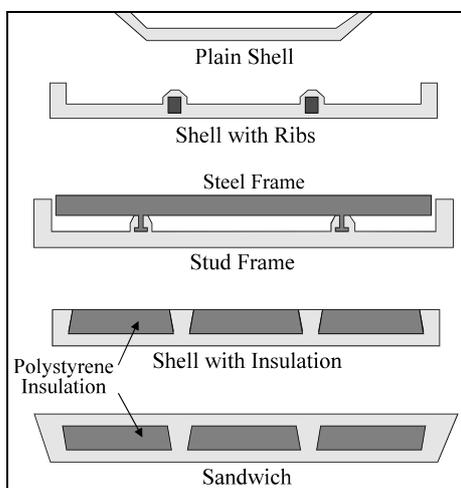


Figure 3: Types of GRC Panels

## THE AUDIT OF THE ACTUAL PROCESS

### The Actual Process

The complete process of GRC facade panels manufacturing is divided in three steps: panel geometry design, off-site manufacturing and on-site assembly.

The panels design is obtained from building facades division, as shown in Figure 4. The facade partition must follows several rules: constructive restrictions, aesthetic criteria, transport and assembly capabilities, etc. see (Pastor et al. 2001). The panel design is used to construct a wood model of panel. This model is used to make one or several moulds of the panel in the same GRC material. The number of panel moulds is a decisive factor to define manufacturing planing and panels delivery to on-site assembly in building facade. However, usually only one mould is made because his cost is very high.

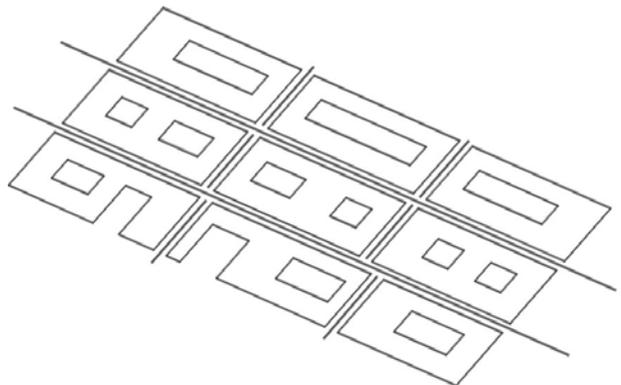


Figure 4. Facade Division in Panels.

To sum up, the manufacturing process is composed of the next tasks, see detailed information in (Peñin et al. 1998):

- a) Mould cleaning and preparing for spraying
- b) Successive sprayings and compactions of GRC layers depending on the type of panel.
- c) Auxiliary elements positioning
- d) Panel hardening during ten hours
- e) Panel extraction and gathering with cranes

The process is made by hand, as shown in Figure 5, where we can see manual spraying and compacting.



Figure 5. Manual spraying and compacting operations

The last step is the panel transport by lorry, and panel assembly on facade building. The panel is usually welded or screwed down to the building structure.

### The Process Audit

The first step of the study has been an audit, (Law 2000). The actual process is manual and must be understood before the automation study.

The production process is not linear but a one-station process. At each of the nine stations, the operations might be divided into, preparing operations, GRC spraying, layers compacting and finishing operations, see Figure 6.

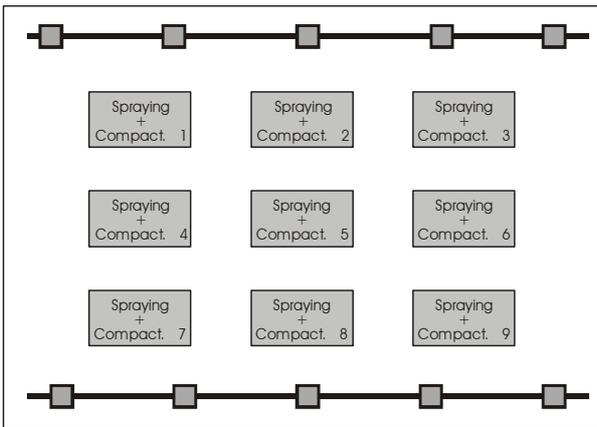


Figure 6: Current One-Station Process

To complete the audit, historical values are obtained to quantify the percentage per type of product and per size of the panel, as shown in Figure 7.

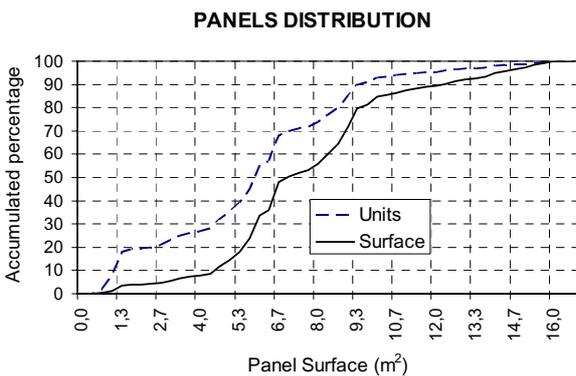


Figure 7: Statistic of Panels Surface Distribution

The first conclusion is that the manufacturing philosophy had to be changed from the fixed stations and the movement of operations between stations to an automated line in which a robot performs the spraying operations and the panels move between manual compacting stations with a set of conveyors. The main side effect is the specialisation of operators as well as the possibility of increasing the size of the panels thanks

to robot spraying. The dimensions of the panel to be manufactured in the automated system are 7 m. long x 3,5 m. wide, total surface of 24,5 m<sup>2</sup>.

The statistic study collects information of nine buildings with around 5.000 panels and a total surface of 48.000 m<sup>2</sup>. The manufacturing percentages of different kind of panels grouped in Shells and Sandwich are: 35% of Shells and 65% of Sandwich.

Due to the initial ignorance on how to optimize the new layout, a simulation model was developed in order to model the initial ideas and test new ones. The simulation tool used to make the model is the last version of SIMFACTORY of Caci company, see program interface in Figure 8.

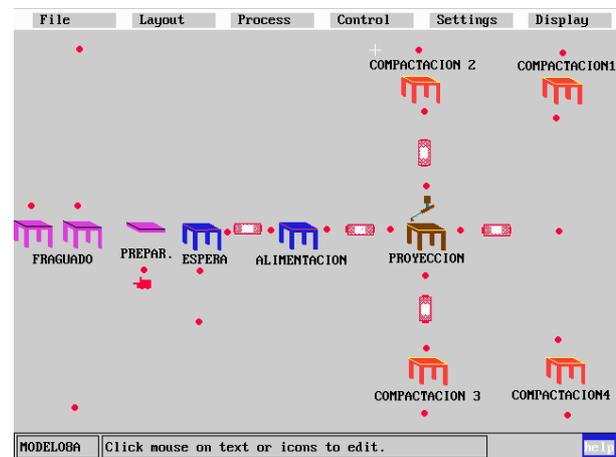


Figure 8: SIMFACTORY Interface

For that reason, a parameterized model was built in which the following variables could be modified:

- ♣ Number of spraying stations
- ♣ Number of compacting stations
- ♣ Number of compacting teams of operators
- ♣ Number of operators per compacting team
- ♣ Conveyor system and its control
- ♣ Feeding system
- ♣ Percentage of inefficiencies
- ♣ Assigning operators to zones or tasks
- ♣ Sequencing of panels

The main decision variables were determined to be:

- ♣ Throughput in an 8 hour shift per product
- ♣ Operators utilization
- ♣ Robot utilization

In order to do a more realistic and reliable simulation, processes times has been obtained directly from one of DRACE factories in Torrejón de Ardoz near Madrid.

The times of preparation operations before spraying and compacting are:

- Cleaning and oiling: 1,7 min/m<sup>2</sup>
- Mould preparing: 3,2 min/m<sup>2</sup>

The operation times after spraying and compacting are:

- Demoulding: 1,3 min/m<sup>2</sup>.
- Panel extraction: 1,2 min/m<sup>2</sup>.
- Panel gathering: 1,9 min/m<sup>2</sup>.

In short, the robot spraying times and compacting times are summarised in Table 1 and Table 2, respectively. Panel are classified in three groups depending of his surface: Small (1-8 m<sup>2</sup>), Medium (8-12 m<sup>2</sup>) and Large (12-24 m<sup>2</sup>). The units of robot spraying times are min/m<sup>2</sup>, and the units of compacting times are min/(team·m<sup>2</sup>) with three operator per team.

Table 1. Robot Spraying Times

Phase	Panel Type	Small Panel	Medium Panel	Large Panel
1	Shell	1,48	2,96	4,44
2	Shell	1,04	2,08	3,12
1	Sandwich	0,92	1,84	2,76
2	Sandwich	1,20	2,40	3,60
3	Sandwich	1,28	2,56	3,84
4	Sandwich	1,12	2,24	3,36

Robot spraying times are obtained from manual spraying taken into account that the velocity and flow can be greater than those in manual spraying. The supposition is that the robot velocity and flow are double that the ones in manual operation.

Table 2. Compacting Times

Phase	Panel Type	Small Panel	Medium Panel	Large Panel
1	Shell	3,52	7,04	10,56
2	Shell	4,10	8,25	12,43
1	Sandwich	3,52	7,04	10,56
2	Sandwich	2,48	4,96	7,44
3	Sandwich	2,48	4,96	7,44
4	Sandwich	4,10	8,25	12,43

The maximum throughput taking into account only the robot spraying capacity with 25% of inefficiencies is around 380 m<sup>2</sup>/shift, with the same distribution of kinds and panel surfaces obtained from statistical studies.

### THE SIMULATION STUDY

With the frameset correctly in place (input and output variables and the simulation model), it was the time to experiment with the model. The improvement routine in this case was based on an intelligent trial and error procedure in which all the people involved had something to say after each individual model was set, run and analyzed. It was an iterative process of learning and adding the acquired knowledge to the system.

What was crucial was that management, operators, automation specialists and software developers worked together in the definition of the input values and the

restriction. As it turned out, the biggest restrictions were in terms of money and the control of the transportation systems, which requires a complicated software to develop and run.

The first considerations help reduce model variables. The number of spraying stations will be one, mainly due to robotized spraying cell costs. The reasonable number of compacting stations according to the tables of process times will be between two and four and the number of teams will be no more than one per table, i.e. between two and four. The number of operators per team will be between two, the minimum to do the compacting task, and four, the maximum to work in a table.

The simulation models are developed from several proposed lay-outs. The lay-outs represent an organisation of the work stations and transportation system. The iterative lay-outs evolution is based on the successive simulation results. The first lay-out was proposed by DRACE company, Figure 9, with one spraying station and three compaction stations.

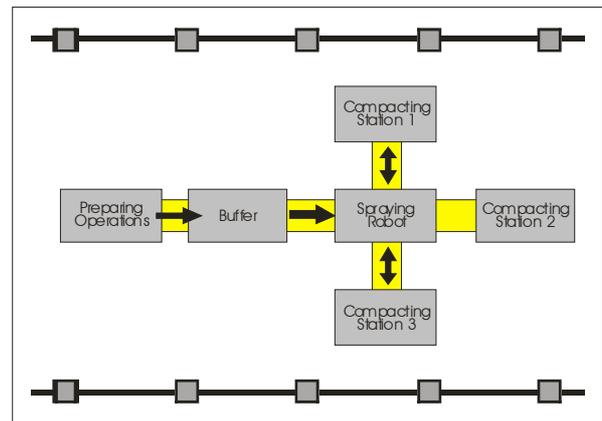


Figure 9: Lay-Out 1

The first simulation results, see Table 3, and the usual work organisation show that the best number of operators per compaction team is three. This is mainly due to the distribution of the different compaction subtasks.

Table 3. Simulation Results for Lay-out 1

Lay-out	N° teams - N° Oper	m <sup>2</sup> /8H	m <sup>2</sup> /8h Oper.	Robot %	Team %
1	2 - 2	185	46,2	35	85
1	3 - 2	211	35,1	39	67
1	4 - 2	229	28,6	42	48
1	2 - 3	268	44,7	51	82
1	3 - 3	288	32,0	55	61
1	4 - 3	315	26,5	60	45
1	2 - 4	316	39,5	60	73
1	3 - 4	320	26,7	61	51
1	4 - 4	337	21,1	63	36

The second proposed lay-out, Figure 10, adds one compaction station and the movement of the robot between two positions of panel moulds in spraying station, one awaiting while spraying in the other one.

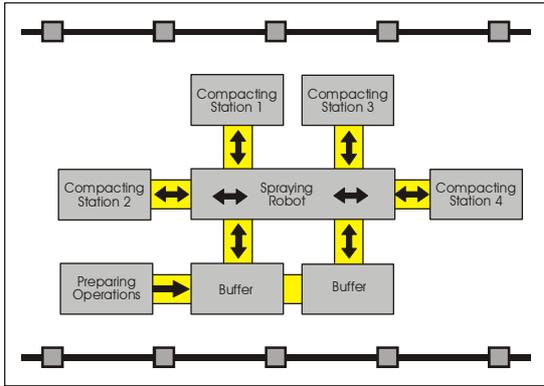


Figure 10: Lay-Out 2

The third proposed lay-out, Figure 11, follows the same philosophy, with slightly different station distribution and transportation system.

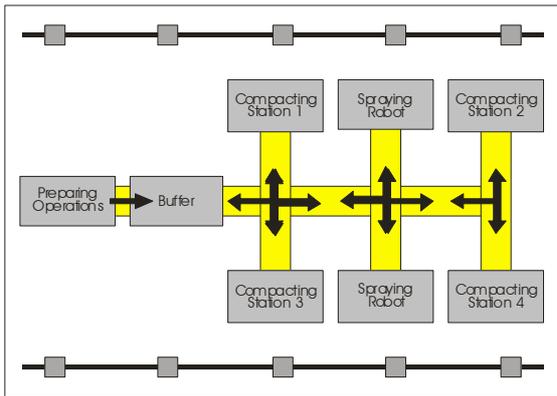


Figure 11: Lay-Out 3

The fourth lay-out, Figure 12, changes the transport organisation. It is a circulating system to do the successive GRC layers in each complete turn. The robot is now in a fixed position.

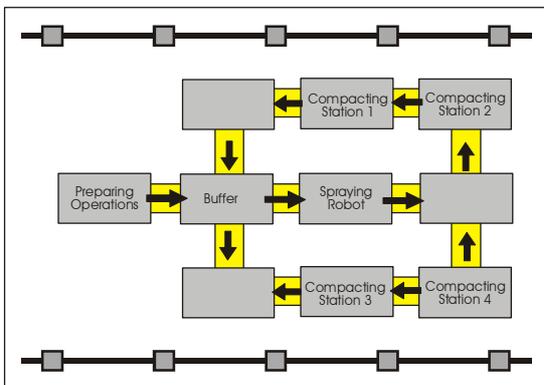


Figure 12: Lay-Out 4

The simulations results obtained to compare with the proposed lay-out are shown in Table 4.

Table 4. Simulations Results for Proposed Lay-outs

Lay-out	N° teams - N° Oper.	M <sup>2</sup> /8H	m <sup>2</sup> /8h Oper.	Robot %	Team %
1	2 - 3	268	44,7	51	82
1	3 - 3	288	32,0	55	61
1	4 - 3	315	26,5	60	45
2	2 - 3	296	49,3	59	90
2	3 - 3	380	42,2	75	81
2	4 - 3	438	36,5	87	62
3	2 - 3	292	48,7	58	89
3	3 - 3	371	41,2	73	80
3	4 - 3	429	35,7	86	61
4	2 - 3	302	50,3	60	92
4	3 - 3	386	42,9	77	81
4	4 - 3	440	36,7	88	62

These results show that the first proposed lay-out can be discarded because of his low throughput compared to the other ones. The results of lay-outs 3 and 4 are good, but similar to lay-out 2, which is a less expensive system.

Table 4 shows too that the use of four teams has a pour efficiency with small throughput increment. The best option is manufacturing usually with two compacting teams and with three teams in peak productions.

The last proposed lay-out, Figure 13, try to join the advantages of the lay-out 1, simplicity and lower cost, and the lay-out 2, good results with high efficiency.

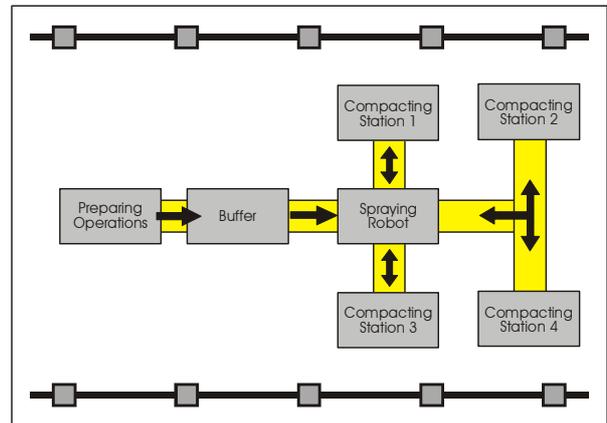


Figure 13: Lay-Out 5 Proposed after Simulation Study

The simulation results, collected in Table 5, permit the comparison of the lay-out 2 with the last proposed lay-out. Lay-out 5 has slightly lower results than lay-out 2, however its lower cost and simplicity justify the election of this final proposal.

Other simulation models were developed with the same lay-out to analyse other characteristics. One type of

modelling assigns each panel and two work stations to a compaction team. These restrictions bear a small reduction of efficiency, however they suppose a better work organisation to assign responsibilities to teams.

Table 5. Simulation Results

Lay-out	Nº teams - Nº Oper.	M <sup>2</sup> /8H	m <sup>2</sup> /8h Oper.	Robot %	Team %
2	2 de 3	268	44,7	51	82
2	3 de 3	288	32,0	55	61
4	2 de 3	255	42,5	50	80
4	3 de 3	271	30,0	53	59

Other family of models analyses the influence of panel sequencing with several criteria: random, by panel type, by panel surface. The best results were obtained sequencing panel by type, from more simple to more complex. Each type must be sequenced by surface, from lower surface to higher surface.

## CONCLUSION

The conclusion is what came out of the simulation analysis was the layout represented in Figure 5. The chosen lay-out is the one represented in Figure 5.

It gives a robust design both under normal operation and under the presence of inefficiencies. Even losses of 25% only result in a 5-8% decrease in throughput with an 80% utilisation of the robot. The theoretical increase of throughput over the manual manufacturing process is of 15%, which is enough for management to proceed with the installation.

It is very important that the sequence of panels is planned by type and surface to optimise the system production. It is advisable to assign to each compaction team two work stations for arriving panels.

## VALIDATION: THE BIRTH OF A NEW PROJECT

After year and a half of operation with the robot, it was time to assess if the results of the simulation model were correct. The utilization of the operators shows an increase of 10%, that, with the 80% utilization of the robot results in a total increase of a 12%. The difference was assigned to some of the new handling operations that take more than initially foreseen.

However the amortization of the robot was well on its way. So well, that the simulation project for the automation of the compacting operations is already under way.

## REFERENCES

Benjamin, P.C., Erraguntla, M. and Mayer, R.J. (1999). Using simulation for robust system design. *Simulation*. Vol. 65, 116-127.

DRACE (2001). "DRACE System of GRC Prefabricated Panels for Building Facades". *DIT: Technical Suitability Document in Non traditional materials and procedures of construction*, Ed: Instituto de Ciencias de la Construcción Eduardo Torroja. No. 367.

Law, A.M (2000). *Simulation modeling and analysis*. Third edition, McGraw-Hill series in industrial engineering and management science, New York.

O'Kane J.F., Spenceley J.R. and Taylor (2000). Simulation as an essential tool for advanced manufacturing technology problems. *Journal of Material Process Technology*. Vol.107, 412-424.

Pastor, J.M., Balaguer, C., Rodríguez, F.J. and Diez, R. (2001). Computer-Aided Architectural Design Oriented to Robotized Facade Panels Manufacturing. *Computer-Aided Civil and Infrastructure Engineering*. Vol. 16, No. 4, 216-227.

Peñin, L.F., Balaguer, C., Pastor, J.M., Rodríguez, F.J. Barrientos, A. and Aracil, R. (1998). Robotized Spraying of Prefabricated Panels. *IEEE Robotics and Automaton Magazine*, Vol. 5, No. 3, 18-29.

Potluri, P. and Atkinson, J. (2003). Automated manufacture of composites: handling, measurement of properties and lay-up simulations. *Composites* ed. Elsevier. Vol. 34, 493-501.



## AUTHORS BIOGRAPHIES

**JOSE M. PASTOR** was born in Leon, Spain. He received the Electronics and Automatic Control Engineering degree from the Polytechnic University of Madrid (UPM) in 1991. In 1992 he begins

his doctoral research in Robotics in Construction Industry. In 1996 Dr. Pastor moves to the Engineering Department of the Carlos III University of Madrid where he is currently an Associate Professor. He received his Ph.D. in Robotics in 1997 from the UPM. His research interests include intelligent manufacturing systems, service robot applications, automation in construction and manufacturing systems simulation. His e-mail address is: [jpastor@ing.uc3m.es](mailto:jpastor@ing.uc3m.es) and the Web-page of the Inteligent Systems Lab is: <http://www.uc3m.es/islab>.



**JAVIER OTAMENDI** received the B.S. and M.S. degrees in Industrial Engineering at Oklahoma State University, where he developed his interests in Simulation and Total Quality Management. Back in his home country of Spain, he received a B.S. in Business Administration and a Ph.D. in Industrial Engineering. He is currently a simulation and statistics consultant and university professor. His e-mail address is: [jotamendi\\_30@yahoo.com](mailto:jotamendi_30@yahoo.com)

**CARLOS CORPAS** was born in Córdoba, Spain. He studied Architecture at the Polytechnic University of Sevilla and obtained his degree in 1972. He works in the technical services of DRACE group since 1974 and has 30 years experience in concrete and GRC prefabricated products. His e-mail address is: [ccr-drace-torreon@dragados.com](mailto:ccr-drace-torreon@dragados.com) and the Web-page of DRACE company is: <http://www.drace.com>.

# TOWARDS COLLABORATIVE NETWORK COMMUNICATION USING SIMULATION-BASED TRAFFIC MODEL

Teruaki Ito and Tomoyuki Hiramoto  
Department of Mechanical Engineering  
University of Tokushima  
Tokushima 770-8506, Japan  
E-mail: ito@me.tokushima-u.ac.jp

## KEYWORDS

collaborative communication, traffic simulation, traffic analysis and ETC system

## ABSTRACT

The authors have been working on a simulation-based solution for traffic issues around ETC toll plaza of expressways in Japan. Timely and detailed discussions are required to propose a better solution. However, it is not always possible to hold timely meetings because of time/distance reasons for participating members. This research proposes an approach of collaborative communication based on IP video conference system with video streaming function, which is designed to support remote audio/video communication along with simulation-based video streaming which can be shared among participating members. This paper describes the simulation model for traffic analysis around ETC toll plaza, shows the basic mechanism of IP video conference with video streaming function, and discusses the validity of our approach using some experimental results.

## INTRODUCTION

The authors have been studying ETC toll plaza issue from various respects using simulation-based traffic model approach, some of which have been reported so far. To study the issue of traffic jams and to find solutions, participating members of the project needs to get together and have discussions in a timely manner. However, the members may be physically distributed and it may not be always easy to get together in one place in a timely manner. So basically what we use in this project is a network-based collaboration approach using IP-based video conference system.

IP-based communication tool, such as video/voice conference systems, is becoming popular these days. These systems can be used quite effectively in our projects. However, it is not always easy to share the most updated information, or to interactively update the information during meetings over the network. File sharing over the network could work quite effectively in

restricted conditions. However, file sharing is not always available for security purposes. Under these circumstances, in addition to conventional network-based communication function with audio/video communication, we are developing VOD-based simulation function, in which we can timely review simulation results during our discussion.

This paper first addresses the issue of traffic jam around ETC toll plaza, a solution to which is the goal of our research. Then the paper presents the overview of our approach using simulation-based model. Describing the basic module of our collaboration system which is called CELAVIS, the paper shows the overall systems using video-on-demand simulation, and discusses the feasibility of our approach.

## SIMULATION MODEL FOR ANALYSIS OVER TRAFFIC JAMS AT ETC TOLL PLAZA

### Model of toll plaza

The model in this study covers traffic jam vehicles ranging from the starting point of traffic jams, all the way to toll plaza which is the bottle-neck of the traffic, and exit of the toll plaza. This section describes the definition of simulation model to represent the traffic jam, and some of the critical parameters used in the model. Analyzing the process of traffic based on the process oriented approach, basic 6 processes starting from the vehicle generation to toll gate exit are defined as follows.

(1) Generation of vehicle entities: Vehicle entities are generated at the starting point in the simulation either as ETC or non-ETC vehicles, of which attributes such as passenger vehicle or large commercial vehicles are assigned here to be used in the simulation process.

(2) Lane selection: One of the lanes is selected from several candidate lanes. Selection ratio is assigned to each vehicle based on predefined probabilities, vehicle types, traffic condition, etc.

(3) Vehicle travel time: The travel time for each vehicle is calculated according to the travel distance

from the starting point to the end of traffic jams, which is up to the traffic condition.

(4) Selection of toll gates: Availability of gates to each vehicle is first made clear. For example, ETC-only gate is only available to ETC vehicles, which means non-ETC vehicle cannot take ETC-only gate. Then, one of the gates will be selected.

(5) Toll payment: Any vehicles arriving at the booth occupy the gate, which means other vehicles need to wait in a queue. Job queue is used to represent the line of vehicles.

(6) Gate exit: After toll payment, vehicles are regarded to leave instantly without any traffic jams because the model covers until the gate exit.

### Input parameters for the model

#### Generation of vehicle entities

Interval of generation for vehicle entities is determined based on the assumption of the number of arriving vehicles per unit time, which is arbitrarily assigned beforehand.

$$t_i = \frac{T}{N} \quad (1)$$

$t_i$  : interval of vehicle entity generation [sec]  
 $T$  : arbitrary time [sec]  
 $N$  : estimated number of arriving vehicles

#### Interval time

The interval time is determined based on exponential distribution curve to randomly generate initial vehicles at the starting point. Each vehicle type is assigned based on some initial parameters, or stochastically assigned.

### Vehicle travel time

Vehicles travel from the initial point to toll gate under the restriction of traffics. The travel time for this movement is defined as Scheme 2.

$$T_d = \frac{[L - (L_s + L_t)]}{V_c} + \frac{L_s}{V_s} \quad (2)$$

$T_d$  : travel time using travel distance [sec]  
 $L$  : distance from starting point to toll gate [m]  
 $L_s$  : distance from traffic-end to toll gate [m]  
 $L_t$  : traffic jam length [m]  
 $V_c$  : ave. travel speed [m/sec]

$V_s$  : ave. speed during deceleration [m/sec]

While all of the parameters with one exception of traffic jam length are determined in Scheme 2, the remaining parameter of traffic jam length is determined by the calculation of travel time in simulation. When a vehicle reaches to the end of traffic jams, the expected travel time to the toll gate is calculated.

#### Service time

Service time at toll booth is determined for ETC/non-ETC vehicles respectively. The time is either set to a fixed value or derived from distribution curve.

### Simulation results

As for simulation environment to implement the model presented in the sections 2.1 and 2.2, this study adopts process simulation software called Arena 6.0 by Rockwell Software Co. Verification of simulation model was carried out based on some real traffic jams data, and significant results were obtained.

Table 1: Gates conditions of simulation

Case	The gate type to be used
Case-1	General gate*3 ETC gate*1
Case-2	General gate*3 Combined use gate*1

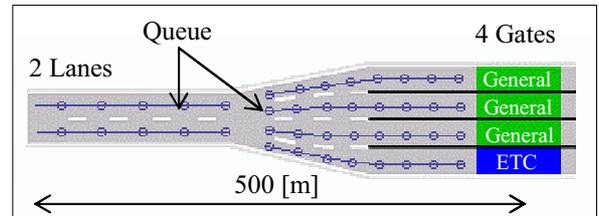


Figure 1: Example of ETC tollgate model

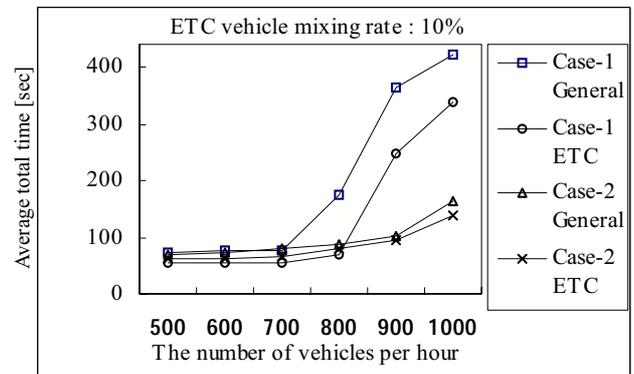


Figure 2: Comparison of average total time for the number of vehicles

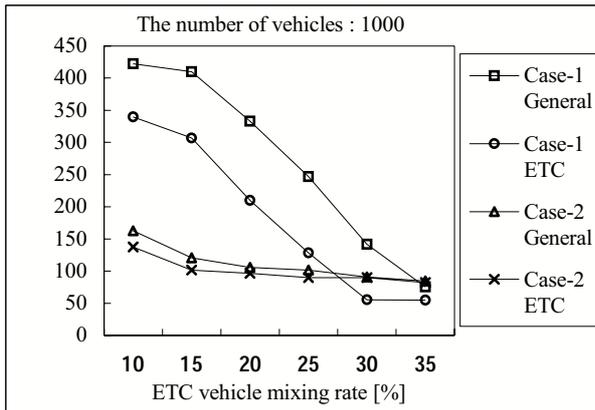


Figure 3: Comparison of average total time for ETC vehicle mixing rate

This paper shows some comparative results using several conditions based on a simple simulation model as show in Figure 1. Service time [sec] of ETC/non-ETC vehicles at the gate is determined by a triangular distribution of (14,16,18) and (3,4,5), respectively, for 1 hour simulation each.

Figure 2 shows the change curve of average travel time according to the increase of traffic volume. It shows that the travel time of Case-1 using ETC-only gate by far exceeds that of Case-1 which adopts a combination gate to comply with the traffic condition. What does this means is that introduction of ETC gate indirectly decreases the availability for non-ETC vehicles, resulting the traffic jams which affect even ETC vehicles.

Figure 3 shows the affect of ETC penetration ratio to traffic jams. If the penetration ratio reaches to 35%, travel time of Case-1 with combination gate would become shorter than that of Case-2, which indicates that 35% is a target ratio to enjoy the benefit of ETC system. This result can give a good reference in designing toll plaza.

#### IP-BASED VIDEO CONFERENCE SYSTEM (CELAVIS)

IP-based video conference system called CELAVIS (Collaborative Engineering Laboratory Video conference System) plays a central role in the communication system presented in this paper. CELAVIS is a custom-made IP-based video communication system which has been developed in CE lab (Collaborative Engineering Laboratory) at the University of Tokushima. Using CELAVIS server based on Flash Communication Server MX (FCS) of Macromedia Co., custom-made client software which we have developed enables audio/video communication based on RTMP (Real Time Messaging Protocol) protocol developed by Macromedia. Client PC requires Flash Player (Plug-in) to use CELAVIS system but no

other specific requirements are needed. This section covers the overview of CELAVIS.

One of the advantages of CELAVIS architecture is that custom-made configuration or modification can be applied to the system based on the need of users. For our research purposes, commercial system is not a good choice for this reason. As for some implemented features of CELAVIS, user/client authentication, bidirectional distribution of video images, two-way audio conversation, text chat, etc.

User/client authentication makes it possible to keep CELAVIS secure and available only to our research community. Accessible client machines are controlled by CELAVIS server to make it secure as well. Two-way audio and/or video function provides basic communication tool but it also enables conference communication for multiple sites.

Users can enjoy the service either by locally installed software or by web access. Figure 4-a shows the client program access, whereas Figure 4-b shows the web browser access. In either access, CELAVIS users can communicate using on-demand video streaming function as well as audio/video functions using typical Laptop/PC.

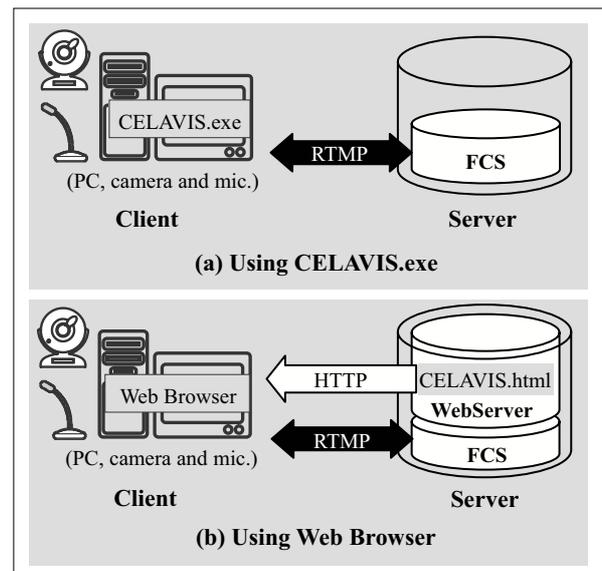


Figure 4: Client connection methods

Figure 5 shows an example for the former access. Upon authentication, the client machine loads audio/video data from microphone and CCD camera, and perform an internal process to communicate with the CELAVIS server. Active user information is managed by the server, and displayed to the user interface. Selecting an active user from the list, communication line will be established. For audio/video conference, each user makes connection to the corresponding users, which will establish conference connection for multi-users.

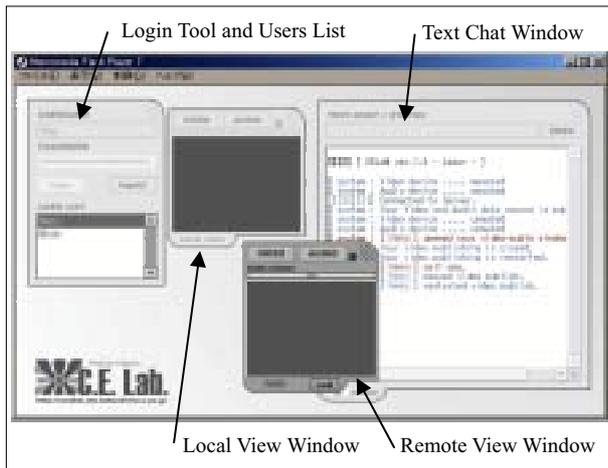


Figure 5: User interface of CELAVIS

Login tool and user list shown on the upper-left of Figure 5 show an active user list. Local and remote view windows, and text chat windows are show as well.

### SIMULATION WITH VIDEO STREAMING

For technical video meeting, sharing of on-demand simulation results as well as audio/video communication can play a critical role, which is the basic idea of this research. On-demand video streaming function was implemented using FCS server. Currently, streaming function is implemented as a separate function to CELAVIS. However, integration to CELAVIS can be available. This section covers how the video streaming function has been implemented.

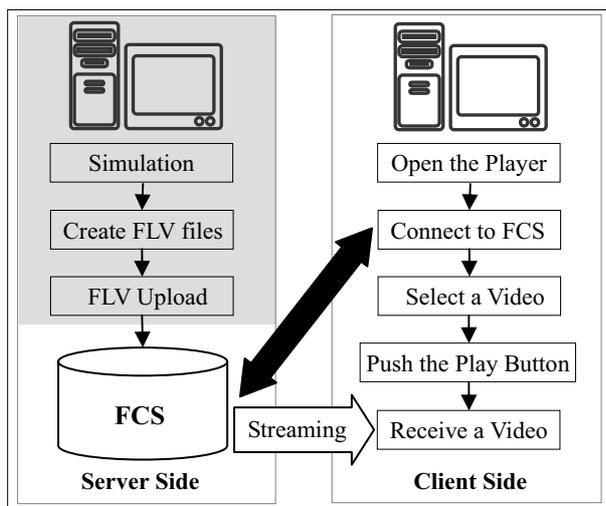


Figure 6: Workflow of server side and client side for video streaming

Figure 6 shows an overview of how to set up the video streaming function in this study. Video streaming data has been prerecorded from simulation processes and converted to FLV format so that it could be uploaded to FCS server. CELAVIS user launches client software or activates it from web site to play the streaming video. Function of player software includes basic functions, including monitor window, video selection, play/rewind/forward/stop buttons, etc.

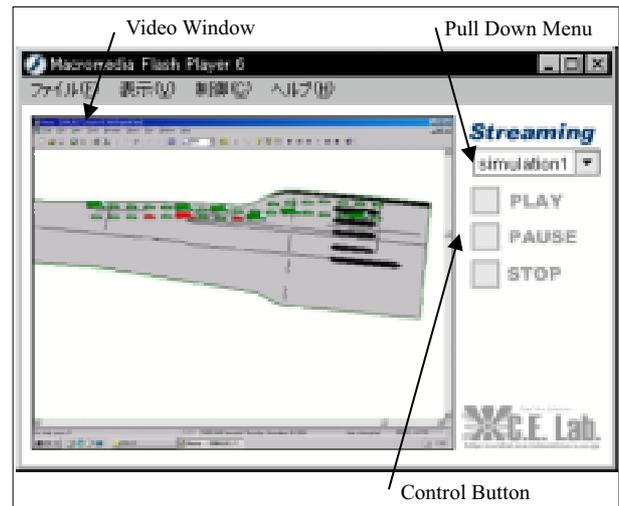


Figure 7: User interface of video streaming tool

When the player is launched, RTMP connection to FCS server is automatically established. During video conference, participants can share simulation results in the form of video streaming on-demand basis, selecting one of the videos using pull-down menu.

Figure 7 shows an example of user interface of video streaming tool which will be integrated into CELAVIS. Video window in Figure 7 shows simulation result based on traffic jams occurred around toll plaza with ETC gate. The system allows users to manipulate basic operations to review the video streaming during video conference. Parameter change in simulation requires rerunning of simulation, which means that interactive parameter change is not allowed here. However, preparation of simulation results beforehand based on several parameters makes it possible to review the results during the video conference.

As for the resolution of video streaming, low resolutions, or 72-144dpi video images were used in this study. This resolution is quite low, however, sharing of video image information did work quite well for video conference to discuss traffic issues, which is not image quality-oriented. If the topic of discussion were more image quality-oriented, for example, image processing, color reproduction, image quality, much higher resolution may be required.

## RESULTS AND DISCUSSION

This paper described the overview of process simulation model to study traffic jams occurred in ETC toll plaza. To verify the model, data collection at a real traffic jam site was performed and used in this study. Much detail regarding the simulation model has been reported.

The paper then presented the integrated test environment including CELAVIS video conference and video streaming function where simulation results can be shared for video conference participants who may be distributed globally. As for the conversation using audio/video communication, the results were quite favorable in most of the cases of video conferences held so far, in terms of quality-wise of video image and audio sound. In the meantime, image quality was not good enough to share detailed images of reference materials on video screen, such as detailed view of physical object. However, low resolution of video streaming images not only worked fine in our study, but also allows us to A few seconds of time delay was observed during conference depending upon the time of the day because of network traffics.

Video streaming was timely and effective to accelerate the sharing feeling of overview information for most of the participants. Detailed information can be more effectively shared by some alternative way such as file sharing. However, timely and effective means of information sharing, the objective of video streaming worked fine. Further study should be conducted regarding the process of video streaming preparation and their management on the web.

## CONCLUDING REMARKS

The paper described the approach of collaborative communication environment using IP-based video conference system with video streaming function. The case study in this research includes the simulation-based approach to problem solving for traffic jam issues occurring expressway toll plaza with ETC gate.

On one hand, face-to-face communication might be more natural for us than network-based communication. On the other hand, the latter provides us more extension of communication environment than the former does. The video conference results using video streaming communication showed good results, which moved this research one step forward to the collaborative communication over the network.

## ACKNOWLEDGEMENT

The authors would like to acknowledge the support and cooperation of Shikoku Branch Office of Japan Highway Public Corporation, which was provided to this research.

## REFERENCES

- Fernandes, R.J. and S. Bampi. 1998. "A software environment to integrate urban traffic simulation tasks", Proceedings of the 10 th European Simulation Symposium, October 26-28, Nottingham, United Kingdom, pp.371-377.
- Gale, C. and M.J.Oliveron and G.Silvan. 2002. "Simulation tool for managing a non-automated distribution warehouse", Proceedings of the 14 th European Simulation Symposium, October 23-26, Dresden, Germany, pp.266-270.
- Horiguchi, R. and M. Kuwahara. 2000. "A theoretical analysis for the capacity of toll plaza partially with ETC tollgates." Journal of Japan Society of Civil Engineers, No.653/IV-48, pp.29-38. (in Japanese)
- Ito, T. and T. Hiramoto. 2004. "Analysis of traffic jams using process simulation model." Vol.3, No.3, pp.371-372 (2004) (in Japanese)
- Ito, T. 2004. "An approach of whiteboard-based interface for collaborative environment." European Concurrent Engineering Conference, Hasselt, Blegium, April 19-21, pp.71-75.
- Johanson, B., A. Fox and T Winograd. 2002 "The Interactive Workspaces Project: Experiences with Ubiquitous Computing Rooms." IEEE Pervasive Computing Magazine 1(2), April-June, 71-78.
- Kiyan, K., et al. 2003. "Construction of regional cooperation activities support system combined television conference and streaming distribution." Journal for academic computing and networking, No.7, pp-33-42. (in Japanese)
- Krajzewicz, D., G. Hertkorn and P. Wagner. 2002. "An example of microscopic car models validation using the open source traffic simulation SUMO", Proceedings of the 14 th European Simulation Symposium, October 23-26, Dresden, Germany, pp.318-322.
- Milne, A. and T. Winograd. 2003. "The iLoft Project: A Technologically Advanced Collaborative Design Workspace as Research Instrument." Proc. Int'l Conf. on Engineering Design ICDE2003, Stockholm, August, pp-no.1657.
- Ponnekanti, S.R., B. Johanson, E. Kiciman and A. Fox. 2003. "Portability, Extensibility and Robustness in iROS." Proc. IEEE International Conference on Pervasive Computing and Communications (Percom2003), Dallas-Fort Worth, TX. March.

# DEPICTION OF TRANSIENT PERFORMANCE MEASURES USING QUANTILE ESTIMATION

Mirko Eickhoff,  
Donald C. McNickle,  
Krzysztof Pawlikowski

University of Canterbury  
Christchurch, New Zealand  
E-Mail: m.eickhoff@cosc.canterbury.ac.nz

## KEYWORDS

Quantile Estimation, Transient Analysis, Multiple Replications in Parallel, Selection of Several Quantiles

## ABSTRACT

For simulation output the estimation of several quantiles usually provides a deeper insight than mean value analysis. So far, quantile estimation has usually been applied to show the long run behaviour of a system. In this paper we describe a method to depict several quantiles over simulation time to show the transient behaviour. This method is based on independent replications and its capability is demonstrated by examples with different kinds of transient behaviour.

## INTRODUCTION

The purpose of steady-state simulation is to study the long-run behavior of a system. Using estimators for mean value analysis, the results of the simulation can answer questions about the average system state like: How many customers are there on average in the queue? On the other hand, quantiles are known to be more robust against outliers than mean values. Quantile estimation can also answer questions like: What is the probability of more than  $k$  customers in the queue? Questions of this kind are often of more interest to the decision-maker. The complexity of quantile estimation is higher than the complexity of mean value estimation, but the estimation of quantiles can give a deeper insight into the system of interest. This is true, especially when several quantiles are estimated. A set of several quantiles can be used to estimate the steady-state distribution function. The estimation of the steady-state distribution is the ultimate goal in steady-state simulation. For details on quantile estimation see e.g. [Heidelberger and Lewis, 1984], [Jain and Chlamtac, 1985], [Raatikainen, 1987] and [Chen and Kelton, 1999].

An extension of the estimation of several quantiles is to estimate these quantiles over model time. This provides a deep insight into the transient behaviour of the system of interest. In steady-state simulation this is useful to verify if a steady-state phase exists, i.e. that the probability distribution function of the analyzed performance measure converges to a steady-state distribution function. Furthermore

it can be verified, e.g. whether the transient behaviour is monotone or if there are some unexpected issues which demand further investigation. However, the truncation point for the estimation of steady-state performance measures should not be determined by a visual inspection of these quantiles over time (compare [Bause and Eickhoff, 2003]).

In applications finite-horizon simulation is frequently used to examine a given situation with a certain initial state. In contrast to steady-state simulation the transient behavior of the system is the central point of analysis, even though, the usual approach is to estimate only mean values. Again, the estimation of several quantiles over time provides a deeper insight and extends the results of commonly used approaches. For a comparison of finite-horizon simulation and steady-state simulation see e.g. [Law and Kelton, 2000] and [Alexopoulos and Kim, 2002].

The main problem in quantile estimation for steady-state performance measures is that the output data  $X_1, X_2, \dots$  of a single simulation run is typically not stationary and is autocorrelated (see e.g. [Lee et al., 1999]). Therefore, the amount of required output data can be immense, which causes a problem when storing and sorting the output data. Using  $p$  independent replications of the simulation is a well known approach to obtain independent sequences of output data. If these replications are synchronized (see [Bause and Eickhoff, 2002]) an independent and identically distributed (iid) random sample  $\{x_{j,i}\}_{j=1}^p$  of  $p$  observations of  $X_i$  is available at each observation index  $i$ . This property helps to overcome the main problem of quantile estimation in a single simulation run and enables the use of traditional quantile estimators for iid random samples.

Let  $F_i(x) = \Pr\{X_i \leq x\}$  denote the cumulative probability distribution function of  $X_i$ . The  $q$ -quantile at observation index  $i$  is defined by the equation  $q = F_i(x_q)$  and, therefore,

$$x_q = \inf\{x : F_i(x) \geq q\} = F_i^{-1}(q)$$

is the location of the  $q$ -quantile in the case of a continuous distribution  $F_i(x)$ . Let  $\{y_{j,i}\}_{j=1}^p$  be the order statistic of  $\{x_{j,i}\}_{j=1}^p$ . A valid estimator for the location of the  $q$ -quantile at observation index  $i$  is given by

$$\hat{x}_q = y_{\lceil pq \rceil, i}$$

The half width of a confidence interval of  $\hat{x}_q$  can be described in two ways:

$$\hat{x}_q \in x_q \pm \epsilon'_q \quad \text{and} \quad \hat{x}_q \in x_q \pm \epsilon_q$$

$\epsilon'_q$  describes an interval in the range of the measure and  $\epsilon_q$  describes an interval in the range of the probability (see [Chen and Kelton, 1999]). Note, the interval  $q \pm \epsilon_q$  should not exceed the bounds 0 and 1. In nonparametric statistics  $\epsilon'_q$  can be calculated from

$$\Pr\{y_{l,i} \leq x_q < y_{u,i}\} = 1 - \alpha_{l,u} \quad (1)$$

$$= \sum_{j=l}^{u-1} \binom{p}{j} q^j (1-q)^{p-j}$$

by decreasing  $l$  and increasing  $u$  until the chosen confidence level  $(1 - \alpha) \leq (1 - \alpha_{l,u})$  is reached (see [Conover, 1999] and [Heidelberger and Lewis, 1984]). In [Chen and Kelton, 1999] is shown that  $\epsilon_q$  can be chosen from the inequality

$$p \geq \frac{z_{1-\alpha/2}^2 q(1-q)}{\epsilon_q^2} \quad (2)$$

where  $z_{1-\alpha/2}$  is the  $1 - \alpha/2$  quantile of the standard normal distribution. Both, Equation (1) and Inequality (2) do not depend on the output data itself. Therefore, both formulas can be used to estimate the half width before the simulation experiment starts. However, both formulas mainly depend on the number of replications  $p$ , because the confidence level  $1 - \alpha$  can be considered as a constant parameter. Therefore,  $p$  is the most important parameter in the methods described in subsequent sections.

To show the transient behavior of the system of interest a plot of several quantiles over time is needed. The quantiles should be chosen with non overlapping confidence intervals. Therefore, a method is needed which determines adequate quantiles based on parameter  $p$ , because the half width of the confidence interval of  $\hat{x}_q$  depends on the number of replications  $p$ . In the following section two alternative methods are proposed and discussed. The better method is used to examine examples with a variety of different transient behaviors. In the last section some conclusions are given.

## SELECTION OF QUANTILES

As already pointed out, the calculation of the confidence interval based on Equation (1) and Inequality (2) does not depend on the output data itself, but on the number of replications  $p$ , the confidence level  $1 - \alpha$  and  $q$  itself. Because the confidence level can be considered as a given parameter the main question is: How to choose several  $q$ -quantiles as a function of  $p$ ? The basic idea of the algorithms described in this section is to choose the 0.5-quantile as the starting point and to choose all other quantiles in a way that their confidence intervals do not overlap. A larger number of replications involves smaller confidence intervals and this enables the selection of more quantiles with non overlapping confidence intervals.

Our first method is based on Equation (1). In the beginning the first quantile 0.5 is given and its confidence interval is calculated by extending  $l$  and  $u$  until the wanted confidence level  $1 - \alpha$  is reached.  $l$  and  $u$  describe the indexes in  $\{y_{j,i}\}_{j=1}^p$  of the bounding values of the confidence interval. The selection of the next two quantiles which have a non overlapping and non disjoint confidence interval is not straight forward, because Equation (1) has no closed form.

Therefore, we perform two binary searches in the directions above and below 0.5. The binary search in the direction below 0.5 stops if a quantile is found with a upper bound  $u'$  being equal to  $l$ . Analogously, the binary search in the upper direction stops if a quantile is found with a lower bound  $l'$  equal to  $u$ . The result of these binary searches are the next displayed quantiles. The binary searches are repeated, until it is not possible to find another quantile with a confidence interval in the unprocessed area between the last  $l$  and 1 (resp.  $u$  and  $p$ ). This calculation can be performed before the simulation experiment starts and, therefore, the run time of this method does not really matter. For convenience a linear search, leading to a worse run time, could be performed instead of the binary search.

Equ. (1)		Inequ. (2)	
$p = 100$	$p = 1000$	$p = 100$	$p = 1000$
$q (l;u)$	$q (l;u)$	$q (q \pm \epsilon_q)$	$q (q \pm \epsilon_q)$
			.003 ( 0;.006)
	.010 ( 5; 16)		.012 (.006;.018)
	.024 ( 16; 32)		.026 (.018;.034)
	.042 ( 32; 53)		.045 (.034;.056)
.08 ( 3;13)	.066 ( 53; 79)	.09 (.04;.13)	.069 (.056;.082)
	.094 ( 79;110)		.098 (.082;.113)
	.127 (110;145)		.131 (.113;.148)
.19 (13;26)	.164 (145;184)	.20 (.13;.26)	.167 (.148;.187)
	.205 (184;226)		.208 (.187;.230)
	.250 (227;273)		.252 (.230;.274)
.34 (26;42)	.297 (273;321)	.34 (.26;.42)	.298 (.274;.322)
	.346 (321;371)		.346 (.322;.371)
	.396 (371;422)		.397 (.371;.422)
	.448 (422;474)		.448 (.422;.474)
.5 (42;59)	.5 (474;527)	.5 (.42;.58)	.5 (.474;.526)
	.553 (527;579)		.552 (.526;.578)
	.604 (579;630)		.603 (.578;.629)
	.655 (630;680)		.653 (.629;.678)
.67 (59;75)	.704 (680;728)	.66 (.58;.74)	.702 (.678;.726)
	.751 (729;774)		.748 (.726;.771)
	.795 (774;817)		.792 (.771;.813)
.81 (75;88)	.836 (817;856)	.80 (.74;.87)	.833 (.813;.852)
	.873 (856;891)		.869 (.852;.887)
	.906 (891;922)		.902 (.887;.918)
.93 (88;97)	.935 (922;948)	.91 (.87;.96)	.931 (.918;.944)
	.958 (948;969)		.955 (.944;.966)
	.977 (969;985)		.974 (.966;.982)
	.990 (985;996)		.988 (.982;.994)
			.997 (.994; 1)

Table 1: This table shows the selected quantiles with their confidence intervals chosen by the method based on Equation (1) and by the method based on Inequality (2) with  $1 - \alpha = 0.9$  and  $p = 100$  resp.  $p = 1000$ .

The first two columns of Table 1 show the rounded result of this method for  $p = 100$  and  $p = 1000$  independent replications with a confidence level of  $1 - \alpha = 0.9$ . The method selects 7 quantiles for  $p = 100$  and 27 quantiles for  $p = 1000$ . The values in brackets show the  $l$  and the  $u$  index of the quantile as defined in Equation (1).

The second investigated method is based on Inequality (2). Again, the starting point is the 0.5-quantile and the method searches for more quantiles in the directions below and above 0.5. In this case a binary search is not needed, because the next quantile can be calculated directly with the help of Inequality (2) and the following condition:

$$q_k - \epsilon_{q_k} = q_{k+1} + \epsilon_{q_{k+1}} \quad (3)$$

This condition is valid for the direction below 0.5, a condition for the direction above 0.5 can be formulated analogously.  $q_k$  is given and  $\epsilon_{q_k}$  can be calculated by Inequality

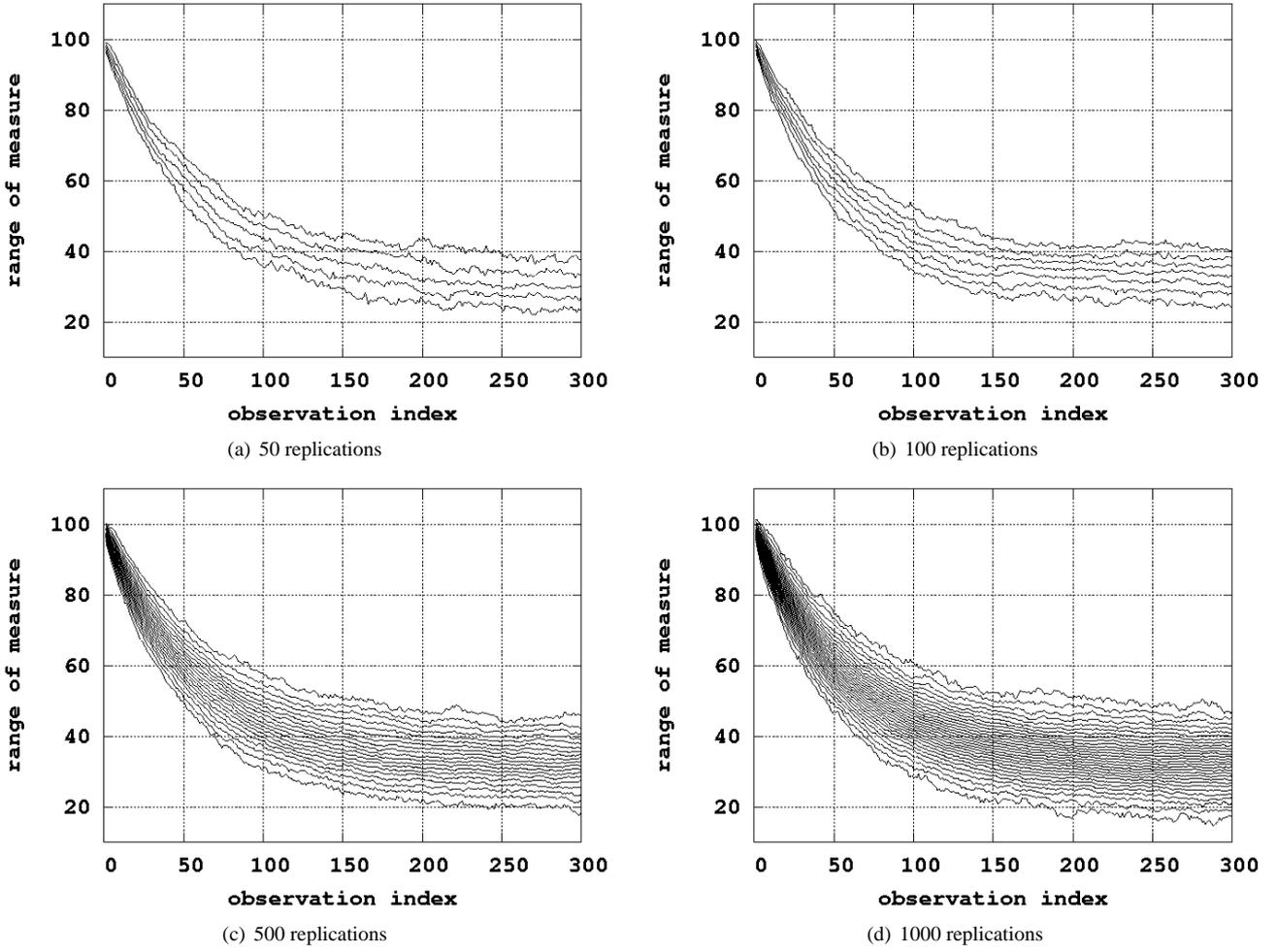


Figure 1: Several quantiles over time: ARMA process.

(2). Therefore, the substitution  $a_k = q_k - \epsilon_{q_k}$  is calculable right away. Equation (3) can be transformed to:

$$a_k = q_{k+1} + z_{1-\alpha/2} \sqrt{\frac{q_{k+1}(1 - q_{k+1})}{p}}$$

Eliminating the square root leads to

$$0 = q_{k+1}^2 b + q_{k+1} c_k + d_k$$

with  $b = \frac{1}{z_{1-\alpha/2}^2} + \frac{1}{p}$ ,  $c_k = -\frac{2a_k}{z_{1-\alpha/2}^2} - \frac{1}{p}$  and  $d_k = \frac{a_k^2}{z_{1-\alpha/2}^2}$ . Finally,  $q_{k+1}$  can be calculated by

$$q_{k+1} = \frac{-c_k - \sqrt{c_k^2 - 4bd_k}}{2b}. \quad (4)$$

Equation (4) is valid for quantiles below 0.5. An equation for quantiles above 0.5 can be derived analogously. The search should be continued until the bounds of the interval  $[0, 1]$  are exceeded.

The rounded results of the second method are shown in the last two columns of Table 1. For  $p = 100$  the second method selects 7 quantiles and for  $p = 1000$  this method selects 29 quantiles. The values in brackets show the confidence interval of the belonging quantile in the range of the probability.

The results of the first and the second method are comparably accurate. However, the binary search of the first method seems to be circumstantial compared to the direct

calculation by Equation (4) in the second method. Furthermore, the calculation of  $\binom{p}{j}$  in Equation (1) involves the handling of very small and very big values. This might lead to problems in computer calculations and rounding errors. Therefore, we recommend the second of the described methods. All depictions in subsequent sections use quantiles selected by the second method.

## EXAMPLES

In the previous section we described how to select several quantiles. In this section we use these selected quantiles to depict stochastic processes with well known statistic properties. This is useful to validate the results as in [Bause and Eickhoff, 2003]. The implementation of the stochastic processes is based on the random number generator described in [L'Ecuyer et al., 2002]. Because this generator allows the choice of many substreams, it is suitable for many independent replications. As already mentioned in the introduction, the independent replications of the stochastic processes are used to collect a set of independent data for each  $X_i$  with  $1 \leq i \leq \infty$ . To realize these stochastic processes the random numbers are transformed as follows.

**ARMA Process:** The behaviour of the first stochastic process is comparable with the behaviour of a storage in an inventory system. It is an ARMA(5, 5) process which is

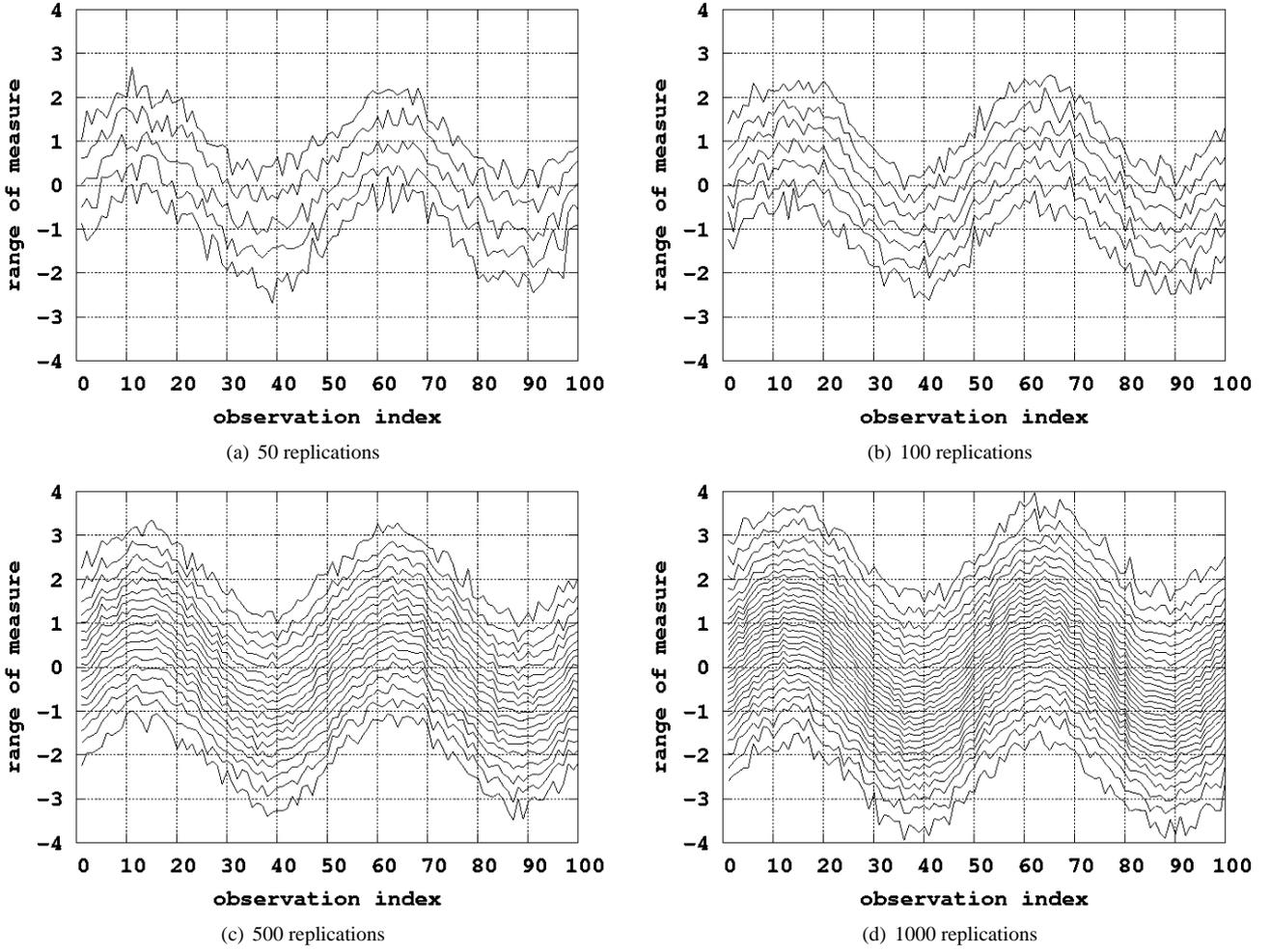


Figure 2: Several quantiles over time: periodic process.

defined by

$$X_i = 1 + \epsilon_i + \sum_{k=1}^5 \frac{1}{2^k} (X_{i-k} + \epsilon_{i-k}), k \geq 0$$

with the starting condition  $X_{-5} = X_{-4} = X_{-3} = X_{-2} = X_{-1} = 100$ .  $\{\epsilon_i\}_{i=1}^{\infty}$  is an independent Gaussian white noise process ([Hamilton, 1994]). We selected four sets of quantiles for  $p = \{50, 100, 500, 1000\}$  independent replications. The results are shown in Figure 1.

**Periodic Process:** The second observed process has a periodic behaviour and is defined by

$$X_i = a \cdot \sin(\omega i) + \epsilon_i$$

The cycle length of the sine oscillation is given by  $T = \frac{2\pi}{\omega}$  with the amplitude  $a$ . And again  $\{\epsilon_i\}_{i=1}^{\infty}$  is an independent Gaussian white noise process. This process is depicted in Figure 2 for  $p = \{50, 100, 500, 1000\}$  independent replications.

**Exponential Process:** The behaviour of the third stochastic process is comparable with the behaviour of a buffer in a queueing system. It is defined by

$$X_i = \epsilon'_i \cdot b(1 - e^{i \frac{\ln(0.05)}{i}}).$$

The process  $\{\epsilon'_i\}_{i=1}^{\infty}$  is similar to the independent Gaussian white noise process, but its distribution is exponential (see [Law and Kelton, 2000]) with  $\beta = 1$ . The parameter

$b$  stretches the distribution. The part in brackets of the formula causes that the process is slowly converging towards its marginal distribution. This is depicted in Figure 3 for  $p = \{50, 100, 500, 1000\}$ .

Of course, a smaller value of  $p$  leads to a smaller set of selected quantiles. Additionally, the quantiles of the smaller set seem to fluctuate more. Further more, the quantiles of areas with lower probability fluctuate more than the ones of high probability. In Figure 1 and Figure 2 this can be observed when comparing the bounds 0 and 1 with the center (around 0.5) of the distribution. Because the distribution in Figure 3 is not symmetrical, the quantiles at bound 1 fluctuate more than the ones at bound 0.

These examples show, that this approach of depicting quantiles is suitable for both symmetrical and asymmetrical distributions, as well as converging and non converging processes. However, we recommend to use at least 50 independent replications. This makes sure that the selected set of quantiles includes at least 5 different quantiles.

**Random Walk:** The last examined process behaves like a random walk between 0 and 100. The random walks are not stopped at this thresholds, but all values higher (resp. lower) than these bounds are reduced to these values (compare [Bause and Beilner, 1999]). The peculiarity of this process is the constant value of the mean (e.g. the 0.5-quantile), whereas all other quantiles are not constant and tend to the thresholds 0 and 100 (see Figure 4(a) and 4(b)). Therefore, the cumulative distribution function is in the be-

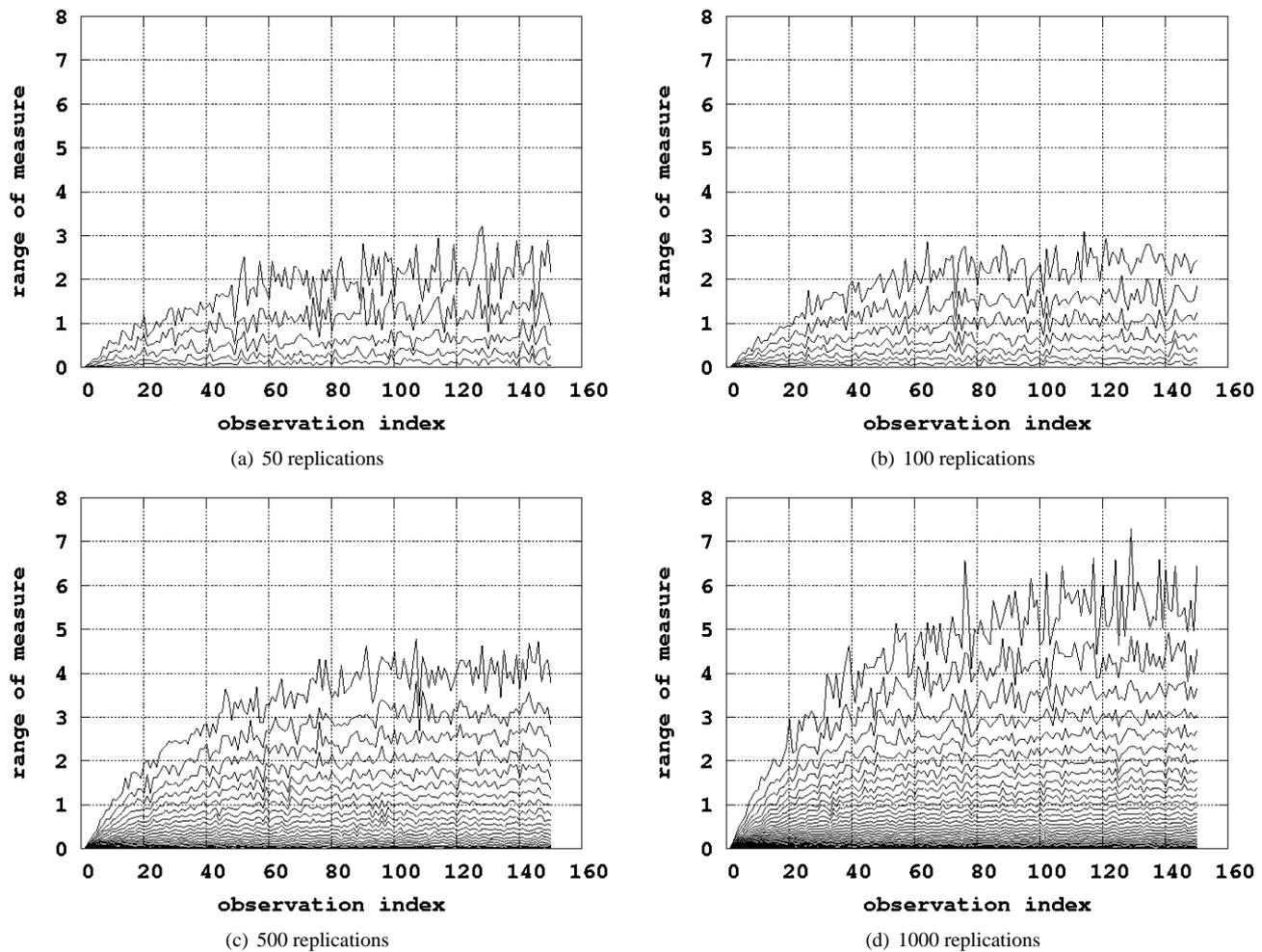


Figure 3: Several quantiles over time: exponential process.

ginning very steep around the 0.5-quantile (Figure 4(c)), but after a long simulation time it is very flat (Figure 4(d)). Analysis of only mean values would show a constant behaviour, even though this example is transient and the cumulative distribution is slowly converging to its marginal distribution.

## CONCLUSIONS

We described two methods of selecting quantiles. Both methods delivered similar results. However, for further investigation we decided to use the method based on Inequality (2) because its complexity is lower.

This approach to depict several quantiles over time appears suitable for a variety of different transient behaviours. We recommend to use at least 50 independent replications to make sure, that the selected set of quantiles is reasonably large. In finite-horizon simulation the replications do not need to be processed in parallel. Therefore, a large number of replications, e.g.  $p = 1000$ , is feasible.

Our last investigated example shows, that the analysis of several quantiles provides a deeper insight into the analyzed process than mean value analysis. Drawing conclusions entirely based on mean value analysis is not recommendable for complex models.

## REFERENCES

- [Alexopoulos and Kim, 2002] Alexopoulos, C. and Kim, S. (2002). Output data analysis for simulations. *Proceedings of the 2002 Winter Simulation Conference*, pages 85–96.
- [Bause and Beilner, 1999] Bause, F. and Beilner, H. (1999). Intrinsic problems in simulation of logistic networks. *Proc. of the 11th European Simulation Symposium and Exhibition (ESS99)*, pages 193–198.
- [Bause and Eickhoff, 2002] Bause, F. and Eickhoff, M. (2002). Initial transient period detection using parallel replications. *Proc. of the 14th European Simulation Symposium*, pages 85–92.
- [Bause and Eickhoff, 2003] Bause, F. and Eickhoff, M. (2003). Truncation point estimation using multiple replications in parallel. *Proc. of the Winter Simulation Conference*, pages 414–421.
- [Chen and Kelton, 1999] Chen, E. J. and Kelton, W. D. (1999). Simulation-based estimation of quantiles. *Proceedings of the 1999 Winter Simulation Conference*, pages 428–434.
- [Conover, 1999] Conover, W. (1999). *Practical Nonparametric Statistics*. John Wiley & Sons, Inc., New York.

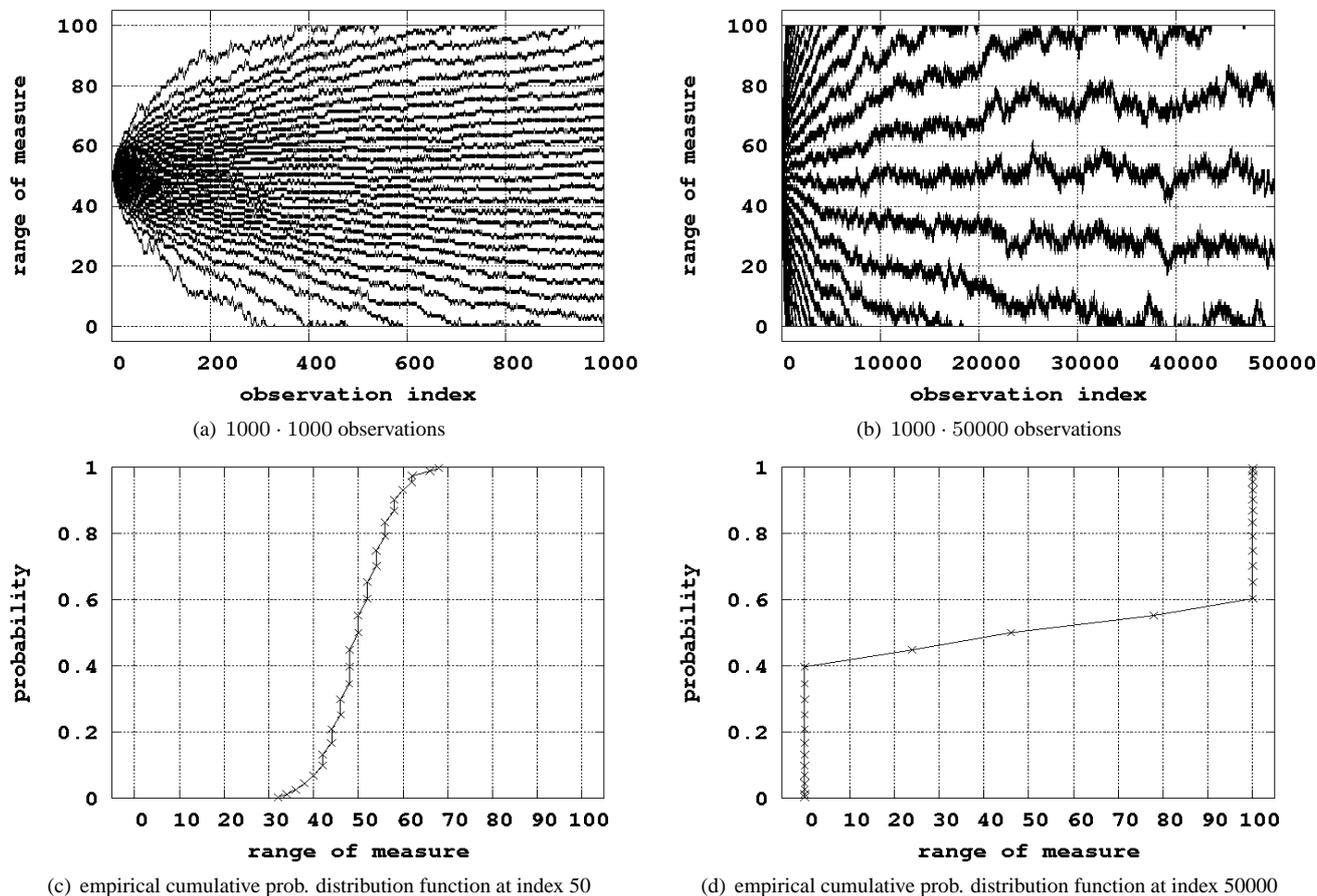


Figure 4: Constant mean value, but other quantiles are changing over time.

[Hamilton, 1994] Hamilton, J. D. (1994). *Time Series Analysis*. Princeton University Press.

[Heidelberger and Lewis, 1984] Heidelberger, P. and Lewis, P. (1984). Quantile estimation in dependent sequences. *Operations Research*, 32(1):185–209.

[Jain and Chlamtac, 1985] Jain, R. and Chlamtac, I. (1985). The  $P^2$  algorithm for dynamic calculations of quantiles and histograms without storing observations. *Communications of the ACM*, 28(10):1076–1085.

[Law and Kelton, 2000] Law, A. M. and Kelton, W. D. (2000). *Simulation Modeling and Analysis*. McGraw-Hill Higher Education, New York.

[L'Ecuyer et al., 2002] L'Ecuyer, P., Simard, R., Chen, E. J., and Kelton, W. D. (2002). An object-oriented random-number package with many long streams and substreams. *Operations Research*, 50(6):1073–1075.

[Lee et al., 1999] Lee, J.-S. R., McNickle, D., and Pawlikowski, K. (1999). Quantile estimation in sequential steady-state simulation. *Proceedings of the 13th European Simulation Multiconference*, pages 168–174.

[Raatikainen, 1987] Raatikainen, K. E. E. (1987). Simultaneous estimation of several percentiles. *SIMULATION*, 49(4):159–164.

## AUTHOR BIOGRAPHIES

**MIRKO EICKHOFF** holds a Diploma degree in Computer Science from the University of Dortmund. His research interests are in the area of output analysis of discrete event simulation using multiple replications. Currently he is a Ph.D. Candidate of the Simulation Research Group of the University of Canterbury. For details see <http://www.cosc.canterbury.ac.nz/research/PG/mei16>. His e-mail address is [m.eickhoff@cosc.canterbury.ac.nz](mailto:m.eickhoff@cosc.canterbury.ac.nz).

**DONALD C. MCNICKLE** is an Associate Professor of Management Science in the Management Department at the University of Canterbury. His research interests include queueing theory, networks of queues and statistical aspects of stochastic simulation. He is a member of INFORMS and the Operational Research Society. His e-mail address is [don.mcnickle@canterbury.ac.nz](mailto:don.mcnickle@canterbury.ac.nz).

**KRZYSZTOF PAWLIKOWSKI** is a Professor in Computer Science at the University of Canterbury, in Christchurch, New Zealand. The author of over 130 research papers and four books; has given invited lectures at over 80 universities and research institutes in Asia, Australia, Europe and North America. Alexander-von-Humboldt Research Fellow (Germany) in 1983-84 and 1999. His research interests include performance modelling of telecommunication networks, discrete-event simulation and distributed processing. Senior Member of IEEE, member of ACM and SMSI. His e-mail address is [krys.pawlikowski@canterbury.ac.nz](mailto:krys.pawlikowski@canterbury.ac.nz).

# DATA COLLECTION FOR SYSTEMS OF PRODUCTION SIMULATION

Zenobia WEISS and Maria PIŁACIŃSKA  
Division of Industrial Information Systems  
Poznan University of Technology  
60-965 Poznan, Poland  
E-mail: maria.pilacinska@put.poznan.pl

## KEYWORDS

production simulation, PPC system, data collection

## ABSTRACT

Despite the usefulness of computer simulation in manufacturing, its prevalence in the area of production preparation and management is still limited. The main reason for this situation is the difficulty with automatic data collection. In the paper the proposal of methodology for the analysis of data accessible in the information systems of an enterprise and required for the purposes of simulation model building is presented. Conclusions from an analysis of a vehicle manufacturer using the presented methodology are also described. For simulation of the production process, the usefulness of data which can be obtained from digital sources seems to be much greater if the information systems for production resource planning are, at least, MRP II class systems. The function of the methodology was successful in the case discussed, but a more detailed practical verification based on a more intricate analysis is required. Results of the study confirm the necessity for commercial enterprises, as well as academic institutions, to be involved in the search for solutions in the area of automatic data collection for simulation purposes.

## INTRODUCTION

In many commercial and academic enterprises computer simulation is now one of the fundamental research or training tools. However, in some areas, such as manufacturing, computer simulation is not yet particularly prevalent, although it is becoming more and more widely used. In the area of manufacturing, computer simulation is mostly applied by designers, constructors and technologists, while its potential is much less frequently used to optimize the processes of production preparation and management.

Such a situation is exacerbated by the lack of suitably educated and experienced analysts, as well as the fact that the tools for the simulation of the production processes are separate applications which are not embedded in the systems of production planning and control (PPC). The process of manual data collection, and the input of the data into the database of the simulation tool, in order to build a model, is very laborious and time-consuming. It is the basic obstacle

to the conducting of simulation projects. This obstacle has two dimensions: economical (time, costs); and psychological (a long wait for results, which may not live up to expectation).

Experienced simulation analysts use models built during one project for working on another. In a survey, conducted by Robertson and Perere during the Winter Simulation Conference in 1999, it was found that 27% of respondents reuse previously built models for a different purpose, while 42% keep the model updated. The poll also revealed that:

- 60% of respondents manually input the data to the model,
- some companies rely 100% on the manual method, whereas others rely 100% on a link to an external system (Robertson and Perere 2001).

The research of Robertson and Perere, observations of realities in which production companies are operating, and demands from industry, as well as the academic environment, indicates the need for an automatic data collection method for simulation models and such automatic data collection could substantially contribute to an increase in the use of simulation in production management. The basic electronically stored data sources which can be used for building production process simulation models are PPC systems (or production planning and control modules of integrated systems).

The awareness of the importance of the integration of Enterprise Resource Planning (ERP) systems with external applications is increasing among producers and efforts are being made to find solutions.

For data which is necessary for the modelling of the production process and which can be exported from a PPC system, the technological process and the current or planned machine capacity must first be dealt with. The automatic exchange, vis-à-vis the integration of simulators with information systems for production management, is a difficult problem which requires a detailed analysis of the data structures, an investigation of the technical alternatives for their exchange and a one by one programming of an interface for each different variation.

This paper presents a methodology for the analysis of data accessible in the information systems of an enterprise and required for the purposes of building a production simulation model. Conclusions from an analysis of a vehicle manufacturer using the presented methodology are also described.

## **METHODOLOGY FOR THE ANALYSIS OF DATA REQUIRED FOR PURPOSES OF SIMULATION**

As mentioned, a step by step process of data collection during the building of a simulation model is very laborious and time-consuming and therefore a decision in an enterprise to look for a digital data source is economically justified. The work invested in the identification of such data sources and an assessment of their adequacy, in the majority of cases, pays off and seldom leads to the conclusion that manual data collection is necessary because of an absence of suitable digital data sources. However, even where such a search is not successful, to judge the time spent on the identification and analysis of potential digital data sources as lost time is, at the least, debatable. An analytically proven absence of digital data sources for simulation purposes can be judged as essential knowledge for model building.

An algorithm of an analysis of potential data sources for the simulation of production processes will be presented below. It can be applied in simulation projects relating to the design of production systems as well as in projects relating to the management of production systems on the operational level.

### **Assumptions**

The model builder and/or project team will have performed the following activities before they start the analysis of digital data sources:

- a) they will have established the assumptions relating to model building (simulation aim, limits of modelling system, level of details in the model),
- b) they will have specified what data is required to build the model,
- c) they will have identified the key data sources (e.g. paper documents, knowledge of employees, observations of real system's behaviour, digital sources).

After the assumptions for the simulation project have been established, the analysis of the potential data sources can begin. The first step will be an analysis of whether the information systems supporting the production management mirror the processes involved in simulation research, and, if so, to what extent. This will be followed by an investigation of the nature of the data stored and processed by the analyzed systems and responsible for describing selected processes or sub-processes. If we are dealing with the application of simulation in the area of *System Management* it is also necessary to investigate the frequency of data modification in comparison to the level of details in the simulation model. The next step is a detailed analysis of the identified data, which will include structure, type, optionality, completeness and redundancy. For each step the results obtained form the input data for the following step. Finally, an auxiliary database can be created where metadata i.e. data about data

(Gramacki and Darulewski 2001) is stored. This tool is useful for the storing, in a structured form, of the results of the detailed analysis of the data which is accessible in the information systems of an enterprise and which is necessary to build a simulation model. Information relating to the necessary modifications and additions to the data can also be stored in such a database.

Each of the described analytical steps increases the accuracy of the knowledge of an analyst or project team relating to such issues as the required data and to what extent this data can be automatically collected. The analytical algorithm is explained in more detail in figure no. 1 (next page).

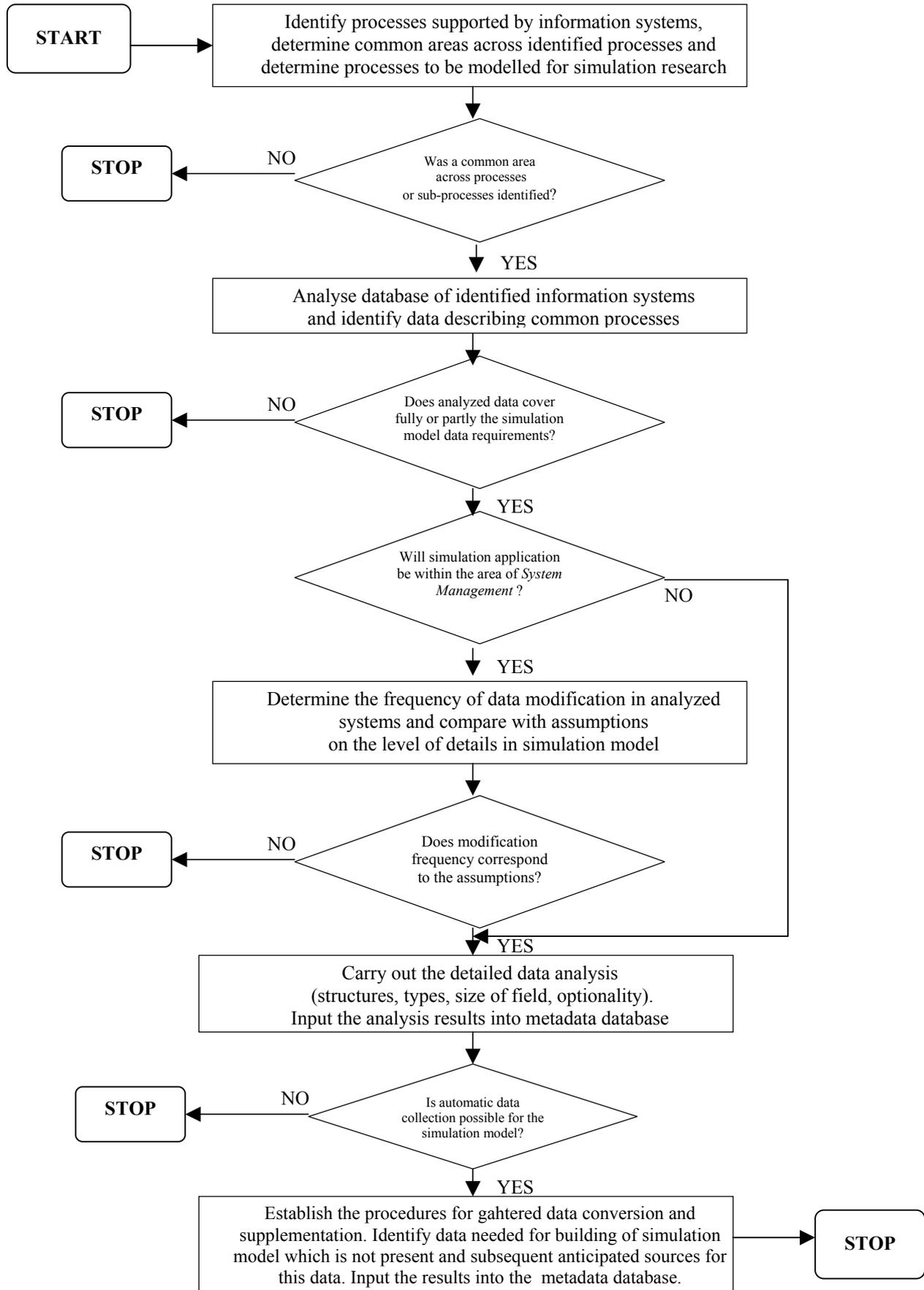
The end result of the method is a database containing metadata, which supplies us with information on what data can be collected automatically and what data must be entered manually into the model. The database also contains information on the procedures which must be executed in order to convert or complete data, or to remove redundant data.

## **DATA ANALYSIS FOR SIMULATION PURPOSE – CASE STUDY**

### **Data Analysis**

The methodology as presented above, excluding the setting-up of a metadata database, was applied to a vehicle manufacturer for the analysis of data accessible in the production management information systems.

In this enterprise, because of the uniqueness of the production process, the manufacturing process is not automated and technological operations are conducted manually. This is a classic example of unit production to customer order. The end product is one of the possible variants of an earlier defined vehicle model, which fulfils customer requirements for construction and technology. The complexity of the production process, and subsequently the complexity of the production management, means that the enterprise is seeking new tools for production planning and its optimization, and intensive work is being carried out to apply discrete event simulation. At the moment, the company uses some systems of production planning and control which can form the potential data source for the building of the simulation model. As the result of the first analytical step at the process level two information systems emerged for further analysis.



Figures 1: The Algorithm of Analysis of PPC Systems Data for Simulation Purposes

System no. I is on the Ms Access® database based tool, which is used for the archiving of vehicles' production plans as well as for the generation of new plans on the basis of historical data. The system is used for operations which include estimating the time for the production process based on the times of the singular technological operations needed to perform the planning of manufacturing resources which are involved during the production process. System no. II is the company dedicated, multiaccessible software for production planning and control which works in the client-server architecture via the Internet. The system is used for the management of manufacturing plans, the estimation of production costs and the planning of material requirements. The system database contains detailed data on customer orders and production plans including times for the release of production orders. The data analysis for the two systems was worked out by utilising the re-engineering concept. The results of the data organisation analysis were documented with entity-relationship diagrams, which usually document the first step of the database design process, the analysis of the so called "miniworld" (figures 2 and 3).

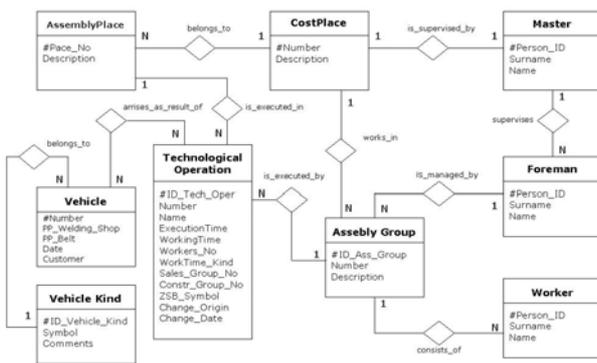


Figure 2: Entity-relationship Diagram from the 1<sup>st</sup> System

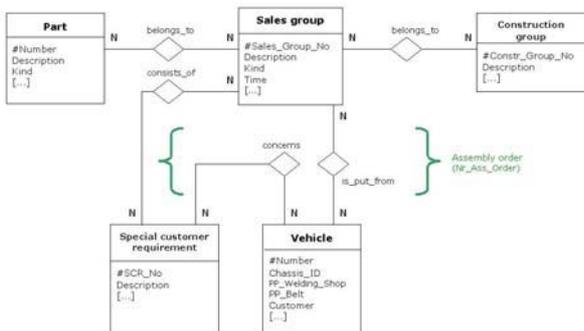


Figure 3: Entity-relationship Diagram from the 2<sup>nd</sup> System

The rough data analysis of both systems revealed that the production process is perceived from a different point of view by each of the two systems. The logic of the data in system no. I reveals features of the process approach. Production data is interpreted by the system through the prism of technological operations, which

are treated as the basic unit of production planning, and the production process is modelled as the sequence of these operations (the entity *Technological Operation* as shown in figure 2). Conversely, the logic of the data in system no. II primarily takes into consideration the construction of the vehicle, and the production process is modelled by means of the view of the end product amalgamation of the objects which represent construction components such as the engine, the driveshaft, or the air-conditioning ( the entity *Sales Group* as shown in figure 3).

Time, which is the key parameter in the modelling of the production process, is an attribute of the *Technological Operation* entity in the first system, whilst, in the second system, time is an attribute of the *Sales Group*. The linking of the time parameter to the *Sales Group* was based on intuition and estimation, whereas the linking of the time parameter to the *Technological Operation* was the result of executed mensuration. The data relating to time and resident in system no. II is unhelpful to the modelling of the production process because the image of the final end product amalgamation of the construction objects does not correspond to the reality of the production process in which components of one *Sales Group* are assembled into the vehicle in assembly process stages which are not always sequential.

Data which is stored and processed by the first system describes many of the production process parameters which are essential from the simulation's point of view. However, they are not complete and need to be converted and completed.

### Simulations as Verification of Data Collected from Digital Sources

On the basis of data identified in the database of system no. I, the simple models of the production sub-processes were built and used to verify the data. The behaviour of the model was compared with the observation of the real system.

The constructed models mirrored the painting processes (Skrzypczak 2004) and the finishing process (Wesołowski 2004) in the vehicle factory. Because of the highly diverse forms of organization in both cases, different modelling and simulation tools were used. In the case of the paint shop where there were not a large number of variants in the production process, and where the random factor had a small influence, the simulation model was built using a spreadsheet. In the area of finishing, where there was a high level of dynamism and a large number of possible process variations, as well as the large influence of disruptions from the earlier stages of the manufacturing process, and a large random factor, in order to model the production system behaviour, the discrete event simulator Arena® was applied.

The diagrams below illustrate example models for data verification (figures 4 and 5).

Process hour		PAINT SHOP									
		okle podlogi	szlifowanie	odtłuszczenie	oklepanie	lak glęboki	szlifowanie	odtłuszczenie	zawieszenie pociągów		
		Pace 1 i 2	Pace 3 i 4	Pace 5	Pace 6	Pace 7	Pace 8	Pace 9	Pace 10		
X37-0005		--	--	--	--	--	--	--	X37-0005		
Y 4516		--	--	--	--	X 4516	--	--	--		
X21-3001		--	--	--	X21-3001	--	--	--	--		
malowanie pod maski		oprawy	plac	lakier pasów	szlifowanie	odtłuszczenie1	odtłuszczenie2	blisko lak kofe.			
		Pace 11	Pace 12	Pace 13	Pace 14	Pace 15	Pace 16	Pace 17	Pace 18		
		--	--	--	--	--	--	--	--		
lak kofce		szlifowanie	odtłuszczenie	plac kofce	oklepanie	konserwacja	plac				
		Pace 19	Pace 20	Pace 21	Pace 22	Pace 23	Pace 24	Pace 25			
		--	--	--	--	--	--	--			

Figure 4: Model of Painting Process for some Types of Vehicle

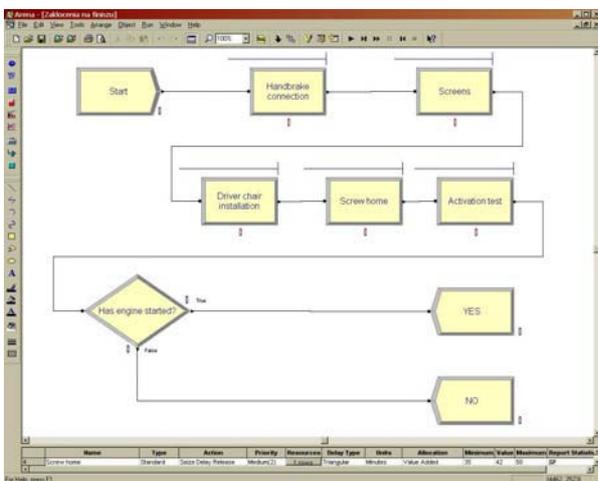


Figure 5: Model of the Finishing Sub-process in the Vehicle Factory

The results obtained reflect the conclusions drawn on the basis of real system observation, which confirms the adequacy and correctness of the data used for simulation.

In the case as discussed, the data analysis resulted in the conclusion that only a fraction of the data, after conversion and supplement, can be used to build a simulation model. In the opinion of the authors, the reason for this is the fact that the PPC systems as used by the vehicle company do not meet the MRP II standard. Although generally accepted standards for data structures do not exist for MRP II/ERP systems (Robertson and Perera 2001), the algorithms of material requirements and manufacturing resource planning require certain data to be present in the database, data that was absent in this case. It can be concluded that the usefulness of data analysis which can be obtained from digital sources, and which can be used for the simulation of the production process, seems to be much greater if the information systems for production resource planning are, at the least, MRP II class systems.

Because of the uniqueness and complexity of the production process in the company concerned, as well as the large frequency of change in the input parameters for simulation, the enterprise has made the decision to produce a dedicated simulator using a high level programming language. The data analysis carried out contributed to this decision and delivered the relevant information on the range of data existing in digital form for simulation purposes, as well as its degree of usefulness.

## SUMMARY

The presented methodology for the analysis of the data which exists in the database of PPC systems, and which is required for the purposes of simulation, is a proposal only. However, in the case discussed, the function of the methodology was successful.

But saying this, a more detailed practical verification based on a more intricate analysis, is required.

The authors of the paper must draw attention to the necessity for commercial enterprises, as well as academic institutions, to be involved in the search for solutions in the area of automatic data collection for simulation purposes. A mutual cooperation in this area between PPC (MRP II/ERP) systems, simulation software vendors and universities could offer many rewards.

## REFERENCES

- Gramacki J. and A. Darulewski. 2002. „Problematyka integracji danych w hurtowniach danych. Przykład rozwiązania opartego o produkty firmy ORACLE oraz inne produkty niekomercyjne”. 8<sup>th</sup> PLOUG Conference, Zakopane.
- Feldhaus U., 2004. „Geordnet simulieren”. In *Digital Engineering Magazin* 2/2004.
- Robertson N. and T. Perera. 2001. „Feasibility for automatic data collection”. In *Proceedings of the 2001 Winter Simulation Conference*, 984-990.
- Robertson N. and T. Perera. 2002. „Automated data collection for simulation?”. *Simulation Practice and Theory* 9 (2002) 349-364.
- Skrzypczak J. „Opracowanie modelu procesu produkcyjnego w fabryce autobusów na przykładzie wydziału Lakierni z uwzględnieniem modeli pojazdów i wariantowości pokrycia lakierniczego”. Master thesis at Poznan University of Technology. Weiss Z. (Supervision), Poznań.
- Weiss Z. and Lach M. 2004. „Aspekty rozszerzenia funkcjonalności systemów klasy ERP”. In *Komputerowo Zintegrowane Zarządzanie*, Knosala R. (Eds.). Warszawa, 653-659.
- Wesołowski K. „Symulacja procesu montażu autobusów”. Master thesis at Poznan University of Technology. Weiss Z. (Supervision), Poznań.

## AUTHOR BIOGRAPHIES



ZENOBIJA WEISS PhD., DSc. Eng. is a full Professor at Poznan University of Technology. She worked 5 years in Machine Tool Factory, later as consultant by Roller Bearing Factory and 2 years as researcher at Institute for Machine Tools and Manufacturing at TU Braunschweig, Germany. Prof. Weiss research interests concern Mechanical Technology, Computer Aided Design and Production Planning, Computer Integration and Supervision of Manufacturing Processes.

She managed many research projects for polish industry, Committee for Scientific Research and international projects. At present she is manager of the VIDA (Virtual Design and Automation) Center of Excellence and Head of Division of Industrial Information Systems at Poznan University of Technology. Her email address is:  
[zenobia.Weiss@put.poznan.pl](mailto:zenobia.Weiss@put.poznan.pl).



MARIA PIŁACIŃSKA graduated from computer science in 2001 at Poznan University of Technology. She is doctoral student in the Division of Industrial Information Systems at Poznan University of Technology. Her research interests

include databases systems, MRPII/ERP information systems and optimization of production schedule. Her email address is:  
[Maria.Pilacinska@put.poznan.pl](mailto:Maria.Pilacinska@put.poznan.pl).

# 3D MANUFACTURING SIMULATION – IMPROVING THE RETURN ON INVESTMENT

Mika Anttila  
Visual Components Oy  
Korppaanmäentie 17 CL 6, 00300 Helsinki, Finland  
E-mail: mika.anttila@visualcomponents.com

## KEYWORDS

3D manufacturing simulation, cost effective, component models

## ABSTRACT

If a picture says 1000 words, then a 3D simulation represents the equivalent of a novel. Correctly built 3D simulation models can accurately depict the appearance and functionality of manufacturing equipment. This includes the equipment motion, parts moving through the production line, timing studies, and material routing. In combination individual machinery models when connected together can look, behave, and function like the actual factory, giving you the capabilities to view a complete line from different angles, while it is running. This provides a much greater level of understanding than a static CAD model or a fixed view video. It enables you to actually see how a line is going to function before it has been installed .



Figures 1: Automotive Production Line Simulation

Selling capital equipment is tough even in the best of times. Communicating concepts and illustrating ROI has been a challenge for any sales team whose product is too large to fit in a brief case. With manufacturing equipment, the challenge is intensified by recent trends toward flexible manufacturing, increasing levels of complexity, and the continuous quest for improved efficiencies. This makes the challenge of justifying new

equipment purchases even more difficult without the proper tools.

Imagine being able to visualize your equipment operation on a notebook computer and being able to snap a line together and adjust parameters that configure the equipments physical size and behavior. Then, by pressing a button, see the effect of your changes and clearly demonstrate to others the results by simply e-mailing them the layouts. This capability can be realized by using the right 3D simulation tools.

## A SALES TOOL

Using simulation as a sales tool is nothing new. A few forward thinking companies embraced the promise 3D simulation offered as a sales tool in the early 90's. 3D simulation enabled these companies to demonstrate to the customer their current facility, how it could be improved with the next round of equipment purchase and, most importantly, the lasting vision of what their manufacturing facility could be capable of in the future. This enabled the equipment manufacturer to gain much more than a sale. They obtained a commitment to future expansion, which resulted in a steady flow of equipment sales.

Although there were a few success stories, these early adopters faced many challenges with 3D simulation including:

- High initial purchase costs
- Learning curves that can easily extend over 2 years for new operators
- Dependency on high priced consultants
- High level of complexity
- Difficulties in distributing the 3D simulation to a wide audience; this forced the use of static images or video as the main mechanism to share a simulation.

Due to these challenges, the use of 3D simulation as a sales tool was usually limited to companies selling very expensive equipment with the high margins necessary to fund the work required to develop, deploy, and maintain 3D simulation models of the company's equipment.

Recent advances in simulation technologies have drastically reduced, and in many cases, eliminated the

challenges faced by the early adopters. Thus, allowing 3D simulation to be deployed on a much larger scale by equipment manufactures. These advancements include:

1. Applying component modelling techniques to simulation models so that a plug-and-play works for connecting equipment together and drastically simplifies the task of line layouts.
2. Encapsulating the complexity within a component to facilitate re-use and streamline maintainability.
3. Simulation products that are layered for different user levels throughout the organization. The most basic of which can be easily used by anyone with basic computer skills.
4. The reduction in purchase price of the products.
5. Advancements in computer and software performance enabling interactive performance on laptop PC's.

The origins of factory simulation began in the 70's, with the products that were mainly statistical in nature. These products took numbers from timing studies, as input and output reports, enabling the industrial engineer to run experiments in order to determine which methods could improve the manufacturing process. Very quickly, these products evolved to include 2D charts and graphs. This technology was called "discrete event simulation" because of the way that manufacturing processes were approximated into single time taking events. This approximation made it possible for entire factories to be simulated on computers that were very slow by today's standards.

The 80's brought a new generation of simulation which is sometimes referred to as "physics based" to distinguish it from discrete event. These products were initially targeted at the Robotics market. They offered the ability to accurately simulate the motion of robots, check for collisions, and simulate the effects of inertia based on payload. The key advancement of these products is that they were completely 3D based. There were a number of success stories using these products for sales presentations and proposals due to the realistic nature of the 3D display.

Advancements of the 90's included a trend to provide 3D graphics capabilities in discrete event simulation products. Unfortunately, the simulation engine used by these products did not contain the accurate motion that was common in the physics based robotic simulation counterparts. While this technology proved effective in identifying bottlenecks and optimizing throughput, it did little to give the customer a clear picture of what they were purchasing and why.

The most recent advancements include the combination of physics based simulation and discrete event technology within the same product. This provides the ideal platform for the sales proposal market, since many of the key decision-makers in large-scale factory installations are non-technical in nature. These products provides a conduit for effective communication including:

1. Communicating new ideas and concepts to a customer thereby obtaining commitment earlier
2. Describing the competitive advantages of your equipment
3. Justifying value
4. Understanding what is being purchased and why

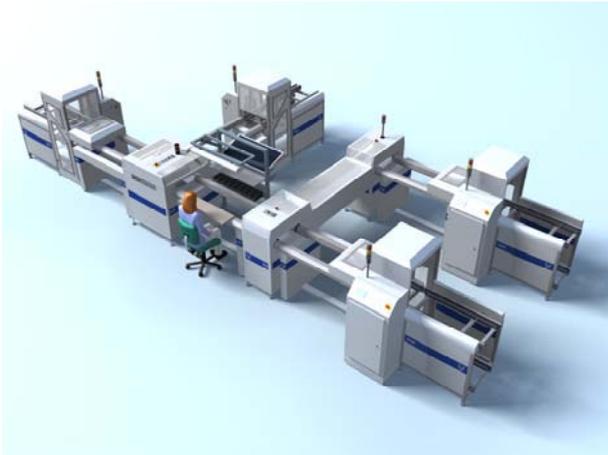
The use of 3D simulation does not stop once the order is received. It provides benefits that extend from the supplier, to the line integrator, and to the end user. By leveraging the model throughout the business process, a large cost savings can be realized, much more than if just the engineering department use simulation to complete layout and process design analysis.

### **COMPLETE PRODUCTION LIFECYCLE SUPPORT**

3D manufacturing simulation can support the complete production lifecycle from the early product designs to the re-configuration of production. 3D simulation can be used in all these steps as a communication tool to convey the idea of how a production works and how it performs to avoid costly mistakes as early as possible. Easier communication speeds up the decision making process creating better results faster.

In the early product development phase 3D simulation can be integrated into the design process in order to see the working product with performance data. This easier virtual prototyping supports the testing of a greater range of alternatives instead of settling into the first found solution found.

The simulation data created in the engineering processes can be reused in the sales and marketing phase when a customer is introduced to the new production capabilities with the modular reconfigurable simulation models. This is the place where new component-based simulation really shows its advantages. Earlier it has been impossible to re-use the engineering data for other purposes, but now production lines and systems can be configured to meet customer requirements in front of the customer showing realistic simulated processes with animated 3D graphics. The same models can be used to create high quality animations and pictures for all marketing and sales needs. Using visual simulation with performance data will give both the customer and the seller a clear understanding of what the customer will receive.



Figures 2: Reconfigurable Simulation Sales Model

Once a production system goes into the implementation phase, previously created simulation models can be used for verifying the line control logic to speed up the ramp up phase. This creates other possibilities to use the same simulation model for training purposes before the actual system is up and running. All this is achieved by using the simulation model as a virtual test bench that is connected to real control systems via a COM interface.

During production, the simulation model can be used for every day planning to validate and optimize production orders in advance. The simulation model may also serve as a 3D interface to the real system for monitoring and remote diagnostic purposes.

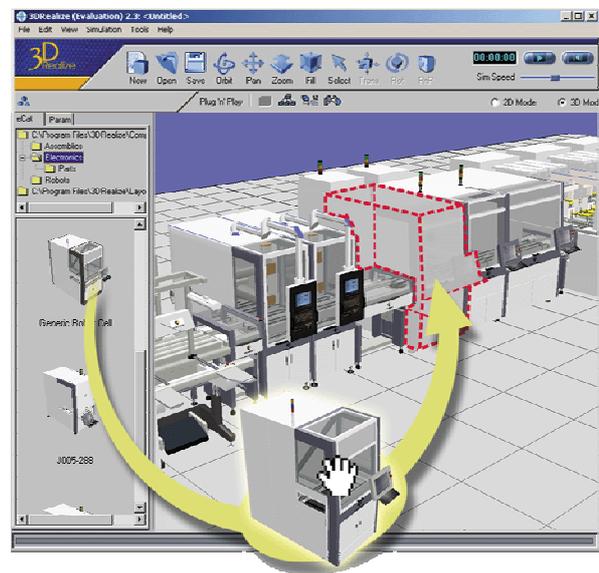
At the end of the production lifecycle, the simulation model can be reconfigured to meet the new production requirements.

### SAY GOOD-BYE TO EXPENSIVE, ONE-OFF SIMULATIONS

One-off simulation models built from scratch by a dedicated simulation engineer are expensive and used mostly for engineering purposes. The new component based simulation model approach re-uses simulation components allowing non-engineers to build simulation models quickly to meet many other needs within an organisation.

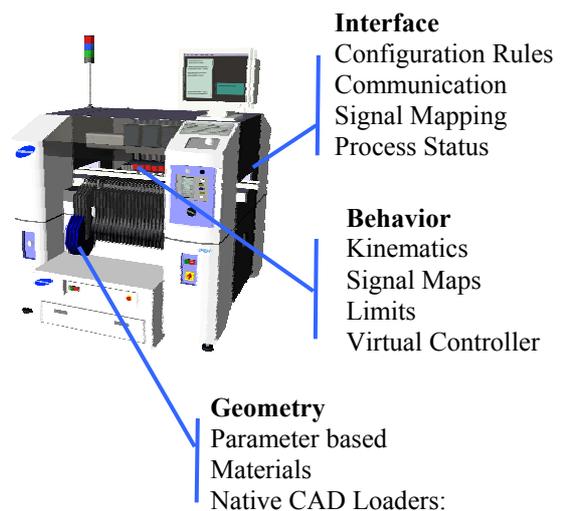
Figures 3: Easy Model Building with Plug&Play

Component based simulation is a similar phase shift, as movement to component-based software has been in the software industry. The same benefits which were discovered in the software industry can now be applied to simulation: components can be developed by a 3rd party, component complexity is encapsulated within the component, components can be added, deleted and exchanged easily, the component lifetime is easier to



manage and it is more future-proof. All this is available as long as the component interface remains same.

The component approach encapsulates machine or cell complexity into a modular simulation component, which interacts with other simulation components through an interface. In the component based approach interfaces have been abstracted to contain not only communication but also the material flow and connectivity.



Figures 4: Component Based Simulation  
Adding parameters to a simulation component also enhances the reusability and maintainability of the simulation models. Equipment manufacturers can represent a product family with one component module, and this is ideal for modular based production equipment. Sales engineers can change a few module parameters that automatically adjust the physical appearance and performance of an equipment model and demonstrate the effect to the customer interactively.

## ROBOTICS AND MATERIAL HANDLING ON THE ONE PLATFORM

The new behavior based component modeling approach now makes it possible to use the same simulation platform for robotics and material handling.

By combining different behaviours a component author can create a simulation module that has robotic mechanisms for assembly and material handling process for transferring parts to new processes. In the electronics PC board assembly lines most machines are a combination of conveyors and robot gantry heads, even a manual workstation is a combination of a complex assembly operator and conveyor belt. With behavior modeling it is easy to build such a component by adding required behaviors and connecting them together. As a basic requirement the simulation platform needs to provide built-in behaviors to author primitive type components such as robots, conveyors, and grippers etc.



Figures 5: Robotic and Material Handling Simulation

## PLC VALIDATION FOR FASTER PRODUCTION RAMP UP

Once simulation layouts are available they should support connection to the real operator interface or control system to validate the control logic. When a simulation model is connected to an industrial production controller most of the operator training and program debugging can be completed before the system is operational. This significantly reduces production ramp-up-time by removing most of the several weeklong debug periods traditionally required after the I/O field check. Real-time connectivity requires virtual time management to synchronize the communication

with external systems. Using an Open Process Control (OPC) connection, the control engineers can continue to use their familiar code development environment for their robot controller or PLC while testing the results on a “virtual factory” test bed.

## AUTOMATING THE PRE-STUDY NOT JUST PRODUCTION

Speeding up the pre-study phase for production with intelligent component based simulation tools allows engineers to keep up with the demands of reducing product cycles and increasing model variants running on the same lines.

As production lines become more automated and ‘digital’ in nature the opportunities now exist to ‘plug and play’ production line components such as conveyors, robots, tooling, and model variants together to evaluate a host of production alternatives very quickly. Using component based simulation makes it possible to check the layout, components, and their interoperability in one environment. Checking out the impact of a configuration or process change allows production management to quickly and directly trial and communicate ‘what-if’ scenarios/layouts without the need to translate the concept via a CAD operator or simulation expert. Earlier it could have taken days or weeks for a dedicated and experienced simulation engineer to create a simulation model for evaluation purposes. Now any engineer with a little training can generate the same models in hours.

## ABOUT VISUAL COMPONENTS

Visual Components is a world leading 3D robotics and manufacturing simulation software provider offering machine builders, system integrators and companies using complex turnkey manufacturing solutions a simple, quick and highly cost effective way to build and simulate their total process solutions. Finnish based Visual Components has global OEM partners and an extensive reseller network. For more information please refer to [www.visualcomponents.com](http://www.visualcomponents.com)



## AUTHOR BIOGRAPHIES

**MIKA J. ANTILA** is a simulation application specialist at Visual Components, with 16 years experience in visualisation, simulation and simulation software development.

# DATA MINING APPLIED TO AGENT BASED SIMULATION

Marco Remondino  
Department of Computer Science  
University of Turin  
C.so Svizzera 185  
10149 Turin, Italy  
E-mail: remond@di.unito.it

Gianluca Correndo  
Department of Computer Science  
University of Turin  
C.so Svizzera 185  
10149 Turin, Italy  
E-mail: correndo@di.unito.it

## KEYWORDS

Data Mining, Agent Based Simulation, validation, emergence, artificial intelligence

## ABSTRACT

Agent Based Modeling is the most interesting and advanced approach for simulating a complex system: in a social context, the single parts and the whole are often very hard to describe in detail. Besides, there are agent based formalisms which allow to study the emergency of social behavior with the creation and study of models, known as artificial societies. Thanks to the ever increasing computational power, it's been possible to use such models to create software, based on intelligent agents, which aggregate behavior is complex and difficult to predict, and can be used in open and distributed systems.

Data mining is born in the last decades in order to help users in finding useful knowledge from the otherwise overwhelming amount of data available nowadays from the web and the data collected every day by companies. Data Mining techniques can therefore be the keystone to reveal non-trivial knowledge expressed by the initial assumption used to build the micro-level of the model and the structure of the society of agents that emerged from the simulation.

## INTRODUCTION

Nowadays the simulation is one of the best paradigms for modeling the behavior of complex systems even though it has some leaks. Above all, the simulation model is only a rough approximation of the real system to study; each approximation produced will not cover the whole set of details we can actually study looking at the real system. The gap between model and reality is well known in fields like Computer Science and Mathematics but the situation is far from being desperate. The gap can be intentional since the realm of interest can be a small piece of the whole sensible world. Moreover the ability to create artificial worlds whose relations and states can be arbitrarily changed allows us to explore the full possibility of the real system. The fact to simulate a system under unnatural conditions can help us to study scenarios of unimaginable flexibility.

The other side of the coin is that the procedure of modeling introduces a bias that it is difficult to detect. How reliable is a model? How to proceed in the model proposal? These are difficult questions to answer when there are no strong and formal fundamentals in model proposal.

Statistical techniques usually try to overcome such bias using distribution hypothesis and strong mathematical foundations for the procedures used during data analysis.

In the present paper the authors try to propose a cross fertilization between Agent Based Simulation and Statistical Learning techniques (or more specifically Data Mining techniques) in order to handle and possibly overcome the limitations of both.

## AN INTRODUCTION TO DATA MINING

Data Mining is the key element of the Knowledge Discovery in Databases (KDD) task. KDD is defined as *“the process of identifying valid, novel, potentially useful and ultimately understandable patterns in data”*. We could finally add that such task involves usually great amount of data, usually stored in analysis oriented data stores called Data Marts.

Data Mining is not a field in itself; it is more a collection of methods of data analysis coming from different fields of computer science, artificial intelligence and statistics. Just statistics supplies mathematical concreteness to many of the data mining methods.

Data mining was born in the latest decades in order to help users in finding useful knowledge from the otherwise overwhelming amount of data available nowadays, both coming from the web and the data collected every day by companies.

The kind of knowledge the users can extract from the raw data is heterogeneous and most depends on the nature of the data available. In fact the nature of the data and the kind of task guide the process of data analysis itself, that is more the production of an artificial process guided by the experiences rather than the result of an automatic process.

The types of tasks Data Mining could accomplish can be roughly divided in two categories: predictive tasks and descriptive tasks. The first type of tasks try to discover a model that drives the behavior of some variables in a system in order to be able to predict such values in zones not covered by the examples. The

second type of task tries to find some categorizations of the data producing a shrunk descriptor for wider segments of data.

### **Predictive Data Mining**

One of the predictive tasks of Data Mining is the task of finding some form of classifications of the items contained in the data mart from a set of raw data. When there is a finite set of classes that describe the domain of the data, the classification can be carried on by some *if-then* rules that help users to classify a new item in one of such predefined classes. Such classification process is based on the values of some characteristics of the item itself and can be deterministic (e.g. there is no doubt about the belonging of the item to the given class) or heuristic (e.g. the association of the item to one or more classes is given with a degree of certainty).

The association model so far extracted can have the form of a decision tree, instead of a set of if-then rules, but the purposes of the model retrieved remains the same. When the classification domain is not finite (e.g. when the variable interested by the prediction process is a real number) the operation is called *regression*. The regression task helps the user to model an analytic function that describes the set of data submitted to the task and that can predict new, not submitted, values.

### **Descriptive Data Mining**

In descriptive Data Mining the task is to discover interesting regularities in the data, to uncover patterns and find interesting subgroups in the bulk of data.

Such kind of Data Mining produce a categorization of the initial amount of data uncovering patterns that were not evident before the execution of the task. Expert of the domain must then interpret the patterns so far uncovered in order to explain them.

A typical product of this kind of task is the discovery of association rules that find untitled relationships between features' values looking at the examples proposed as training.

Such association rules can be used as classifiers to find some subgroups dividing the population in relevant clusters. The division in clusters reflects some important division present in the data that could be crucial in order to reason using a small number of stereotypes instead of a huge number of single items.

Another important task associated to Data Mining is the use of advanced techniques of visualization. In fact, since data analysts and domain specialists do most of the work of discovery, it is very important to find good visual metaphors to give users right intuitions to guide the analysis.

Naturally such metaphors are only useful to guide the intuition, in order to provide mathematical soundness the Data Mining is supported by statistical methods such as probabilities laws for the items values' prediction, Bayesian theorems for defining some sort of causality and so on. The techniques of Data Mining, having their foundations in statistic, require a large number of items to build satisfying results. When only a

small number of examples are available, techniques of Machine Learning, coming from AI and inductive logic fields, are suggested. Such techniques find their fundamentals in symbolic reasoning and non-classical logics and do not require statistical tools for soundness checking.

### **AGENTS FOR SOCIAL SIMULATION**

The concept of software agent originates in the early fifties with J. McCarthy, while the term has been coined by O.G. Selfridge some years later, when both of them were working at the Massachusetts Institute of Technology. Their original project was to build a system which, given a goal, could be able to accomplish it, looking for human help in case of lack of necessary information. In practice, an agent was considered a software robot that lives and acts in a virtual world. In (Wooldridge and Jennings 1995): "... a hardware or (more usually) software-based computer system that enjoys the following properties:

- *autonomy*: agents operate without the direct intervention of humans or others, and have some kind of control over their actions and internal state;
- *social ability*: agents interact with other agents (and possibly humans) via some kind of agent-communication language;
- *reactivity*: agents perceive their environment, (which may be the physical world, a user via a graphical user interface, a collection of other agents, the internet, or perhaps all of these combined), and respond in a timely fashion to changes that occur in it;
- *pro-activeness*: agents do not simply act in response to their environment, they are able to exhibit goal-directed behaviour by taking the initiative." The Wooldridge and Jennings definition, in addition to spelling out autonomy, sensing and acting, allows for a broad, but finite, range of environments. They further add a communications requirement.

Franklin and Graesser (1997) also try to find the typical features of agency, deriving them from the word itself: an "agent" is 1) one who acts, or who can act, and 2) one who acts in place of another with his permission. Since "one who acts in place of " acts, the second usage requires the first. Humans act, as do most other animals. Also, some autonomous mobile robots act, for example Brooks' Herbert (Brooks 1990; Franklin 1995). All of these are real world agents. Software agents "live" in computer operating systems, databases, networks, MUDs, etc.

Finally, artificial life agents "live" in artificial environments on a computer screen or in its memory (Langton 1989, Franklin 1995).

Each is situated in, and is a part on some environment. Each senses its environment and act autonomously upon it. No other entity is required to feed it input, or to interpret and use its output. Each acts in pursuit of it's own agenda, whether satisfying evolved drives as in humans and animals, or pursuing goals designed in by

some other agent, as in software agents. (Artificial life agents may be of either variety.) Each acts so that its current actions may effect its later sensing, that is its actions effect its environment. Finally, each acts continually over some period of time. A software agent, once invoked, typically runs until it decides not to. An artificial life agent often runs until it's eaten or otherwise dies. Of course, some human can pull the plug, but not always. Mobile agents on the Internet may be beyond calling back by the user.

These requirements constitute for sure the essence of being an agent, hence the definition by Franklin and Graesser (1997):

*An autonomous agent is a system situated within and a part of an environment that senses that environment and acts on it, over time, in pursuit of its own agenda and so as to effect what it senses in the future.*

And the very general, yet comprehensive one by Jennings (1996):

*...the term is usually applied to describe self-contained programs which can control their own actions based on their perceptions of their operating environment.*

Agents themselves have traditionally been categorized into one of the following types (Woolridge and Jennings, 1995):

- Reactive
- Collaborative/Deliberative
- Hybrid

When designing any agent-based system, it is important to determine how sophisticated the agents' reasoning will be. Reactive agents simply retrieve pre-set behaviors similar to reflexes without maintaining any internal state. On the other hand, deliberative agents behave more like they are thinking, by searching through a space of behaviors, maintaining internal state, and predicting the effects of actions. Although the line between reactive and deliberative agents can be somewhat blurry, an agent with no internal state is certainly reactive, and one that bases its actions on the predicted actions of other agents is deliberative.

In Mataric (1995) we read that reactive agents maintain no internal model of how to predict future states of the world. They choose actions by using the current world state as an index into a table of actions, where the indexing function's purpose is to map known situations to appropriate actions. These types of agents are sufficient for limited environments where every possible situation can be mapped to an action or set of actions.

The purely reactive agent's major drawback is its lack of adaptability. This type of agent cannot generate an appropriate plan if the current world state was not considered a priori. In domains that cannot be completely mapped, using reactive agents can be too restrictive.

Different from reactive agents are the deliberative ones. The key component of a deliberative agent is a central reasoning system (Ginsberg, 1989) that constitutes the intelligence of the agent. Deliberative agents generate plans to accomplish their goals. A world model may be used in a deliberative agent, increasing the agent's ability to generate a plan that is successful in achieving its goals even in unforeseen situations. This ability to adapt is desirable in a dynamic environment.

The main problem with a purely deliberative agent when dealing with real-time systems is reaction time. For simple, well known situations, reasoning may not be required at all. In some real-time domains, such as robotic soccer, minimizing the latency between changes in world state and reactions is important.

Hybrid agents, when designed correctly, use both approaches to get the best properties of each (Bensaid and Mathieu, 1997). Specifically, hybrid agents aim to have the quick response time of reactive agents for well known situations, yet also have the ability to generate new plans for unforeseen situations.

### **Multi Agent Systems (MAS)**

A multi agent system can be thought of as a group of interacting agents working together to achieve a set of goals. To maximize the efficiency of the system, each agent must be able to reason about other agents' actions in addition to its own. A dynamic and unpredictable environment creates a need for an agent to employ flexible strategies. The more flexible the strategies however, the more difficult it becomes to predict what the other agents are going to do. For this reason, coordination mechanisms have been developed to help the agents interact when performing complex actions requiring teamwork. These mechanisms must ensure that the plans of individual agents do not conflict, while guiding the agents in pursuit of the goals of the system.

### **AGENT BASED SIMULATION**

The most diffused simulation paradigms are: Discrete Event (DE) Simulation, System Dynamics (SD) and Agent Based (AB) Simulation.

The term DE simulation applies to the modeling approach based on the concepts of entities, resources and block charts describing entity flow and resource sharing. DE simulation is usually applied to process modeling, hence the definition of "process simulation", which is a sub-set of the DE one.

According to Jay W. Forrester in the 1950s, SD is "*the study of information-feedback characteristics of industrial activity to show how organizational structure, amplification (in policies), and time delays (in decisions and actions) interact to influence the success of the enterprise*". SD heavily relies upon systems of differential equations, which best represents the feedback loops typical of this approach.

In (Ostrom 1988), agent based simulation is described as a third way to represent social models, being a

powerful alternative to other two symbol systems: the verbal argumentation and the mathematical one. The former, which uses natural language, is a non-computable way of modelling though a highly descriptive one; in the latter, while everything can be done with equations, the complexity of differential systems rises exponentially as the complexity of behaviour grows, so that describing complex individual behaviour with equations often becomes an intractable task. Simulation has some advantages over the other two: it can easily be run on a computer, through a program or a particular tool; besides it has a highly descriptive power, since it is usually built using a high level computer language, and, with few efforts, can even represent non-linear relationships, which are tough problems for the mathematical approach. According to (Gilbert, Terna 2000):

*“The logic of developing models using computer simulation is not very different from the logic used for the more familiar statistical models. In either case, there is some phenomenon that the researchers want to understand better, that is the target, and so a model is built, through a theoretically motivated process of abstraction. The model can be a set of mathematical equations, a statistical equation, such as a regression equation, or a computer program. The behaviour of the model is then observed, and compared with observations of the real world; this is used as evidence in favour of the validity of the model or its rejection”*

In Remondino (2003) we read that computer programs can be used to model either quantitative theories or qualitative ones; simulation has been successfully applied to many fields, and in particular to social sciences, where it allows to verify theories and create virtual societies. In order to simulate the described problem, multi-agent technique is used. Agent Based Modelling is the most interesting and advanced approach for simulating a complex system: in a social context, the single parts and the whole are often very hard to describe in detail. Besides, there are agent based formalisms which allow to study the emergency of social behaviour with the creation and study of models, known as artificial societies. Thanks to the ever increasing computational power, it's been possible to use such models to create software, based on intelligent agents, which aggregate behaviour is complex and difficult to predict, and can be used in open and distributed systems. The concept of Multi Agent System for social simulations is thus introduced: the single agents have a very simple structure. Only few details and actions are described for the entities: the behaviour of the whole system is a consequence of those of the single agents, but it's not necessarily the sum of them. This can bring to unpredictable results, when the simulated system is studied.

In an AB model, there is not a place where the global system behavior (dynamics) would be defined. Instead, the modeler defines behavior at individual level, and the global behavior emerges as a result of many (tens, hundreds, thousands, millions) individuals, each

following its own behavior rules, living together in some environment and communicating with each other and with the environment. That is why AB modeling is also called bottom-up modeling.

The agent-based view takes a different approach to modeling. Instead of creating a simple mathematical model, the underlying model is based on a system comprised of various interacting agents. Therefore, its structure and behavior have potential to resemble the actual economic theory and reality better than simple mathematical models. Especially, when the underlying real relationships are complex.

In (Bonabeau, 2002), we read that AB paradigm can be used successfully to model different situations, like flows, markets, organizations, social diffusion of phenomena.

## **DATA MINING IN AGENT BASED SIMULATION TASKS**

While in the process simulation the focus is on the function description of the single parts that are modeled in detail, in agent based simulation the most important facet is the interaction among entities. In fact it is such interaction that produce a variety of behavior that was not explicitly described in the model of the single parts. In agent based simulation there are therefore two main levels that use distinct languages with distinct purposes. A micro-level used to describe a simple local behavior and a macro-level whose effects derive in part from the micro-level and in part from the interaction of more elements. Such emergent behaviors could be revealed by non-explicit patterns in the simulation data and a following phase to the simulation can be needed in order to reveal the model that subtend the data production. Data Mining techniques can therefore be the keystone to reveal non-trivial knowledge expressed by the initial assumption used to build the micro-level of the model and the structure of the society of agents that emerged from the simulation.

Data Mining, and Machine Learning in general can be used in a number of ways in agent-based simulation, we can classify these contributions in two main tasks:

- *Endogenous modeling.* Where Machine Learning and Data Mining techniques can be used to provide the single agent a sort of intelligent behavior that analyze the data of past executions of the simulation learning from experience and tuning some initial parameters of the simulation in order to reach some local maximum (Remondino, 2003).
- *Exogenous modeling.* Where the final results of a simulation are analyzed using Data Mining techniques in order to reveal interesting patterns in data that could help to better model the behavior of the overall systems. Note that the system's behavior is usually more that the sum of the parts and it is not described in the first phase of the simulation task. Data Mining could be used to build a model supported by statistical evidence that

could validate or refuse some initial hypothesis on the system.

### **Endogenous Modeling**

A lot of models used in agent-based simulation tries to capture the emergent unpredictable behavior of rational agents when they interact with a population of peers. The machine learning algorithms allows an agent to learn from its past history in a human similar way, that is to say, by induction. we can choose to create agents with the ability to compute rules and strategies, and evolve according to the environment in which they act; in order to model them, we can use some methods derived from the studies on artificial intelligence, such as artificial neural networks and evolutionary algorithms. While the former is a collection of mathematical functions, trying to emulate nervous systems in the human brain in order to create learning through experience, the latter derives from observations of biological evolution. Genetic Algorithms derive directly from Darwin's theory of evolution, often explained as "survival of the fittest": individuals are modelled as strings of binary digits and are the encode for the solution to some problem. The first generation of individuals is often created randomly, and then some fitness rules are given (i.e. better solutions for a particular problem), in order to select the fittest entities. The selected ones will survive, while the others will be killed; during the next step, a crossover between some of the fittest entities occurs, thus creating new individuals, directly derived from the best ones of the previous generation. Again, the fitness check is operated, thus selecting the ones that give better solutions to the given problem, and so on. In order to insert a random variable in the genetic paradigm, that's something crucial in the real world, a probability of mutation is given; this means that from one generation to the next one, one or more bits of some strings can change randomly. This creates totally new individuals, thus not leaving us only with the direct derivatives of the very first generation. Genetic Algorithms have proven to be effective problem solvers, especially for multi-parameter function optimization, when a near optimum result is enough and the real optimum is not needed. This suggests that this kind of methodology is particularly suitable for problems which are too complex, dynamic or noisy to be treated with the analytical approach; on the contrary, it's not advisable to use Genetic Algorithms when the result to be found is the exact optimum of a function. The risk would be a convergence to some results due to the similarity of most the individuals, that would produce new ones that are identical to the older ones; this can be avoided with a proper mutation, that introduces in the entities something new, not directly derived from the crossover and fitness process. In this way, the convergence should mean that in the part of the solution space we are exploring there are no better strategies than the found one. It's crucial to choose the basic parameters, such as

crossover rate and mutation probability, in order to achieve and keep track of optimal results and, at the same time, explore a wide range of possible solutions. Classifier Systems derive directly from Genetic Algorithms, in the sense that they use strings of characters to encode rules for conditions and consequent actions to be performed. The system has a collection of agents, called classifiers, that through training evolve to work together and solve difficult, open-ended problems. They were introduced in (Holland 1976) and successfully applied, with some variations from the initial specifics, to many different situations. The goal is to map if-then rules to binary strings, and then use techniques derived from the studies about Genetic Algorithms to evolve them. Depending on the results obtained by performing the action corresponding to a given rule, this receives a reward that can increase its fitness. In this way, the rules which are not applicable to the context or not useful (i.e. produce bad results) tend to loose fitness and are eventually discarded, while the good ones live and merge, producing new sets of rules. In (Kim, 1993) we find the concept of Organizational-learning oriented Classifier System, extended to multi-agent environments with introducing the concepts of organizational learning. According to (Takadama 1999), in such environments agents should cooperatively learn each other and solve a given problem. The system solves a given problem with multi-agents' organizational learning, where the problem cannot be solved simply by the sum of individual learning of each agent.

### **Exogenous Modeling**

In particular, the exogenous modeling can be an important task in agent-based simulation since it provides safe techniques to analyze the results of this kinds of simulation paradigm. In fact, one of the most debated issues in agent based simulation community is the absence of a safe technique for validate the results of the simulations. This kind of statistical analysis of the results of the simulation could provide a real added value to this kind of representation of social models. In fact, in modeling social systems, the first step is to create a metaphor of the real system. Such models of the reality suffer, as we said in the introduction, of some initial hypothesis that must test when the first results came up. The usual validation is based upon the matching of the simulation values; if the model predicts, to some extent, the values observed in reality then this is taken as a proof of validity of the model itself (Gilbert, Terna 1999). The goodness criterions follow usually statistical theories and make reference to the knowledge of hypothesis testing, where a distribution of values is compared to a reference distribution in order to come up with a fitness number.

Using Data Mining we can use statistical foundations in order to deduce from the values of the simulation a model that well describe such values. Such models provided by statistical analysis are relative to the whole system; they try to describe, with simple and

deterministic models, how the single entities cooperate in order to produce the observed behavior.

There are many Data Mining tools that can be used in order to help the analysts to extract valuable knowledge about the reality whose drives the modeling phase or about the model itself. In the following we will provide a short overview of those whose are more interesting in our point of view, but the discussion is far from being closed. This is just a hint in order to stimulate the discussion.

### Analysis of variance

The analysis of variance is one method used in statistical analysis to discover unsaid relationships between variables of a system. In few words, variables are related if the distribution of their values systematically corresponds. For example, in a population, the height is related to weight because typically tall individuals are heavier than short ones.

Analysis of variance can be a good starting point in model proposal. In fact, looking at the system to be modelled, the user can be prompted to recognize some relationships existents between internal variables trying to model such relationships accordingly.

### Multiple regression

In multiple regression, as well as in the analysis of variance, the goal is to find relationships between variables of a system. The difference in multiple regression, and in regression in general, is that such method tries to estimate such relationship rebuilding an equation that describe the behaviour of one or more dependant variables in function of one or more independent variables. There is more than one method in order to operate such regression whose main distinction can be seen from linear methods (where the equation obtained is linear in the input parameters) and non-linear methods (where the equation can be a polynomial or other functions).

Pushing further the concept of preliminary analysis of the system to simulate, we can use multiple regression in order to:

- guiding the modeling phase proposing some algorithm that code the so far discovered behaviour
- make the tuning of some initial parameters of the simulation before the simulation starts
- use the multiple regression above the real system and the modelled one in order to provide a degree of adherence of the model to the real world

### Cluster analysis

In cluster analysis the goal is to retrieve some collections of individuals whose description (or behaviour) is alike. In clustering analysis, the users can define a distance measure based on the properties of single agents. Moreover he can recognizes if, within the system, are present well-defined set of individuals that are similar, based of the given distance measure.

This is useful in order to decrease the number of element to describe within the system. In fact, instead of focusing over the single agent behaviour in an object-oriented way, the user could look at the system as a set of clusters whose elements are in some way equivalent. Recognizing the fact that the description of single elements can be summarized by the description of few clusters can help to decrease the heterogeneity of the system.

### Association rules

In this method the aim is to find regular patterns that describe categorical data and express such patterns using “if then” rules that recall a causal semantics. The rationale used to extract these rules is quite simple, the hard part is to apply it to huge amount of data. The method records the frequencies of certain patterns within a load of observations. For example, if every time the variable “a” has value “1” then the variable “b” has the value “0” we can deduce that the rule “ $a=1 \rightarrow b=0$ ” holds. We are not able to say why it is like that, but the available observations give us a certain degree of certainty.

The causal semantics associated to the results and its algorithmic nature provide us with a natural instrument to explore the hidden model followed by the system

### Iterative process in modeling phase

By using the above described methods, and many others not mentioned here, we can define a modeling and model revision process. Such process starts from the first task of model building (Model Building task in Figure 1) where a first proposal of model is done and will be tested after various runs. As we introduced in the first part of the paper, such task suffers from a set of initial hypothesis and it produces a first proposal of model used in the simulations. In this very first phase Data Mining (DM Analysis in Figure 1) can be used in order to make safe hypothesis over the real behavior of the system (or at least for that portion of the behavior that is observable, simulation is a good way to discover new scenarios that are not observed before).

When the simulation has produced a good amount of observations to work with (Simulation task in Figure 1) a new phase of Data Mining analysis can be used to make hypothesis above the model produced (DM Analysis task in Figure 1). Such results could validate or refuse the initial hypothesis about the real world and could guide a revision process in order to refine our knowledge about the overall system (Model Revision task in Figure 1).

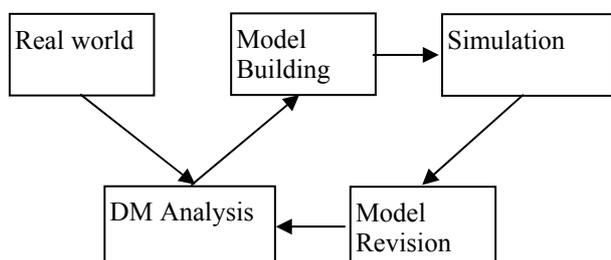


Figure 1: DM revision process applied to AB Simulation

Such iterative process could produce finer and finer model hypothesis until a desired convergence is found. Moreover, during the revision process the user could have a sound statistical theory as a guidance that provides him/her with a measure of the fitness of the model.

## CONCLUSIONS AND FUTURE DIRECTIONS

In our work we explored the ways in which Data Mining techniques could be successfully applied to Agent Based Modeling and Simulation, in order to exploit hidden relations and emergent behavior. We found that Data Mining, and Machine Learning in general can be used in a number of ways in agent-based simulation, we can classify these contributions in two main tasks: Endogenous modeling, where Machine Learning can be used to provide the single agent a sort of intelligent behavior and Exogenous modeling, Where the final results of a simulation are analyzed using Data Mining techniques in order to reveal interesting patterns in data that could help to better model the behavior of the overall systems. We provide an overview of the tools that we think could be useful to accomplish this task: Analysis of variance, Multiple regression, Cluster analysis, Association rules. By using the above described methods, and many others not described here, we can define a modeling and model revision process. In future works we plan to apply the techniques described here to simple agent based models and demonstrate they can be useful for model validation and hidden patterns analysis.

## REFERENCES

- Bensaid N., & Mathieu P. (1997), A hybrid architecture for hierarchical agents
- Bonabeau, E. 2002. "Agent-based modeling: Methods and techniques for simulating human systems", PNAS 99 Suppl. 3: 7280-7287.
- Brooks, R. A. (1990). Elephants Don't Play Chess, *Designing Autonomous Agents*, Pattie Maes, ed. Cambridge, MA: MIT Press
- Franklin, S. (1995). Artificial Minds, Cambridge, MA: MIT Press
- Franklin, S., & Graesser, A. (1997). Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents, *Proceedings of the Agent Theories, Architectures, and Languages Workshop, Berlin* (pp. 193-206). Springer Verlag.
- Gilbert, N. and Terna, P. 2000. "How to build and use agent-based models in social science", *Mind & Society* 1, 57-72
- Ginsberg M. (1989). Universal planning: An (almost) universally bad idea. *AI Magazine* 10(4) pp. 40-44
- Langton, C. (1989). Artificial Life, Redwood City, CA: Addison-Wesley
- Mataric, M. (1995). Issues and approaches in the design of collective autonomous agents. *Robotics and Autonomous Systems* 16 (pp.321-331)
- Ostrom T. (1988). Computer simulation: The third symbol system. *Journal of Experimental Social Psychology*, 24 (pp. 381-392)
- Remondino, M. (2003). Agent Based Process Simulation and Metaphors Based Approach for Enterprise and Social Modeling, *ABS 4 Proceedings* (pp.93-97). SCS Europ. Publish. House – ISBN 3-936-150-25-7
- Woolridge, M., & Jennings, N. (1995). Intelligent agents: Theory and practice. *Knowledge Engineering Review* 10(2) pp.115-152

## AUTHORS BIOGRAPHY



**MARCO REMONDINO** was born in Asti, Italy, and studied Economics at the University of Turin, where he obtained his Master Degree in March, 2001 with 110/110 cum Laude et Menzione and a Thesis in Economical Dynamics. In the same year, he started attending a PhD at the Computer Science Department at the University of Turin, which he got in January 2005. His main research interests are Computer Simulation applied to Social Sciences, Enterprise Modeling, Agent Based Simulation, Multi Agent Systems and BDI agents. He has been part of the European team which defined a Unified Language for Enterprise Modeling (UEML). He is also participating to a University project for creating a cluster of computers, to be used for Social Simulation.



**GIANLUCA CORRENDO** was born the 12th of August 1974, he started his research in 1999 with a laurea thesis in model based diagnosis in a joint project sponsored by ASI (Italian Space Agency). Gianluca Correndo got his PhD in January 2005 with a thesis in ontologies in medical applications. His research activities are model based diagnosis, agent technologies, medical informatics and data mining. He has started a post doc in data mining field at the TOTAL CeRT (Centre of Research and Technique) where has the role of data mining engineer.

# ANT COLONY ROUTE OPTIMIZATION FOR MUNICIPAL SERVICES

Nikolaos V. Karadimas, Georgios Kouzas, Ioannis Anagnostopoulos, Vassili Loumos and Elefterios Kayafas  
School of Electrical & Computer Engineering  
Department of Communications, Electronics & Information Engineering  
National Technical University of Athens

15780, Athens, Greece

E-mail: [nkaradim@central.ntua.gr](mailto:nkaradim@central.ntua.gr), [gkouzas@ece.ntua.gr](mailto:gkouzas@ece.ntua.gr), [janag@telecom.ntua.gr](mailto:janag@telecom.ntua.gr), [loumos@cs.ntua.gr](mailto:loumos@cs.ntua.gr),  
[kayafas@cs.ntua.gr](mailto:kayafas@cs.ntua.gr)

## KEYWORDS

Ant Colony Optimization (ACO) Algorithm, Quality of Service, Solid Waste, Cost Optimization, Simulation.

## ABSTRACT

In the present paper the Ant Colony Optimization (ACO) Algorithm is introduced for best routing identification applied in urban solid waste collection.

The proposed solid waste management system is based on a geo-referenced Spatial Database supported by a Geographic Information System (GIS). The GIS takes account of all the required parameters for solid waste collection.

These parameters involve static and dynamic data, such as positions of trash-cans, road network, related traffic and population density, In addition, time schedule of trash-collection workers and track capacities and technical characteristics are considered.

ACO spatio-temporal statistical analysis model is used to estimate interrelations between dynamic factors, like network traffic changes in residential and commercial areas in a 24 hour schedule, and to produce optimized solutions.

The user, in the proposed system, is able to define or modify all required dynamic factors for the creation of an initial scenario. By modifying these particular parameters, alternative scenarios can be generated leading to the several solutions.

The Optimal solution is identified by a cost function that takes into account various parameters, for instance labor and equipment costs as well as social implications.

## INTRODUCTION

The last years, much effort has been made in Urban Solid Waste Collection and Transport Management. The problem can be classified as either a Traveling Salesman or a Vehicle Routing Problem. For this particular problem, several solutions and models have been proposed.(Pham and Karaboka, 2000; Ducatelle and Levine, 2001; Bianchi et al. 2002; Chen and Smith, 1996; Glover and Laguna, 1992).

The complexity of the problem is high due to many alternatives that have to be considered. The number of possible solutions is considerably high. Fortunately, many algorithms have been developed and discussed in order to find a shorter path to the optimized solution (Tarasewich and McMullen , 2002).

The most popular algorithms used today in similar cases include the Genetic Algorithm, the Simulated Annealing, the Tabu Search, the Ant Colony Optimization (ACO and others. In the present work, some of these algorithms have been tested and discussed to a certain depth.

Genetic algorithms (Pham and Karaboka, 2000; Chen and Smith, 1996; Glover et al. 1992) use biological methods such as reproduction, crossover, and mutation to quickly search for solutions to complex problems. Genetic algorithm begins with a random set of possible solutions. In each step, a fixed number of the better current solutions are saved and they are used to the next step to generate new solutions using genetic operators. Crossover and mutation are the most important genetic operations are used. In the crossover function parts of two random solutions are chosen and they are exchanged between two solutions. As a result two new child solutions are generated. The mutation function alters parts of a current solution generating a new one. The mutation function is included to keep from becoming trapped at a local optimum. These procedures are repeated for a predefined number of iteration until an acceptable solution is generated.

The Simulated Annealing was inspired from the behavior of solids in temperature (Pham and Karaboka, 2000); a solid is heated to a high temperature and then slowly cooled, until the desired properties of the solid are obtained. When the Simulated Annealing begins, an initial solution is generated as the first solution. Then the "temperature" is symmetrically reduced and neighboring solutions are generated. If one of the neighboring solutions is better than the current solution, then it becomes the current one. If not, these solutions remain as candidate solutions and one of them can become the final one, if it satisfies some predefined criteria. The acceptance of inferior solutions allows the search, of many different locations, so the probability of falling in

a local optimal solution decreases dramatically. This procedure is repeated, until some stopping criteria are met.

Tabu (or taboo) search as described by (Glover, 1986) is a meta-heuristic. The basic gist of tabu search is to iteratively try to find solutions to the problem, but to keep a short list of previously found solutions and to avoid 're-finding' those solutions in subsequent iterations (Battini and Tecchioli, 1994). Basically, if you try a solution, it becomes tabu in future tries.

The Ant Colony Optimization algorithm (Dorigo & Maniezzo, 1996), was inspired through the observation of swarm colonies and specifically ants. Ants are social insects and their behaviour is focused to the colony survival rather the survival at the individual. Specially, the way ants find their food is noteworthy. Although ants are almost blind, they build chemical trails, using a chemical substance called pheromone. The trails are used by ants to find the way to the food or back to their colony. The ACO simulates this specific ants' characteristic, to find optimum solutions in computational problems, such as the Traveling Salesman Problem. As this context is mainly focused on the ACO algorithm and its testing to the solid waste collection problem, the ACO is analytically described in the next section.

As it was mentioned above, the ACO algorithm is tested to the problem of collection and transport of solid waste from any loading spot in an area under study to the transshipment or the disposal sites. Of course, our research only covers how the algorithm applies to the routes included in the examined area.

Therefore, in this context, a framework (schema) for the design and implementation of a solution for solid waste collection and transport is proposed. According to this schema, the ACO algorithm, an innovative algorithm in the specific research area, is introduced and implemented, for monitoring, simulation, testing, and cost optimization of alternative scenarios of a solid waste management system.

This schema is described in the sections that follow. More specifically, section 2 introduces and describes the ACO algorithm. In Section 3, the waste management problem in the selected case study area is introduced. Section 4 describes the methodology and the proposed system and how it is applied in the current situation using the ACO algorithm and at the same time outlining some of the proposed variants. Finally, Section 5, illustrates the simulation results achieved comparing them to present solutions and other algorithms. Conclusions and future developments are also discussed in this section.

## ANT COLONY OPTIMIZATION ALGORITHM

### Real Ants

The basic idea of ACO algorithms was inspired through the observation of swarm colonies and specifically ants (Beckers et al, 1989). Insects like ants are social. That means that ants live in colonies and their behaviour is directed more to the survival of the colony as a whole, rather than to that of a single individual. Most species of ants are blind. However, while each ant is walking, it deposits on the ground a chemical substance called pheromone (Dorigo & Caro, 1999). Ants can smell pheromone and when choosing their way, they tend to choose, in probability, paths with high pheromone density. The ants using the pheromone trail have the ability to find their way back to the food source. The pheromone evaporates over time. It has been shown experimentally (Dorigo & Maniezzo, 1996) that the pheromone trail following behaviour can affect the detection of shortest paths. For example, a set of ants built a path to some food. An obstacle with two ends was then placed in their way, such that one end of the obstacle was more distant than the other. In the beginning, equal numbers of ants spread around the two ends of the obstacle. Since all ants have almost the same speed, the ants which choose the path of the nearer end of the obstacle return before the ants that chose the path of the farther end (differential path effect). The amount of pheromone deposits by the ants on the shortest path increases more rapidly than the farther one and so, more ants prefer the shortest path. Finally, with time the pheromone of the longest path evaporates and the path disappears. This cooperative work of the colony determines the insects' intelligent behaviour and has captured the attention of many scientists and the branch of artificial intelligence called *swarm intelligence* [1, 4].

### Artificial Ants (ACO)

Now in artificial life, the Ant Colony Optimization (ACO) uses artificial ants, called agents, to find good solutions to difficult combinatorial optimization problems (Bonabeau, Press). The behavior of artificial ants is based on the traits of real ants, plus additional capabilities that make them more effective, such as a memory of past actions. Each ant of the "colony" builds a solution to the problem under consideration, and uses information collected on the problem characteristics and its own performance to change how other ants see the problem.

Compendiously, ACO algorithms are based on the following ideas:

- Each path followed by an ant is associated with a candidate solution for a given problem.
- When an ant follows a path, the amount of pheromone deposited on that path is proportional to the quality of the corresponding candidate solution for the target problem.

- When an ant has to choose between two or more paths, the path(s) with a larger amount of pheromone have a greater probability of being chosen by the ant.

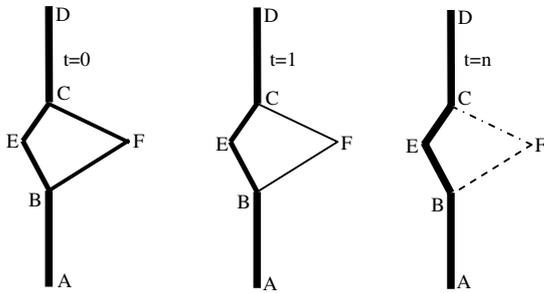


Figure 1: The ACO algorithm process

Let see an example of artificial ants' movement. We suppose that at time  $t=0$ , a number of ants are moving from point A (colony) to D (food) as it depicted in Figure 1. When ants arrive to point B they have to choose between BEC and BFC route. Initially the pheromone trail is the same for two alternative routes, so half of them will choose the first route and rest of them the second one.

The ants which chose the BEC will return in shorter time than the rest of them. That means, that the pheromone trail was deposited on BEC route evaporates less than BFC route. At time  $t=1$ , ants start again their route to the food. When they arrive in point B, the pheromone trail in BEC will be stronger than in BFC route, so more ants will choose the first route. After several cycles the pheromone trail in BFC, completely evaporates and all ants choose the BEF trail which is the shortest path.

## CASE STUDY

In this context, a suburb of Athens was chosen as the case study area. The municipality of Athens has empirically divided its area in about 145 solid waste collecting programs. Figure 2 illustrates one of these collecting programs.

This area of Athens comprises a region of about 0,5 km<sup>2</sup>, with a population of more than 8500 citizens and a production of about 3800 tones of solid urban waste per year, according to the latest statistics taken by the municipality of Athens.

Figure 2 also illustrates the approximately 100 loading spots. Any garbage truck that is responsible for the collection of the solid waste in that given area must visit all in order to complete its collection program.

The definition of these loading spots is beyond the scope of the present paper, but their placement is empirically able to cover the needs of the citizens. Additionally, any difficulties that the truck might face while following a given route or any other information

that could be useful to future considerations during the design and implementation of alternative route is recorded and available for further utilization.

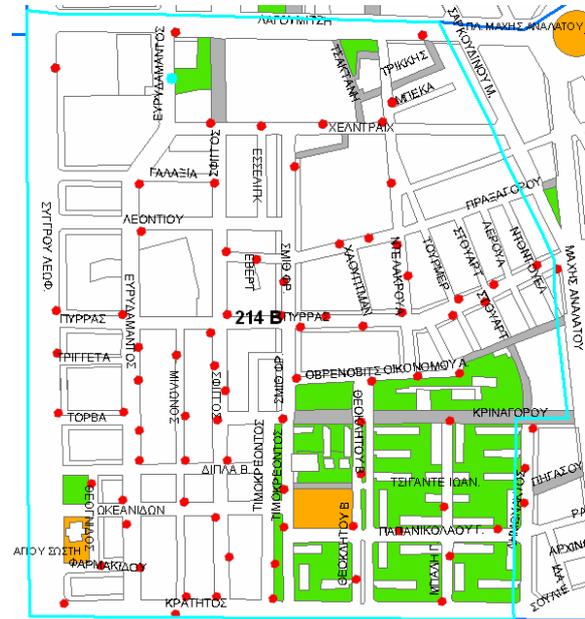


Figure 2: The suburb of Athens used in our experiments

According to the above, the urban solid waste collection and transport is a complex problem with many limitations. Minimization of cost means minimization of collection time and not necessarily choosing the minimal route. There is a crucial set of factors, such as the route traffic, the width of the roads that a specific route contains, the number of turns, the parked cars that in many cases block the smooth traffic flow, etc.

On the other hand, each garbage truck is able to collect a specific quantity of solid waste due to its limited waste capacity. So, the collected area, considering all parameters for that part of the problem, should be fragmented to sub programs which produced quantity of solid waste, equal to or less than the capacity of each truck (max\_quantity). All these parameters are included in the transportation cost calculation model. Historical data provide us with the ability to extract the 24<sup>th</sup> hour distribution of each factor.

Therefore, the problem in our case, as it mentioned above, it can be classified as a Traveling Salesman Problem (TSP): "Given a set of  $n$  loading spots and the transport cost between any loading spots, the TSP can be stated as the problem of finding a minimal cost closed tour that visits each loading spot once".

## PROPOSED SOLUTION

In the proposed solution the ACO algorithm is tested in the solid waste collection and transport problem. Any

garbage truck must travel among a set of loading spots, passing from each bin only once. A colony of artificial ants is created and at first it randomly travels complete circuits that contain every loading spot of the given set. During the first step, local travel to the closer loading spots is favored. After a complete circuit is determined, “pheromone” is deposited on each of link. The amount of pheromone is inversely proportional to the length of the circuit; shorter distances receive more pheromone. The colony is then released to travel circuits again, but this time ants favor links with higher concentrations of pheromone in addition to the links that are shorter. The pheromone evaporates at a constant rate, and links that are not part of efficient overall circuits eventually fall out of favor. The ant approach to this problem also provides the advantage of backup routes. Since the ants are continuously exploring different paths, alternative routes already exist if the link between two loading spots becomes unusable (for example, if weather conditions or road construction constitute impossible the movement between two loading spots).

## Methodology

The schema, which was chosen for the solution of our problem, is the Ant cycle algorithm from (Dorigo, & Maniezzo, 1996; Dorigo and Caro 1999), where each ant is a simple agent with the following characteristics:

- Initially an ant is placed in every loading spot. The number of ants is equal to the number of loading spots.
- Every ant chooses the bin to go to with a probability that is a function of the movement cost between two loading spots and of the amount of trail pheromone.
- Movements to already visited loading spots are disallowed until a tour is completed.
- When a tour is completed, ants update pheromone on each edge (i, j) they visited.

As mentioned above, the optimization quantity is the collecting time and not necessarily the distance of the route. Thus, the truck movement cost between loading spot i and j, is a function of all separate costs for each factor which affects the track route:

$$d_{ij} = \alpha \cdot da_{ij} + \beta \cdot db_{ij} + \gamma \cdot dc_{ij} + \dots \quad (1)$$

Let  $\tau_{ij}(t)$  be the *intensity of trail* on edge (i,j) at time t. Each ant at time t chooses the next loading spot, where it will be at time t+1. Therefore, if we call an *iteration* of the ACO algorithm the n moves carried out by the n ants in the interval (t, t+1), then for every n iterations of the algorithm (which we call a cycle) each ant has completed a tour. At this point the trail intensity is updated according to the following formula:

$$\tau_{ij}(t+n) = \rho \cdot \tau_{ij}(t) + \Delta \tau_{ij} \quad (2)$$

Where  $\rho$  is a coefficient such that (1 -  $\rho$ ) represents the *evaporation* of trail between time t and t+n,

$$\Delta \tau_{ij} = \sum_{k=1}^m \Delta \tau_{ij}^k \quad (3)$$

Where  $\Delta \tau_{ij}^k$  is the quantity per unit of length of trail substance (pheromone in real ants) laid on edge (i,j) by the k-th ant between time t and t+n; it is given by:

$$\Delta \tau_{ij}^k = \begin{cases} \frac{Q}{L_k} & \text{if k ant uses edge (i, j) in its tour} \\ 0 & \text{Otherwise} \end{cases} \quad (4)$$

where Q is a constant and  $L_k$  is the tour length of the k-th ant.

The coefficient  $\rho$  must be set to a value <1 to avoid unlimited accumulation of trail (see note1). In our experiments, we set the intensity of trail at time 0,  $\tau_{ij}(0)$ , to a small positive constant c.

In order to satisfy the constraint that an ant visits all the n different loading spots, we associate with each ant a data structure called the *hlist*, that saves loading spots already visited up to time t and forbids the ant to visit them again before n iterations (a tour) have been completed. When a tour is completed, the *hlist* is used to compute the ant's current solution (i.e., the movement cost of the path followed by the ant). The *hlist* is then emptied and the ant is free to choose again.

$$\eta_{ij} = \frac{1}{d_{ij}} \quad (5)$$

We call *visibility*  $\eta_{ij}$  the quantity  $1/d_{ij}$ . This quantity is not modified during the run of the AS, as opposed to the trail which instead changes according to the previous formula (5). We define the transition probability from loading spot i to loading spot j for the k-th ant as

$$p_{ij}^k = \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{k \in \text{allowed}_k} [\tau_{ik}(t)]^\alpha \cdot [\eta_{ik}]^\beta} \quad (6)$$

where  $\text{allowed}_k = \{N - \text{hlist}\}$  and where a and b are parameters that control the relative importance of trail versus visibility. Therefore the transition probability is a trade-off between visibility (which states that close loading spots should be chosen with high probability, thus implementing a greedy constructive heuristic) and trail intensity at time t (which states that if there is a lot of traffic on edge (i,j) then this edge is highly desirable, thus implementing the autocatalytic process).

## CONCLUSIONS

In this paper, a new solution for the collection and a transport of the Solid Waste has been introduced. There is no simple solution to this kind of problems due to interactions between conflicting requirements.

Therefore, an innovative approach for Solid Waste Management, based on the ACO algorithm, has been

applied. This algorithm has been implemented in the C++ programming language environment.

The system was simulated and tested for different periods during the day, as the involved parameters are drastically changed.

Further on, the area under consideration had to be divided in a segments adequate for the execution of the ACO algorithm. In each segment, all possible collection routes are considered and the optimal one is identified.

Then, the results of the ACO algorithm were compared with the corresponding ones of those produced by the Genetic, Simulated Annealing, and Tabu Search algorithms. On the other hand, the standards of an existing empirical model used by the municipality of Athens, were used as a benchmark for the algorithmic methods.

In conclusion, a clear improvement in time and cost for waste collection and transport has been observed in the case of the ACO based algorithmic implementation. Comparing this method with Genetic Algorithms and Tabu Search, the ACO algorithm seems to function slightly better. On the other hand, Simulated Annealing showed the worst results, but still better than those produced by the empirical model.

#### ACKNOWLEDGMENTS

The current article presents the results of an innovative routing algorithm applied in the case of municipal services. The relative research project was co-financed by the Operational Programme for Education and Initial Vocational Training, a program of the Hellenic Ministry of National Education and Religious Affairs.

#### REFERENCES

- Battini R, Tecchiolli G, 1994, The Reactive Tabu Search Journal on Computing Vol 6, No. 2, pp 126-140
- Beckers R., Goss S., Deneubourg J.L. and Pasteels J.M., 1989, "Colony size, communication and ant foraging strategy", *Psyche* (Cambridge), 96(3-4), 239-256.
- Bianchi, L., Gambardella L.M., Dorigo M., 2002, „An ant colony optimization approach to the probabilistic travelling salesman problem". In Proceedings of PPSN-VII, Seventh Inter17 national Conference on Parallel Problem Solving from Nature, Lecture Notes in Computer Science. Springer Verlag, Berlin, Germany.
- Bonabeau E., Dorigo M., & Theraulaz G. "Intelligence: From Natural to Artificial Systems", Oxford University Press.
- Chen S., Smith. S., 1996 "Commonality and genetic algorithms". Technical Report CMU-RITR-96-27, The Robotic Institute, Carnegie Mellon University, Pittsburgh, PA, USA.
- Dorigo M. and Caro G.D., 1999, "The Ant Colony Optimization Meta-heuristic," in *New Ideas in Optimization*, D. Corne, M. Dorigo, and F. Glover, Eds. London: McGraw-Hill, pp. 11-32
- Dorigo M., and Caro G.D., 1999, "Ant Algorithms Optimization. *Artificial Life*", 5(3), 137-172.

- Dorigo M., and Maniezzo V., 1996, "The ant system: optimization by a colony of cooperating agents". *IEEE Transactions on Systems, Man and Cybernetics*, 26(1), 1-13.
- Ducatele F. and Levine J., 2001, "Ant Colony Optimisation for Bin Packing and Cutting Stock Problems," presented at UK Workshop on Computational Intelligence (UKCI-01), Edinburgh.
- Glover F. and Laguna M., 1992, "Tabu Search: in Modern Heuristic Techniques for Combinatorial Problems", Blackwell Publishing.
- Glover F., 1986, " Future paths for Integer Programming and Links to Artificial Intelligence", *Computers and Operations Research*, 5:533-549.
- Glover F., Kelly J. and Laguna M. 1992, "Genetic Algorithms and Tabu Search: Hybrids for Optimization Manuscript", University of Colorado, Boulder.
- Pham D.T. and Karaboka D., 2000 "Intelligent Optimization Techniques". Springer-Verlag, London
- Tarasewich P, and McMullen R.P, 2002, "Swarm Intelligences", *Communications of the ACM*, Vol. 45, No. 8, pp 62-67

#### AUTHOR BIOGRAPHIES



**NIKOLAOS V. KARADIMAS** was born in Athens, Greece and he graduated from Patra T.E.I. in 1995 with a Bachelor's degree in Electrical Engineering. He then went to receive a Bachelor's degree with Honors in Electronic Engineering and a Masters degree in Computer Science from Glasgow Caledonian University, Scotland in 1997 and 1998, respectively. He also received a Masters degree in Distributed and Multimedia Information Systems from Heriot-Watt University, Scotland in 1999. Since 2002 he is a Ph.D candidate in National Technical University of Athens. Furthermore, since 2001 he is teaching Informatics in Hellenic Army Academy and New York College, and since 2003 he is teaching in T.E.I. of Chalkida. He is a member of the Greek Chamber of Engineers, member of IEEE and member of IEE. His research interests are in the fields of Databases, Information Retrieval and Multimedia.



**GEORGIOS KOUZAS** was born in Switzerland in 1976. He received the B.Sc degree in the Department of electrical Engineering, National Technical University of Athens, Greece in 1999. In 2000, he worked in kapaTel, with the main responsibility of the development of mobile application, using the WAP protocol. Since 2001, he is a PhD candidate, Research Associate with the Multimedia Technology Laboratory of the Department of Electrical and computer Engineering of the National Technical University of Athens working in ERMIS (IST) project. His research interests are focused in the Area of network management

and web engineering. His research also includes multimedia applications, Computer Vision Systems, Information retrieval techniques and image processing.



**IOANNIS ANAGNOSTOPOULOS** was born in Athens, Greece in 1975. He received his diploma from the Department of Electrical Engineering and Computer Technology, University of Patras, Greece, in 1998 and his Ph.D

from the School of Electrical and Computer Engineering, National Technical University of Athens (NTUA), Greece, in 2004. Since 2001, he works as Research Associate in various research areas at the Multimedia Technology Laboratory, School of Electrical and Computer Engineering, National Technical University of Athens. He is also a visiting lecturer at the Department of Information and Communication Systems Engineering. His research interests is focused in the area of Intelligent Information Systems, Telecommunication Networks, Neural Networks and e-Commerce. His research also includes multimedia applications, web based tools and network management. He has published more than 40 papers in journals, books and conference proceedings. Dr. Ioannis Anagnostopoulos is a member of the technical chamber of Greece, IEEE and IEE.



**VASSILI LOUMOS** received the Diploma of Electrical Engineering and the PhD degree in Computer Science from the National Technical University of Athens (NTUA), Greece in 1975 and 1989. From 1975

to 1978 he was a research engineer at the Centre

National de Recherche Scientifique, Paris, France. From 1980 to 1982 he was with CGEE ALSHTOM as a field engineer. In 1990 he joined the faculty of the School of Electrical & Computer Engineering of the NTUA where he is presently a professor. He is teaching Multimedia Technologies and Computer Graphics. His research interests are in the fields of Multimedia, Computer Vision and Internet Navigation.

**ELEFTERIOS KAYAFAS** received his B.Sc. degree from Athens University in 1970, and the M.Sc. and



Ph.D. degrees in Electrical Engineering from the University of Salford, England, in 1975 and 1978, respectively. He worked with the Civil Aviation Service (1973-74) and with the Hellenic Aerospace Industry (HAI 1978-79). In 1979 he joined the

National Technical University of Athens (NTUA) as Lecturer at the Electrical Engineering Department, then Asst. Professor (1987), Associate Professor (1992) and finally Professor of Applied Electronics (1996). Professor Kayafas is a member of the Greek Chamber of Engineers, member of IEEE, member of IMEKO (International Measurement Confederation, member of TC-4 committee). His research interests are applied electronics, multimedia applications and multimedia communication systems. He has published 35 papers in journals and presented more than 100 works in conferences, related to the subjects of his interest.

# A DSS SIMULATION MODEL FOR OUTSOURCING STRATEGIES IN LARGE-SCALE MANUFACTURING

Romeo Bandinelli  
Dipartimento di Energetica  
"Sergio Stecco"  
Università di Firenze  
r.bandinelli@ing.unifi.it

Alessandra Orsoni  
School of Business Information  
Management  
Kingston University  
Kingston upon Thames  
Surrey KT2 7LB, UK  
a.orsoni@kingston.ac.uk

## KEYWORDS

Supply Chain Management, DSS, production planning, outsourcing strategies, production network management.

## ABSTRACT

The paper illustrates the design and use of a distributed simulation system for the assessment of competing outsourcing strategies in the context of large scale manufacturing. The work consists in the evaluation of modelling approaches for the systematic assessment of the candidate solutions in terms of their direct production costs and estimated production losses. The paper mainly focuses on the development of a simulation framework to describe a typical production system including a main contractor and several suppliers where jobs are exchanged on a daily basis. A simple application concerning a single contractor and four suppliers was built for testing purposes, the details of this implementation are presented in the paper.

## INTRODUCTION

Production planning and control have become increasingly critical activities, as competition in the markets is leveraging on an increasingly large multitude of factors ranging from product quality, to delivery times, pre-sales and after-sales services. Among all of the production and planning activities, scheduling decisions represent the final decision-making phase and typically require the plant and supply chain managers' intervention to accommodate any short noticed changes to maintain satisfactory levels of performance across the entire production system.

In the context of distributed production planning and scheduling, the formulation of suitable outsourcing strategies involves difficult trade-offs between the cost of outsourcing and the reliability of the external production resources. The decision parameters typically include the number of external production resources, the sizes of their operations, and the variability of their committed production capacities. The specification of these decision parameters frequently leads to the identification of multiple competing solutions (De Toni

et al., 2000), which need to be systematically assessed and compared. A simulation model capable of supporting this type of analysis should incorporate complex mathematical models representing the details of the suppliers behaviour and the interactions among the different supply chain stakeholders.

Simulation techniques are well-established means to assess and validate scheduling policies and their performance in the manufacturing context (Sargent et Al, 1999) (Chang et Al, 2001). However, nowadays simulation techniques are hardly ever used in the industry to evaluate or improve scheduling performance. In the authors' opinion, this is due mainly to two reasons. First, stand alone simulation is not capable of representing large-scale manufacturing enterprises and their distributed operation environment. Second, the development of scheduling software is often commissioned to small software houses, with limited capability to robustly test and validate large scale software solutions.

Only the integration of simulation techniques and scheduling software based on advanced techniques (such as artificial neural networks or other artificial intelligence techniques), into simulation models can properly describe the distributed environments typical of the European production systems and to adequately support modelling for the evaluation of outsourcing strategies.

The development of a full simulation model capable of describing the complexity of the relationships among the main firm and the different suppliers, of reproducing the suppliers behaviour, and of analyzing their performance in terms of the choice of outsourcing strategies, is not a simple task. The key steps required to achieve this goal are illustrated in the paper.

## RESEARCH CONTEXT

This research defines modelling approaches for the systematic assessment of the candidate outsourcing solutions in terms of their direct production costs and estimated production losses. The paper proposes a simulation framework for the representation of a typical

environment consisting of a main contractor and several suppliers with jobs being exchanged on a daily basis.

Preliminary research work has provided the statistical basis for estimating the production losses caused by the late completion of the outsourced work. Specifically, a statistical model has been devised for a sub-problem pertaining to the outsourcing of a single batch of intermediate products of a same type to a choice of  $n$  different subcontractors (Bandinelli and Orsoni, 2005). The paper illustrates how this statistical model can be integrated with a stochastic, discrete event simulation model of the main production processes to handle the systematic testing of more complex situations. Specifically the research addresses production network performance in the context of multi-product and multi-site manufacturing. The focus is on the assessment of different outsourcing strategies in terms of their direct production costs and of the expected production losses, for a large-scale manufacturer outsourcing intermediate operations to multiple neighbouring/satellite subcontractors.

In order to evaluate the performance impact of several different strategies, a distributed simulation environment is proposed, which consists of one or more simulation models representing both the main firm's and the suppliers' behaviour to support overall cost estimating. In particular, the stochastic behaviour of the suppliers can be modelled either through the punctual values produced by an external extractor, operating on the relevant probability distributions, or through a simulation model. Either way the information required of the suppliers' models includes their available production capacity, relative to the committed production capacity, as negotiated with the main contractor, and their estimated production rates.

## MODELLING FRAMEWORK

Starting from a single simulation model, where both the main contractor and the suppliers are modelled, the authors propose a distributed framework where different simulation packages can be used to model the different stakeholders of the chain. These include the option of externally representing the suppliers through an extractor operating on their statistical distributions.

The idea behind the modelling framework is that, once the information that needs to be exchanged between the main contractor and the subcontractors has been defined, these can be considered as separate shells, and modelled independently. The definition of the information that needs to be exchanged should include all the factors that the main contractor needs to account for, when deciding where to allocate or commission a particular job. It should also include all the factors that the main contractor requires in order to update the scheduling of the main production process, once the supplier has completed the outsourced job.

By isolating the different components it is possible to build a modelling infrastructure which, even though initially developed as a single model, can easily be exported into a distributed environment by means of HLA or other Inter Process Communications (IPC) standard.

Building from previous research, where the effectiveness of job allocation was evaluated based on the direct costs of outsourcing and on the corresponding production losses (Bandinelli and Orsoni, 2005), the proposed framework requires the following exchange of information between the main contractor and the designated suppliers:

- direct production cost
- agreed delivery time
- existing stock of finished product to compensate for late delivery
- current capacity utilization for the subcontractor
- forecast of future main contractors needs for a product of the same type
- a supplier-specific factor, based on historical records, indicating its reliability on the delivery times.

Part of this information, when available, represents a major advantage for the main contractor, while other parts may be an advantage for the suppliers. Information sharing will occur to a certain degree which is linked to the level of affiliation between the suppliers and the main contractor. In any case the model will have to support the exchange of the entire set of information listed above, using the XML format, whereby a standard value will be assigned to all the variables that the different stakeholders may not wish to share.

In the example application described later in the paper, the set of information feeds into a cost function which is briefly characterized in the following section. More information can be obtained in (Bandinelli et al., 2005)

## ESTIMATING COSTS AND RISKS: COST FUNCTION DESCRIPTION

Referring to a production unit consisting of a specified batch of production with a specified set of operations that are intended for outsourcing, the total cost of outsourcing can be split into a direct cost, consisting of the actual cost of external processing, and an estimated production loss, associated to the cumulative probability of stock-out events due to late job completion times.

### External Processing Costs

For ease of calculation the direct cost of production for the outsourced jobs can be broken down in terms of production capacity and time requirements as indicated in equation 1.

$$C_e = \sum_{ij} U c_i \times C_i \times T_{ij} \quad (1)$$

where:

$U c_i$  is the cost of the job per unit capacity and per unit time for the  $i$ -th sub-contractor.

$C_i$  is the capacity committed by the  $i$ -th sub-contractor.

$T_{ij}$  is the time required for the  $i$ -th sub-contractor to complete the  $j$ -th operation (it is a stochastic variable).

The unit cost,  $U c_i$  needs to be estimated for each case based on the type of operations required, on the number of sub-contractors involved, and on the capacity of the sub-contractor. A correlation among these variables is currently being established, as part of parallel research, using Artificial Intelligence (AI) techniques based on Neural Networks (ANNs).

### Production Losses

The estimation of the production losses caused by the late completion of the outsourced work and possible stock-out events can be addressed building from the probability distributions associated to the production and consumption rates of each subcontractor and of the main contractor, respectively. These can typically be built from historical data for each plant.

A brief description of the equations required to estimate the expected production losses for the general case of a main contractor allocating a batch of a single product type to  $n$  subcontractors is provided in the remainder of this section. Further details on the statistical model may be found in some of the referenced work (Bandinelli and Orsoni, 2005).

The total number of units to be externally processed is  $O_{tot}$ , of which the  $i$ -th subcontractor is allocated a specified quantity  $O_i$ . The delivery time associated to each subcontractor is a cumulative function of the production rate and available production capacity over the required processing time. These values may be averaged if the processing time is sufficiently short compared to the time-scale of the other simulated processes.

The expected time of delivery for the entire outsourced quantity, from the perspective of the main contractor, can be estimated as a weighed average of the delivery times calculated for the different subcontractors, where the weights are the sizes of the jobs allocated to the different subcontractors (i.e. the different  $O_i$ s). It is the particular combination of delivery time and consumption rate at the main contractor's (i.e. the rate at which the intermediate products are used and therefore need to be fed into the

main production process) that determines whether a stock-out event may occur and how long it is likely to last. Building from the probability distributions associated to the processing times of the different contractors, including the main contractor, it is possible to determine which combinations of parameters lead to stock-out events and estimate their probability of occurrence. In particular, for each level of consumption rate  $C_x$  it is possible to identify the earliest delivery time that will cause a stock-out event, and that is given by equation 2:

$$t(x) = \frac{L - L_{min}}{C_x} \quad (2)$$

where:

$L$  is the stock level at main contractor's at the beginning of the time period.

$L_{min}$  is the minimum stock level at the main contractor's

$C_x$  is the consumption rate considered.

By discretizing the probability distributions associated to the consumption rate and to the delivery time for the outsourced production, it is possible to estimate the production losses associated to  $C_x$  for each interval of delivery times that may cause a stock-out event, given the unit loss for missed production time ( $PL_U$ ). The estimated total loss ( $PL_{TOT}$ ) can be expressed according to equation 3:

$$PL_{TOT} = \sum_x \sum_y PL_U \cdot [t_y - t(x)] \cdot p_y \cdot p_x \quad (3)$$

where  $p_y$  and  $p_x$  are the probabilities associated to the  $y$ -th arrival time interval (in the vicinity of  $t_y$ ) and to the  $x$ -th consumption rate interval (in the vicinity of  $C_x$ ), respectively. The sizes of such intervals depend on the criteria used for the discretization of the corresponding probability distributions, which typically are case-specific.

It should be observed that the summation (equation 3) should only be extended to the delivery times ( $t_y$ ) exceeding  $t(x)$  as those are the ones that determine a stock-out event. The difference  $[t_y - t(x)]$  represents the duration of the stock-out event.

Given the complexity of the calculations involved in the estimation of the production losses, even for very simple application cases, it is important to automate their computation procedure for its use in the systematic assessment of alternative outsourcing strategies. The modelling framework presented in the next sections of the paper incorporates the cost functions within the core of the simulation model where they can be systematically evaluated at each scheduled parameter update.

## MODEL DESCRIPTION

In order to test the modelling framework, a simple model was developed in the Arena® package, distributed by Rockwell. This model represents the behaviour of a main contractor with three different suppliers.

The production process of a single product type is considered, which consists of five phases. The first and the last phase are completed directly at the main contractor's, while the other phases are outsourced to local subcontractors. A flow diagram of the production processes is represented in figure 1. A daily exchange of intermediate parts occurs at fixed times during the day, as long as a threshold batch size requirement is met. If the daily number of parts to be processed is less than the required minimum, the intermediate parts are held in stock until the next day.

In order to choose the most effective combination of scheduling policies and outsourcing strategies, a cost function was introduced as discussed in the previous section. The application of such a function allows to simultaneously assess both the effectiveness of the cost function itself and the performance of the model.

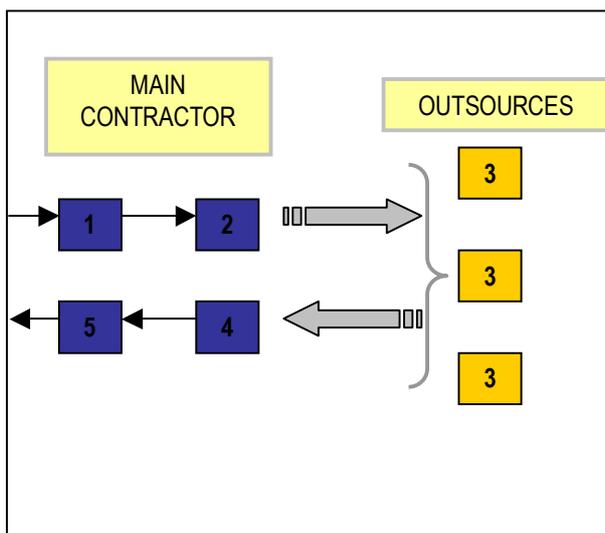


Figure 1: Flow Diagram of the Production Process

The information exchange between the main contractor and the suppliers was represented by means of an Arena® built-in VBA block. By these means the logics for the main contractor and for the different supplier models can be separated, even though they all run within the same software application. In particular the VBA allowed to reproduce the cost function, as described in earlier sections, without characterizing the models specifically for that function.

### Information Management

The model, as it stands, is locally run, however the infrastructure for the exchange of information has been designed to be portable for direct use in a distributed

environment. Specifically, each shell, whether intended to model the main contractor's or one of the subcontractor's behaviour, includes a "buffer" for intermediate products. While such a zone physically represents a storage facility for in-coming or out-going jobs, waiting to be transferred, it is also used to retain/store the information pertaining to the different stakeholders: it is based on such information that job allocations decisions are made. Specifically, a number of variables have been defined in Arena® for each of the stakeholders: these are the variable that should be re-evaluated and updated whenever a job needs to be allocated for outsourcing.

Because the transfer of intermediate products occurs in a discrete fashion and is performed on a daily basis, subject to meeting a minimum batch size, the relevant information is averaged with respect to the number of jobs of a same type to be externally processed. Once the minimum batch size required for outsourcing has built up in the buffer, the current values of the relevant variables are transferred to the core of the model, where the statistical cost function is implemented and used to evaluate the most effective outsourcing strategy.

The core of the model is also responsible for retrieving the information from the suppliers' side, which is necessary to make actual job allocation decisions (i.e. current stock level, production costs, etc.). Once the information has been retrieved, the core of the model uses it to evaluate the cost function and allocates the job to one of the suppliers. The decision is then communicated both to the main contractor and to the supplier.

The transfer of information is currently handled within the simulation model itself, however the model can be generalized for use in a distributed architecture, where an interaction system is applicable. An example of it, for instance, is HLA. The information dispatched to the two actors involved (i.e. the main contractor and the subcontractor) triggers a status update for all other actors, as well as the allocation of the job for delivery to the designated supplier.

The other information transfers pertaining to the completion of the outsourced jobs are handled exactly in the same way. These events cause the delivery of the processed batches from the supplier to the main contractor, and the update of the corresponding stock levels.

The current version of the model handles the delivery of externally processed jobs from the supplier to the main contractor instantaneously, regardless of the actual batch size. While this way of handling the return of processed jobs to the main contractor may not be generally applicable to different industrial cases and contexts, it is not believed to constitute a limiting

assumption for the purposes of testing the effectiveness of the proposed model.

### Time management

Time management and event synchronization among the different actors, just as the management of information transfers, have been designed for operability in a distributed environment.

The exchange of batches of intermediate products has been devised to occur on a daily basis and this is just one of the possible assumptions, which does not limit the model functionality in any way.

According to the framework design, the flow of jobs within each shell is independent of external events: it does not depend on external events and does not constrain their occurrence in any way.

Whenever the production cycle for a particular job requires external processing, the job is put on hold until it has been allocated to a particular supplier, and, more generally, until it is ready for the next processing phase.

Considering a model consisting of  $n$  independent simulators, time advancement can be based on a time-stepped logic, because the exchange of information between the main contractor and the subcontractors occurs on a daily basis and at fixed times, for instance every morning and every afternoon.

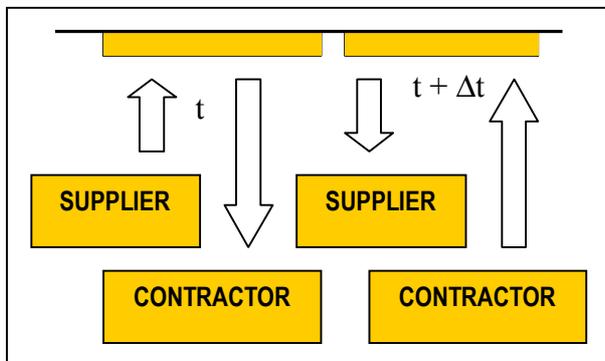


Figure 2: Example Information Flow with Time-Stepped Approach

Figure 2 shows a possible pattern of information flow between the suppliers and the main contractor. At time  $t$  (i.e. in the morning of a given day), the supplier updates the main contractor with its available production capacity and stock level. This information is received by the main contractor and used to formulate the outsourcing strategy for the next time step (i.e., the evening of the same day). At time  $t + \Delta t$ , the information flow is reversed, and the main contractor communicates the new job allocations to the suppliers. However, at time  $t + \Delta t$  a status update will still be sent from the subcontractor to the main contractor, as described above, for the allocation of the next jobs.

The definition of the information to be exchanged among the different actors at each step of the simulation clock depends on the designated IPC standard. When HLA is used, for instance, the reciprocal use of the information exchanged at a particular point in time is not possible, therefore a time-stepped approach is adopted whereby jobs are allocated for outsourcing to the different subcontractors based on their availability evaluated at the previous time step. Another option is to tailor the process of event generation to enable a two-way information exchange prior to the allocation of a job for outsourcing. More details on the use of the Next-Event approach for the generation of dummy events can be found in the referenced literature (Bandinelli et al., 2003).

When using other IPC infrastructures, such as the RMI or the CORBA standard, it is necessary to establish time-advancement logics which are standard-specific. The exchange of information among the different actors in the proposed framework will be the object of further research by the authors.

### CONCLUSION & FUTURE WORK

A modelling framework integrating a model based on the process statistics of the main contractor's and of the subcontractors' behaviour has been developed to support the effective choice of outsourcing strategies in large-scale manufacturing. A prototype model representing the suppliers' behaviour and availability, together with the main contractor's model, has been built and tested for performance on a single processor. This way, the main contractor can simulate the process of production scheduling and analyze the effects of competing outsourcing strategies, based on their overall costs. These costs account for process inefficiencies due to stock-out events and late deliveries.

The model was developed in Arena® maintaining the representation of the different supply chain actors independent of one-another. For this purpose a VBA block has been devised to separate the information to be exchanged between the main contractor and the different suppliers. The architecture to support the exchange of such information and their synchronization has also been presented. The model has been locally tested, with the implementation of simple dispatching and allocation rules for job outsourcing purposes. Finally, the cost function based on actual process statistics relevant to the different supply chain stakeholders, has been implemented in the model to support systematic cost estimating for different outsourcing strategies.

This work represent a further step towards the development of a multi-level shell capable of representing different levels of affiliation between the subcontractors and the main contractor, while keeping the same information exchange format and infrastructure. Concurrent research by the authors is

looking at Artificial Intelligence (AI) techniques based on Neural Networks (ANNs) for estimating the direct costs of the outsourced jobs. In particular correlations are sought between the unit cost of the outsourced work, the number of sub-contractors potentially available for the job, and the sizes of their production capacities. This will provide a complete DSS aimed at cutting the costs associated to late deliveries and stock-out events, and will effectively support strategic decision making for outsourcing.

## DEBITS

Arena® is a software of © 2005 Rockwell Automation, Inc. (<http://www.arenasimulation.com/>)

CORBA® is a registered trademark and the CORBA Logo™ is a trademark of the Object Management Group, Inc. Trademark Information

Java™ and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries.

## REFERENCES

- Bandinelli R., Orsoni A. (2005) Towards a DSS for Outsourcing Strategies in Large-Scale Manufacturing in *Proceedings of the 8th International Conference on Computer Modelling and Simulation (ICCMS)* pp. 138-144.
- Bandinelli R., Rapaccini M., Terzi S., & Macchi M., Proposal for a framework for production plants remote control: a preliminary test case, *Proc. 15<sup>th</sup> ESS Conf., Delft, NL, 2003*, 343-348.
- Chang Y, & Maskatsoris H., Supply chain modeling using simulation, *International Journal of Simulation Systems, Science and Technology*, 2(1), 2001, 24-30.
- De Toni A., Meneghetti A., The production planning process for a network of firms in the textile-apparel industry, *Int. J. Production Economics* 65 (2000) 17-32
- DMSO, "High Level Architecture, Federation Development and Execution Process (FEDEP) Model", version 1.4, June 9, 1999, <http://www.dmsomil/>
- Gunasekaran A, C. Patel, & E. Tirtiroglu, Performance measures and metrics in a supply chain environment, *International Journal of Operations and production Management*, 21(1), 2001, 71-78.
- Hwang K., *Advanced computer architecture – parallelism, scalability, programmability* (New York, NY: McGraw-Hill, 1993).
- Jain R., *The art of computer systems performance analysis* (New York, NY: J.Wiley & Sons, 1991).
- Java RMI, <http://java.sun.com/products/jdk/rmi/>, last access April 2005
- Juhasz Z, Turner S., Kuntner K., & Gezson M., A performance analyzer and prediction tool for parallel discrete event simulation, *International Journal of Simulation Systems, Science and Technology*, 4(1), 2003, 7-22.
- Kempf K. R., Uzsoy, Smith S. and Gary K.. *Evaluation and comparison of production schedules*. In: *Computers in Industry*, Vol. 42, (2000), pp. 203-220.
- Orsoni A, Bruzzone A., & Revetria R., Framework development for web-based simulation applied to supply chain management, *International Journal of Simulation Systems, Science and Technology*, 4(1), 2003, 1-6.
- Proud F.J., *Master scheduling. A practical guide to competitive manufacturing. Second edition* (New York, NY: J.Wiley & Sons, 1999).
- Sargent R.J., Validation and verification of simulation models, *Proc. Winter Simulation Conf., IEEE, Piscataway, NJ, 1999*, 39-48.
- Williams E.J. & Narayanaswamy R., Application of simulation to scheduling, sequencing, and material handling, *Proc. Winter Simulation Conf., IEEE, Piscataway, NJ, 1999*, 861-865.

## AUTHOR BIOGRAPHIES

**ROMEO BANDINELLI** received his laurea degree in Mechanical Engineering from the University of Florence in 2002 discussing the thesis "Remote Factory Control with Distributed Simulation". He is currently a PhD candidate in the Department of Energetic, Plants and Industrial Technologies at the University of Florence. His research interests include parallel and distributed simulation for industrial and supply chain applications, ICT, and business process re-engineering.

**ALESSANDRA ORSONI** is currently a senior lecturer in the School of Business Information Management at Kingston University (Kingston, UK). She received both her MS in Mechanical Engineering and her ScD in Engineering Systems Design and Innovation from the Massachusetts Institute of Technology (Cambridge, MA). Prior to joining Kingston University she was a research associate in the Department of Materials Science and Metallurgy at the University of Cambridge (Cambridge, UK) and worked as an assistant professor at in the Department of Production Engineering at the University of Genova (Genova, Italy). Her research interests include modelling, simulation, and AI techniques applied to the design and management of complex industrial systems.

# MULTI ECHELON SPARE PARTS INVENTORY OPTIMISATION: A SIMULATIVE STUDY

Simone Zanoni, Ivan Ferretti and Lucio Zavanella  
Dipartimento di Ingegneria Meccanica  
Università degli Studi di Brescia  
Via Branze 38, 25123 Brescia, Italy  
E-mail: [zanoni@ing.unibs.it](mailto:zanoni@ing.unibs.it)

## KEYWORDS

Inventory, Simulation, Spare parts, Multi Echelon.

## ABSTRACT

The research in this paper is motivated by a real life spare parts networks for complex technical systems in oil refining sector.

At the customers the availability of the installed technical systems often is essential for the primary process. Hence, they require a high availability. For improve the management of the spare parts inventory the main company (the customer) has developed a new strategies for pushing down to the supply echelon some inventory so as to increase availability of finished parts or components while the properties is still kept by the supplier until its usage, so as sharing in such a way the inventory costs.

In this paper after reviewing the literature on spare parts management the industrial case is discussed and based on it a model with closed queuing network approach will be shown. Due to its analytically difficultness the optimality of the parameters will be addressed via simulative way.

## INTRODUCTION

The importance of service parts management has increased in the past decades. One reason is the fact that system availability and high quality after sales service have become important criteria when selecting suppliers of technically advanced systems. A second reason is the increasing value of service part inventory investment. A survey by Cohen et al. (1997) reports that service parts inventories equal 8.75% of the value of product sales in their sample, being over \$23 mln. inventory investment on average.

The spare parts are needed to maintain an installed base of technical systems. Examples are aircrafts, locomotives, frigates and computer systems.

Service parts are often supplied via a multi-echelon distribution network, i.e. a hierarchical network of stocking locations through which service parts are supplied to the installed base at customer's sites. A reason to have a multi-echelon structure is the need for both local stocks close to the customer's sites in order to achieve fast supply and the need for stock centralisation to reduce holding costs. Cohen et al. (1997) report that

three-echelon networks are prevalent in their sample followed by two-echelon systems. Four-echelon networks occur in practice as well. There is a trend however to reduce the number of echelons and the number of locations per echelon in order to reduce fixed warehousing costs and service parts obsolescence costs. All these characteristics cause that service parts management is an increasingly important, yet complex task. A key challenge is to attain high availability of the installed base at low service costs. These service costs include costs for stock holding, warehousing, transportation, service engineers, repair shops and overhead.

## Literature review

In the literature, various ways to deal with finite capacity in service part networks have been discussed. One of these methods is to model the network as a closed queuing network (Jackson network, cf. Gross et al., 1978, 1983). This method provides very good estimations of the steady state probabilities in a closed network with fixed parameters, but the numerical algorithms involved make it difficult to find optimal stock levels for each location and each part type. Another approach is based on Markov processes; see Albright and Soni (1988), Gupta and Albright (1992) and Albright and Gupta (1993). A drawback of this approach is the fact that the number of states may become very large and that existing methods to reduce the model size to acceptable dimensions are rather rough.

A similar approach is developed by Avsar and Zijm (2000). They construct an excellent approximation for a two-echelon inventory model, where repair shops can be modelled as open Jackson queuing networks. However, their model considers only item-dedicated repair shops and is difficult to extend to multi-echelon model or model with different types of repair shops, as we consider.

Another possibility is to extend the VARI-METRIC method to deal with finite capacity by replacing the M/G/∞ queuing model for the repair shop by some finite capacity system, cf. Aboud (1996), Diaz and Fu (1997), Kim et al. (2000) and Perlman et al. (2001). They use their method to analyse the impact of finite capacity. Diaz and Fu (1997) show that finite capacity has a serious impact on system performance for a single

indenture, two-echelon system with only one central repair shop. They model the repair shop as a GI/G/k multi-class queuing system, where the part flow of one item type is modelled as one class in the queuing system. Although they discuss formulas for multi-server queues, their numerical results refer to single server queues only. In addition, they discuss an alternative method to plug in throughput times as observed in practice in the  $M/G/\infty$  model, so that waiting times are included. This approach is also used in the case study for the Caracas Metro subway system that Diaz and Fu (1997) present. Then the impact of finite capacity is less, but still significant, and as we mentioned already this procedure is not suitable for what-if analyses.

From the literature review it is quite clear how the more convenient way to model such spare parts inventory situation, considering also finite capacity repair shops, is the queuing network approach. There is an extensive literature on queuing analysis, see e.g. Kleinrock (1975) and Gross and Harris (1998). Applications are particularly in computer and telecommunication system analysis and the analysis of manufacturing systems, cf. Hall (1991), Papadopolous (1993) and Buzacott and Shanthikumar (1993).

### PROBLEM DEFINITION

The context from which we have taken inspiration for the development of the present research is real practice observed in Italy. In brief the situation concerns the field of the oil refining, and involve three different companies:

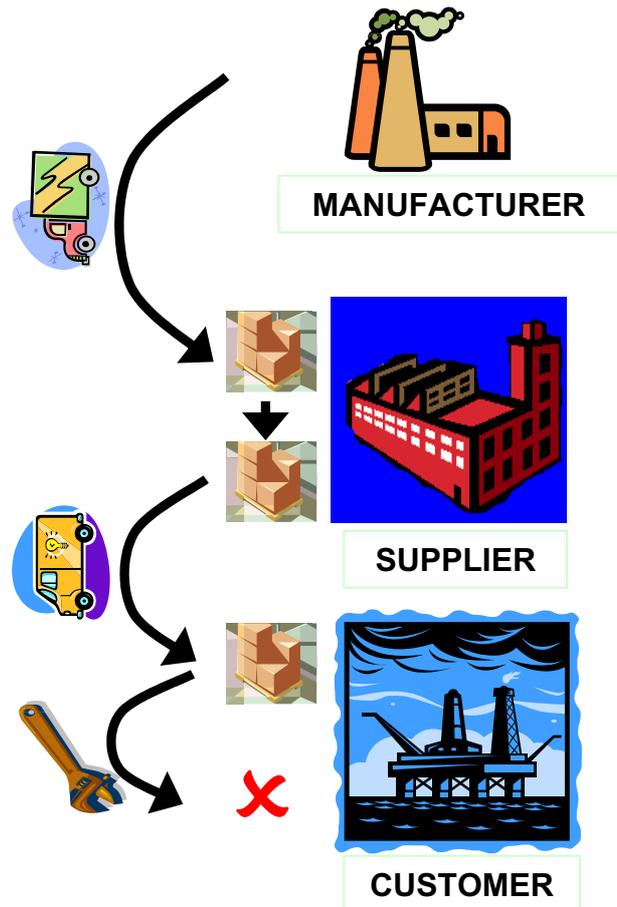
- Co1: industry of oil refining and distribution, situated in the gulf of Cagliari (Sardegna, Italy);
- Co2: supplier of special power plant installations;
- Co3: manufacturer of mechanical components, i.e. as particular pump equipment.

Recently the company Co1 promote some challenging objective, that in some cases involves also its suppliers. One of this programme is centered on inventory management optimisation. The company Co1 started to study the possibility of establish a new warehouse for stocking the spare parts that need with certain regularity or with short lead time: the warehouse could be conveniently located in the company Co2, so as it could optimize its production/assembly process located. In the same way company Co2 tried to react with company Co3, establishing there its components warehouse.

This is the problem that we will face in the following part of the paper.

### Problem Formulation

A rough representation of the problem described is following reported:



Figures 1: Scheme of the case study

We have reduced the system to three actors: customer (Co1), supplier (Co2) and producer (Co3). The productive structure of the customer is subject to breakdowns that happen at a given failure rate. Such breakdowns are repaired with spare parts supplied from a warehouse managed through a particular agreement with the supplier. The latter guarantees to the customer a given service level. The management of the inventory is entrusted to the supplier who controls the levels of supply and avoids the overcoming of specific limits (defined with contract). The components remains of property of the supplier until when it is used, and paid subsequently. Therefore the cost of maintenance to supply is subdivided in the following way:

- the financial cost is paid by the supplier;
- the physical storage cost is paid by the customer.

Such a situation may be also interpreted as a variant of the Consignment Stock inventory management practice. It is obvious that this practice may facilitate remarkably the customer, pull down the inventories with less costs. However it is not difficult to understand that this type of agreement assure in such a way the supplier that benefit of an exclusive contract supply. Moreover also the supplier has the component inventory whose level is guaranteed by a similar relationship to that customer-supplier with the manufacturer.

## Consignment Stock

The CS (Consignment Stock) policy is an industrial approach to stock management in the Supply Chain, firstly observed in the automotive field. Its principles and modelling are discussed in Abdel-Malek et al. (2002), Valentini and Zavanella (2003) and Braglia and Zavanella (2003), with reference to deterministic demand and to its extension to a stochastic environment. Let's see more in detail the cost structure under the consignment stock policy. The inventory cost per unit  $h$  is driven by two main components: a financial one ( $h_{fin}$ ) and storage one ( $h_{stock}$ ). Under a typical supply agreement, these costs are borne as indicated in Table 1.

Table 1: Inventory costs in traditional agreements

		Position of Raw Material	
		Supplier	Company
Relevant costs	Supplier	$h_{s,fin} + h_{s,stock}$	0
	Company	0	$h_{c,fin} + h_{c,stock}$

It should be noted that  $h_{c,fin} + h_{c,stock}$  is generally greater than  $h_{s,fin} + h_{s,stock}$ , mainly because of the financial component, which increases as it goes down the supply chain. The different situation brought about by CS is outlined in table 2.

Table 2: Inventory costs under CS policy

		Position of Raw Material	
		Supplier	Company
Relevant costs	Supplier	$h_{s,fin} + h_{s,stock}$	$h_{s,fin}$
	Company	0	$h_{c,stock}$

As can clearly be seen, the main difference is found in the case where the material has already been delivered to the company. In fact, the company incurs the storage cost, given that the material is located in its warehouse, but it does not yet sustain the financial cost. In fact, given that a good is formally purchased only after its consumption, the supplier is still bearing the financial opportunity cost. Thus, while calculating the total cost for the system, we may reasonably assume that the storage component  $h_{stock}$  of the total inventory cost may be considered as more or less identical for the supplier and the company, i.e.  $h_{c,stock} = h_{s,stock}$ . As a consequence, referring to the same average stock level, the total storage cost of the system is lower in the CS case (as we assume  $h_{c,fin} > h_{s,fin}$ ), even if a part of the cost is "shifted" onto the supplier. However, the supplier perceives some advantages as counterpart: the average quantity of the material stored in his own inventories decreases and, consequently, space is available to allocate other items; finally, the supplier may manage his production plan more flexibly, as it is not constrained by closed-orders. On the other hand, the company "sees" a lower inventory cost per unit, that is, only  $h_{c,stock}$  instead of the entire ( $h_{c,fin} + h_{c,stock}$ ).

## MODEL DESCRIPTION

In this section we introduce our model. Firstly, we present the assumptions and notations used in the model. Secondly, we describe the model for the evaluation of a stocking policy and lastly, we formulate the optimization problem.

### Assumptions and notations

We model the situation of the three independent companies. In particular we take into account the situation in which both the supplier and the customer keep spare parts on stock for the customer technical systems. We will consider a single type of spare parts type. These systems installed in the customer facility in number  $N$  are subject to failures. Failures occur according to Poisson processes with constant rates  $\lambda_f$ . Each time that a failure occur to one of the systems the customer maintenance staff is devoted to completely replace the system with a new one: if the system is available in the customer MRO inventory it could be replaced with a small lead time ( $\lambda_m$ ) otherwise it should be ordered to the supplier and it additionally requires a certain transportation lead times ( $\lambda_{t1}$ ). Moreover if the system is not available at the supplier finished product inventory an additionally assembly lead time is needed ( $\lambda_p$ ). It has to be noticed that with a stochastic probability the system may be repaired, but this operation it has been performed only by the supplier. So as some percentage of the fault system are sent back to the supplier for its supplier (after that they have been replaced with new one) with a certain lead time ( $\lambda_{t2}$ ). Repair operation are always cheaper than new system assembly, so when it is possible all the repairable system are repaired.

Parameters name used in the model, as well theirs symbol, are reported in the following table:

Table 3: Relevant parameters of the model

Parameter	Symbol
Failure rate [day]	$\lambda_f$
Average repair time [day]	$\lambda_m$
Average shipment lead time (supplier-customer) [day]	$\lambda_{t1}$
Average supplier assembly time [day/system]	$\lambda_p$
Average shipment lead time (manufacturer-supplier) [day]	$\lambda_{t2}$
Average remanufacturing time [day/system]	$\lambda_r$
Holding cost for systems at the customer stocking point (physical component) [€/system-day]	$h_{pc}$
Holding cost for systems at the customer stocking point (financial component) [€/system-day]	$h_{fc}$
Holding cost for systems at the supplier stocking point (financial and physical component) [€/system-day]	$h_{s1}$
Holding cost for components at the supplier stocking point (financial and physical components) [€/system-day]	$h_{s2}$
Fixed cost for the maintenance operation [€/repair]	$c_{fm}$
Set-up cost incurred by the supplier [€/Set-up]	$scs$
Cost for plant unavailability [€/hour of downtime]	$cvm$
Transport cost from customer to supplier and from supplier to manufacturer [€/system]	$ct$

## Objective function

With all the above defined assumptions and notations, we can now formulate our optimization problem as follows:

Minimize (TC)

With:

$$TC = \sum_{i=1}^7 C_i$$

Where:

$C_1$  = average opportunity loss for the downtime of the system (evaluated on the hypothesis that the part of the plant served by the system fell down couldn't work and the customer face the contribution margin of the production lost)

$C_2$  = average costs of maintenance operations (paid by the customer)

$C_3$  = average holding costs for the systems stocked in the customer warehouse (in the standard case fully paid by the customer and in the CS case the financial component paid by the supplier and physical one by the customer)

$C_4$  = average holding costs for the systems stocked in the supplier finished products warehouse (paid by the supplier)

$C_5$  = average holding costs for the components stocked in the supplier raw material warehouse (paid by the supplier)

$C_6$  = average transportation costs of the systems from the supplier to the customer (paid by the supplier)

$C_7$  = average set-up costs for the systems assembly in the supplier facility (paid by the supplier)

The objective is to find a stocking policy under which the average total cost (for the whole supply chain system) is minimized. In such a way two different approach may be pursued:

- traditional supply agreement
- CS supply agreement

Moreover we will take into account the whole supply chain as well as its different components that pertain to the different actors (i.e. supplier and customer).

We will consider for the three different stocking point the same control policy, i.e an (s,S) one (for additional detail Silver et al., 1998). Therefore we optimize 6 parameters:

- (s1, S1) for the customer;
- (s2, S2) for the supplier;
- (s3, S3) for the manufacturer.

## SIMULATION STUDY

As discussed in previous paragraph the inventory management of multi-echelon inventory items structures has been studied largely. However, some restrictions still exist on its applications to real case problems. To surpass some of these limitations a simulation model is developed in this work.

In particular in this study, we use Arena to build our simulation model. The embedded OptQuest is applied to search the optimal decision variables. OptQuest includes sampling techniques and advanced error control to find better answers faster, and incorporates algorithms based on Tabu search, scatter search, integer programming, and neural networks (Kelton et al, 2002). Preliminary experiments have been conducted to validate our simulation model as well as to evaluate the solution quality of OptQuest. It shows that the results between analytical and the simulation models are comparable. However, the disadvantage of using OptQuest is that it still takes long time to find the (near) optimal solutions.

To create uncertainty we use a fixed seed stream in Arena. Each source of uncertainty in this model has its own seed for generating random numbers.

More precisely, in our simulation experiments, we model as stochastic processes the failure process, the production/assembly process and lead times and

In this way the different models are comparable with each other and possible differences due to the use of different random seeds can be excluded.

## Simulation results

With the real data provided by the company (here reported slightly changed due to the confidentiality required) it has been possible to perform a complete simulative optimisation for all the parameters and for each of that perform a sensitive analysis. In particular we have taken into account the failure rate ( $\lambda_i$ ) as the main parameters for investigate the response of the system.

We will consider the plant of the company consisting of 50 equal pumping groups variously located in the refinery plant. Detailed data for each parameter is reported in Table 4.

Table 4: Parameters value used for the case study

Symbol	Traditional	CS
$\lambda_f$	POIS(15÷120)	POIS(15÷120)
$\lambda_m$	UNIF(2;4)	UNIF(2;4)
$\lambda_{i1}$	NORM(4,3)	NORM(4,3)
$\lambda_p$	NORM(2,1)	NORM(2,1)
$\lambda_{i2}$	NORM(5,4)	NORM(5,4)
$\lambda_r$	UNIF(10;5)	UNIF(10;5)
$h_{pc}$	9	5
$h_{fc}$		2
$hs_1$	6	6
$hs_2$	4	4
$cfm$	2	2
$scs$	200	200
$cvm$	45	45
$ct$	3	3

In the following figures it is possible to compare results of the optimised configuration for the two cases under analysis, while varying the failure rate. The simulation length as been set to 3650 period, i.e. considering a planning period of 10 years.

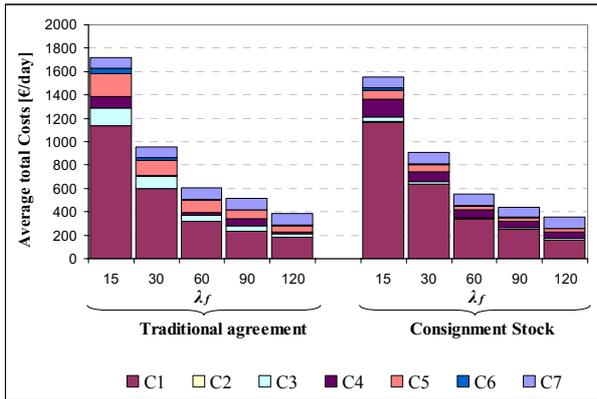


Figure 2: Average costs distribution while varying the failure rate within the two agreement option

With the CS agreement the average total cost is always lower than in the traditional agreement case. In the following table it is reported the percentage improvement in the objective function.

Table 5: Average cost improvement associated with the CS policy adoption

$\lambda_f$	15	30	60	90	120
Average cost improvement	-9.5%	-4.4%	-8.4%	-15.4%	-7.2%

It is also interesting to investigate how the system responds in terms of service level, measured as the effective available time of the plant on the requested time of its availability, while varying the failure rate with the two different policy.

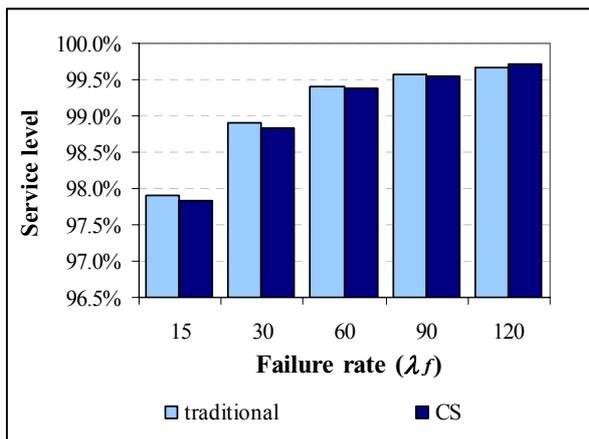


Figure 3: Service level while varying the failure rate

Moreover with the simulation optimisation performed it has been possible to design a trade-off curve between the service level and supply chain costs: curve is reported in figure 4.

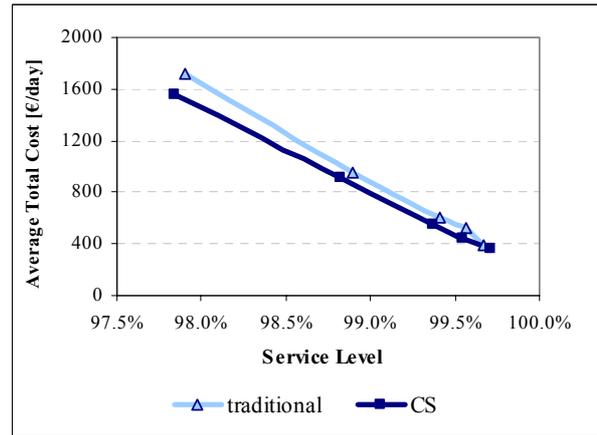


Figure 4: Trade-off curve between average total costs and service level

Also in this case the CS policy agreement perform better than the traditional one.

## CONCLUSIONS

Various process and manufacturing plants using complex machines often require large quantities of spare parts to guarantee high system availability which in turn results in excessive holding cost.

On the one hand, companies can find themselves carrying an excessive number of spare parts. On the other hand, if they were not available when needed, companies will face severe downtime consequences. As many parts are very expensive, critically important and their failure rates are so low that they are difficult to forecast, spare parts inventory management within these industries is one of the hardest problems to deal with.

In this paper we have presented a simulative approach for planning the spare parts inventory in a refinery plant, where a large amount of components have a failure rate that significantly affect the availability of the system. Moreover we have also shown how under a particular supply agreement, named Consignment Stock, the whole system perform better, both in terms of total costs and service level with a given failure rate.

The same approach and methodology may be applied to other supply-chain cases, thus offering the proper information to undertake the best course of action in the multi echelon spare parts inventory management.

## REFERENCES

- Abdel-Malek; L., Valentini G. and L.E.; Zavanella, 2002, Managing stocks in Supply Chains: modelling and issues, in Seuring S.; Goldbach M., Cost Management in Supply Chains, Heildeberg: Physica Verlag, pp.325-335.
- Aboud, N.E., 1996, "The Markovian two-echelon repairable item provisioning problem", Journal of the Operational Research Society 47, 284-296.
- Albright, S.C. and A. Gupta, 1993, "Steady-state approximation of a multi-echelon multiindented

repairable-item inventory system with a single repair facility”, *Naval Research Logistics* 40(4), 479-493.

Albright, S.C. and A. Soni, 1988, “Markovian multi-echelon repairable inventory system”, *Naval Research Logistics Quarterly* 35, 49-61.

Avsar Z.M. and W.H.M. Zijm, 2000, “Resource-constrained two-echelon inventory models for repairable item systems”, working paper, University of Twente, Enschede, The Netherlands.

Axsäter, S. 1990. “Modelling emergency lateral transshipments in inventory systems”. *Management Science*, Vol. 36, 1329-1338.

Braglia, M. and L.E. Zavanella, 2003, An industrial strategy for stock management in Supply Chains: modelling and performance evaluation, *International Journal of Production research*, vol. 41, no. 16, 3793–3808.

Buzacott, J.A. and J.G. Shanthikumar, 1993, *Stochastic models of manufacturing systems*, Prentice Hall, Englewood Cliffs, NJ

Cohen, M.A., Y-S. Zheng and V. Agrawal, 1997, “Service parts logistics: a benchmark analysis”, *IIE Transactions* 29, 627-639.

Diaz, A. and M.C. Fu, 1997, “Models for multi-echelon repairable item inventory systems with limited repair capacity”, *European Journal of Operational Research* 97, 480-492.

Gross, D. and C.M. Harris, 1998, *Fundamentals of queuing theory*, Wiley, New York

Gross, D. and J.F. Ince, 1978, “Spares provisioning for repairable items: Cyclic queues in light traffic”, *AIEE Transactions* 10(3), 1978, 307-314.

Gross, D., D.R. Miller and R.M. Soland, 1983, “A closed queueing network model for multi-echelon repairable item provisioning”, *IIE Transactions* 15(4), 344-352.

Gupta, A. and S.C. Albright, 1992, “Steady-state approximations for multi-echelon multi indented repairable-item inventory system”, *European Journal of Operational Research* 97(3), 340-353.

Hall, R.W., 1991, *Queueing methods for services and manufacturing*, Prentice Hall, Englewood Cliffs, NJ

Kelton, W. D., Sadowski, R. P. and Sadowski, D. A. (2002), *Simulation with Arena*, McGraw-Hill, Boston.

Kim, J-S, K-C Shin and S-K Park, 2000, “An optimal algorithm for repairable-item inventory systems with depot spares”, *Journal of the Operational Research Society* 51, 350-357.

Kleinrock, L., 1975, *Queueing systems*, Wiley, New York

Papadopolous H.T., C. Heavey and J. Browne, 1993, *Queueing theory in manufacturing systems analysis and design*, Chapman & Hall, London

Perlman, Y., A. Mehrez and M. Kaspi, 2001, “Setting expediting repair policy in a multiechelon repairable-item inventory system with limited repair capacity”, *Journal of the Operational Research Society* 52, 198-209.

Silver, E.A., Pyke, D.F. and R. Peterson, 1998, *Inventory Management and Production Planning and Scheduling 3<sup>rd</sup> Ed.*, John Wiley & Sons, New York.

Valentini, G. and L.E. Zavanella, 2003, The Consignment Stock of Inventories: Industrial Case and Performance Analysis, *International Journal of Productions Economics*, Vol. 81-82, pp.215-224.

## AUTHORS BIOGRAPHIES



**SIMONE ZANONI, Ph.D** is currently an assistant professor in the Mechanical Engineering Department at the University of Brescia (Italy). He graduated (with distinction) in 2001 in Mechanical Engineering and took his a Ph.D at the University of Brescia (Italy) in 2005. His main research interests are focused on Layout Design, Inventory Management and Closed-loop Supply Chain. His e-mail address is : [zanoni@ing.unibs.it](mailto:zanoni@ing.unibs.it).



**IVAN FERRETTI**, graduated in 2003 in Industrial Engineering at the University of Brescia (Italy). Since 2004, he is a Ph.D student at the same university. His main research interests are focused on Simulation, e-business, Supply Chain Management and Reverse Logistics. His e-mail address is : [ivan.ferretti@ing.unibs.it](mailto:ivan.ferretti@ing.unibs.it).



**LUCIO ZAVANELLA** graduated in 1982 in mechanical engineering at the Politecnico di Milano. Since 1986, he has been a researcher in the field of production systems at the Università degli Studi di Brescia. Presently, he is associate professor of Industrial Plants. His research activities mainly concern manufacturing systems, also with reference to environmental and safety aspects. His e-mail address is: [lucio.zavanella@ing.unibs.it](mailto:lucio.zavanella@ing.unibs.it).

# UML 2 AS A MODELLING LANGUAGE IN DISCRETE EVENT SIMULATION

Nicolas Knaak and Bernd Page  
Department for Informatics  
University of Hamburg  
Vogt-Kölln-Straße 30, 22527 Hamburg, Germany  
E-mail: [knaak, page]@informatik.uni-hamburg.de

## KEYWORDS

UML 2, Discrete Event Simulation, Model Driven  
Architecture, Software Engineering

## ABSTRACT

Due to the close relation between object-oriented modelling and the domain of discrete event simulation (DES) the Unified Modelling Language (UML) has in the past been frequently applied for simulation modelling. However, most of this work is restricted to particular applications or diagram types and based on the UML 1.x versions that had some drawbacks concerning dynamic modelling and formal precision. In this paper we provide a short overview of the new UML version 2.0 and sum up its benefits for discrete event simulation. We evaluate related work concerning the use of UML in DES and state some general directives of how to apply several diagram types. The treatment concentrates on UML 2 activity diagrams and relates them to the well-known DES modelling world-views. Finally we provide an outlook to further work that is especially concerned with generating simulation code from UML diagrams for our Java-based simulation framework DESMO-J.

## 1 INTRODUCTION

As Paul Fishwick pointed out in a panel discussion at the 2003 Winter Simulation Conference, “modelling is one of the primary components of simulation” (Barton et al. 2003, p. 2045). One of the most crucial steps during the course of a simulation is to build a *conceptual model* from system descriptions, observations, data, hypotheses and a-priori knowledge according to a problem definition (Page 1991, p. 13). The conceptual model is required to be *transparent, valid, understandable and preferably easy to transform into a computer implementation*.

In the history of discrete event simulation (DES) a great deal of different notations and formalisms like event graphs (Schruben 1983) or Petri nets (e.g. (Bause and Kritzing 1996)) have been proposed and applied for conceptual modelling. In an object-oriented discrete simulation context it seems obvious to make use of today’s standard graphical notation for object-oriented modelling, the Unified Modelling Language (UML).

Using UML stands to reason even more, since the new version 2.0 contains significant improvements concerning *dynamic behaviour modelling* as a key aspect in discrete simulation. *UML behaviour diagrams* are more common in practical and educational applications than formalisms like Petri-nets, but they are nevertheless given a quite concise semantic in version 2.0.

The idea of using *UML as a modelling language in simulation* is surely not new. However, existing work often either concentrates on particular application domains (e.g. network performance analysis (De Wet and Kritzing 2004)) or presents specific extensions and applications of certain diagram types (e.g. UML 1.x activity diagrams in (Oechslein et al. 2001)), often without regard to the UML standard and its inherent extension mechanisms.

In this contribution we state a proposal of how to *generally apply and extend several UML diagram types for DES modelling*. In particular, we show relations between the elements of UML 2 activity diagrams and the dominant world views of DES, i.e. event-scheduling, process-interaction, activity-scanning and transaction-oriented simulation (see e.g. (Page 1991, p. 25)).

The paper is organized as follows: In section 2 we provide a short overview of the new UML 2.0 and sum up important differences to the preliminary versions. We introduce the *modelling language’s different diagram types*, its main design principles and shortly discuss *metamodel and extension mechanisms*. In section 3 we review related work concerning the use of UML in simulation that is mostly based

on the UML 1.x versions. Section 4 contains some general directives of how several UML diagram types can be employed for DES. Section 5 concentrates on *UML 2 activity diagrams* and relates them to the well-known DES modelling world-views. Section 6 concludes the paper and provides an outlook to our further work that is especially concerned with generating simulation code from UML diagrams for our Java-based simulation framework DESMO-J (Lechler and Page 1999).

The UML 2 specific facts presented in the text are mostly based on the german textbooks “UML 2 Glasklar” by (Jeckle et al. 2002) and “Softwareentwicklung mit UML 2” by (Born et al. 2004).

## 2 DIAGRAM TYPES AND DESIGN PRINCIPLES OF THE UML

According to the *UML reference manual*, the *Unified Modelling Language* is “a general-purpose visual modeling language that is used to specify, visualize, construct, and document the artifacts of a software system” (Rumbaugh et al. 1999, p. 3). As (Jeckle et al. 2002, p. 10) point out, it is *not* “complete, not a programming language, not a formal language, not specialized to an application area and [...] first of all not a method or software process”.

The term “unified” refers to the fact that the UML is a unification of various popular object oriented (OO) modelling techniques from the beginning 1990s. This era, that is sometimes menacingly referred to as the “method wars” (see e.g. (Born et al. 2004, p. 13)), was ended when the authors of the most widely used OO techniques, James Rumbaugh, Ivar Jacobson and Grady Booch, “began to adopt ideas from each other’s methods” (Booch et al. 1999, p. xix) by the mid 1990s. As a result of these efforts the Unified Modelling Language version 0.9 was presented in 1996 (Jeckle et al. 2002, p. 12).

The development of UML 1.x was carried on under the patronage of the Object Management Group (OMG). Due to several drawbacks of the existing UML, a strongly revised version 2.0 has been proposed and recently adopted as the official UML version.<sup>1</sup>

As the major version step indicates, the UML 2.0 contains a large number of modifications compared to its predecessors. The new version intends to improve the language structure mainly in the following aspects: The *semantic precision* of the graphical notation is enhanced, thus making *direct execution*

*of models* possible, and the complex language specification was redesigned to become smaller and *more concise* (Jeckle et al. 2004).

In the context of simulation it is of special importance that the UML 2.0 contains numerous improvements concerning *dynamic behaviour modelling*: It introduces two new behaviour diagram types (*timing diagrams* and *interaction overview diagrams*), strongly enhances the expressiveness and formal semantics of the existing types (activity diagrams, interaction diagrams and, to a lesser extent, statecharts) and finally integrates the *notion of time* (Jeckle et al. 2004). Though the UML is still no formal language, there are ambitions to make UML models completely executable in the context of the *Model Driven Architecture* (MDA) (Born et al. 2004, pp. 273).

UML 2.0 contains a total of 13 diagram types to visualise different aspects of object-oriented models. According to (Jeckle et al. 2002, p. 16) the UML diagrams can be broadly divided into three classes:

- *Structural diagrams* serve to model the static structure of a system. Among them are *class diagrams*, *object diagrams*, *package diagrams*, *component diagrams*, *composition structure diagrams* and *deployment diagrams*.
- *Behaviour diagrams* are used to display the dynamic behaviour of objects or components at different levels of detail. This includes *use case diagrams*, *activity diagrams*, *statechart diagrams* and several *interaction diagram* types.
- *Interaction diagrams* are special behaviour diagrams that focus on the interactions going on between two or more objects in a system. Interaction diagrams can be divided into *sequence diagrams* and *timing diagrams* that emphasise the temporal order of interaction events on the one hand and *communication diagrams* that highlight the general structure of the cooperation between partners in an interaction on the other hand (Jeckle et al. 2002, p. 391). A new interaction diagram type is the *interaction overview diagram* that represents a mixture between activity diagrams and interaction diagrams showing the causal and temporal interplay between different interaction scenarios (Jeckle et al. 2002, p. 419).

Without diving into details of the respective diagram types, it becomes obvious that the UML is a complex and voluminous visual language incorporating a large number of different modelling techniques. Since complexity and partial inconsistency have been a major criticisms of the 1.x versions (Jeckle et al. 2002, p. 13), the language definition of the UML 2.0

---

<sup>1</sup>See <http://www.uml.org>.

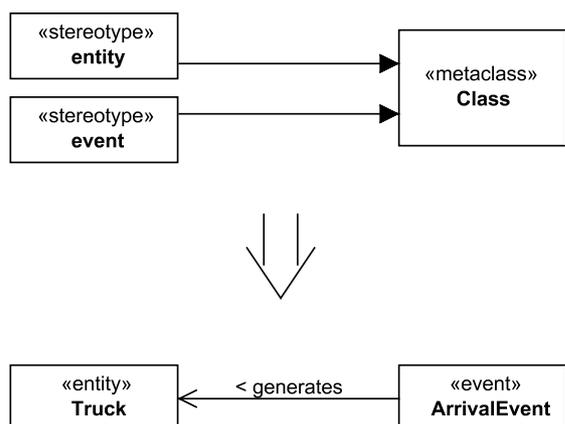


Figure 1: An example for simulation specific extensions of the UML using the stereotype mechanism. In the example, entity types and event classes are specified as extensions of classes in UML class diagrams. The new concepts are used in class diagrams by marking classes with the corresponding stereotypes (in imitation of (Jeckle et al. 2002, p. 94)).

adheres to some general principles that (Born et al. 2004, p. 10) identify as

- hierarchical and object oriented language design,
- language definition by meta-modelling,
- separation between vocabulary of concept and notation, and accordingly model and diagram.

The notion of meta-modelling refers to the fact that the concepts and notations of the UML are themselves defined in an object-oriented model that is expressed in terms of the UML (Born et al. 2004, p. 12). This object-oriented language definition makes extensions of the UML quite easy. As a software framework builds the basis for domain-specific software-applications, *UML concepts and diagrams can be customized and extended for special fields like simulation.*

Such extensions are either stated as extensions of the metamodel itself, or by using a lightweight mechanism called *stereotyping* (Born et al. 2004, p. 245). According to (Jeckle et al. 2002, p. 95) a stereotype is “*a class in the metamodel that is able to further specify other classes [...] by extension*”. The definition of stereotypes is expressed in UML notation using an “extension” arrow with a filled arrowhead (Jeckle et al. 2002, p. 93).

Figure 1 shows how stereotypes are used to implement simulation specific UML extensions. To represent entity types in DES models, the metaclass *Class* is extended with a stereotype «entity». We

state a further specification following (Spaniol and Hoff 1995, p. 9) by saying that objects of classes tagged with this stereotype are “able to actively move forward in simulation time”. Now entity types in class diagrams are marked by attaching the word «entity» in angle brackets to the respective model elements. Event classes and further extensions in the following sections are specified in a similar manner.

### 3 RELATED WORK

Traditionally there is a close link between object oriented modelling and the domain of DES. First of all this becomes obvious in the fact that the pioneering object-oriented programming language SIMULA (Birtwistle 1979) was originally designed for DES. Though conceptual object-oriented modelling in the simulation domain is quite frequently based on UML, we found relatively few literature on how to systematically apply the different UML diagram types during the course of a simulation study.

(Richter and März 2000) report on an attempt to apply UML and the Rational Unified Process to the design of simulation models. Their example is not a discrete event model but a simulator for evolutionary algorithms. During the different development phases they apply class diagrams to document the static structure of their simulation software, sequence diagrams to depict interactions between the simulator’s components and statecharts to model the component’s behaviour. It should be noted that in this work *UML is used only to document the simulation software* and not the domain model.

(Arief and Speirs 2000) present a UML tool that is able to generate simulation code for the process-oriented DES library JavaSim from class and sequence diagrams. Other simulation world views or diagram types are not supported, but the tool incorporates random variables and simulation statistics. An approach by (Marzolla and Balsamo 2004) concentrates on the performance evaluation of UML models by mapping them to the process-oriented simulation library *libcppsim*. They use deployment, use-case and activity diagrams for modelling and focus on the parametrisation and observation of UML models according to the OMG’s *UML profile for schedulability, performance and time* (Object Management Group 2005).

(Oechslein et al. 2001) apply modified UML 1.x activity diagrams to agent-based simulation modelling. They incorporate a large number of modelling elements like object-nodes and send-/receive-signal actions and define their own extensions for timed states and “emergency-rules” anticipating some UML 2.0 elements. However, the extended notation can

be handled and executed exclusively by their development tool SeSAM. (Köhler et al. 2000) employ class, statechart, collaboration and so called story diagrams to model and simulate production systems with their UML case tool Fujaba.

(De Wet and Kritzinger 2004) use UML 2.0 component and statechart models for the performance analysis of network systems. The formal semantic of these UML diagrams is enhanced with inscriptions in the *Specification and Description Language* that is common in communication systems engineering. The UML 2.0 compatible case-tool Tau Telelogic is used as an editor. There is a code generator based on the Velocity template engine that generates simulation programs for the process-oriented SimmCast framework.

UML is also applied in textbooks on DES. (Garrido 2001) uses mainly UML 1.x class, statechart and collaboration diagrams for process-oriented modelling. Some extensions for resource modelling are defined without regard to the UML stereotype mechanism. In (Page et al. 2000) UML 1.x class, activity, statechart and sequence diagrams are applied for the documentation of simulation models and -software. There are some ad-hoc extensions for modelling time consumption in the process-oriented world view but their semantics are not very concise.

## 4 APPLICATIONS OF UML 2 IN DISCRETE EVENT SIMULATION

Due to the *enhanced precision and expressiveness of the dynamic diagram types* the new UML 2.0 is a promising modelling language for DES. Based on the literature review and our experience in simulation teaching and practice, we are now presenting a summary of the benefit of UML 2 and its particular diagram types for DES.

In the first place UML is suited for the DES domain because of the event-based communication model underlying all behaviour diagrams (Jeckle et al. 2002, pp. 172). Due to the widespread use of the modelling language in industry and teaching, simulation modellers, developers and users benefit from UML diagrams as a common and simulation-software independent basis for documenting, visualizing and understanding the model structure (see also (Richter and März 2000, p. 395)). The different UML diagrams provide multiple views focusing on either *structural, behavioural or interaction-related aspects* of the model.

The quite concise semi-formal semantics of UML 2 behaviour diagrams with relations to for-

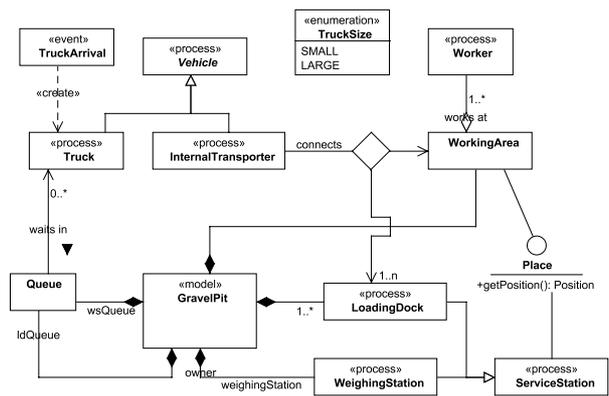


Figure 2: A class diagram of a “Gravel-Pit” model (adopted with modifications from (Birtwistle 1979)) from our simulation courses that serves as a continuing example in this paper.

mal models like automata and Petri-nets also provide support for the task of *model validation and verification*. The generation of executable code from static and dynamic UML models is another important means to narrow the gap between conceptual and computer models in simulation (Klügl 2001, p. 80). A current drawback of adopting UML 2.0 for model driven design<sup>2</sup> is the fact that there are hardly any case-tools at hand supporting the new version (Jeckle et al. 2002, p. 20).

Summing up our experience from literature and practice, the different UML diagram types might be best applied for the following purposes in simulation modelling:

- *Structural Diagrams: Class and object diagrams* are useful to depict entities and relations of the system under study during the conceptual modelling phase (see figure 2). Stereotypes can be employed to relate classes to notions from discrete simulation like e.g. processes or events. *Package and class diagrams* are also suitable to display the structure of simulation software during implementation. Though *component, composition structure and deployment diagrams* might render themselves useful in component-based and distributed simulation, we have not evaluated the benefit of these diagram types yet, but an example is shown in (De Wet and Kritzinger 2004, p. 5).
- *Behaviour Diagrams: Use case diagrams* can be applied to provide a broad overview of relations between actors and activities in the target system that builds the basis for more detailed

<sup>2</sup>The basic idea behind the notion of model-driven design is that “the model is the implementation” ((Selic 2003) cited from (De Wet and Kritzinger 2004, p. 2)).

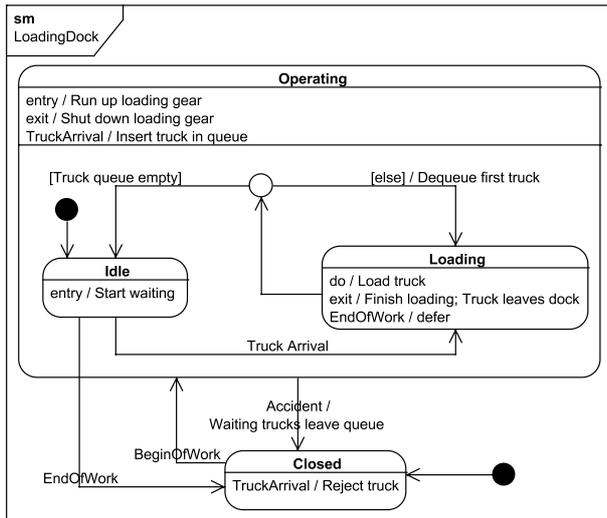


Figure 3: The “Gravel Pit” model’s loading dock modelled as a hierarchical state machine. The “emergency exit” behaviour shown here is a popular pattern in hierarchical state machines that can e.g. be found in (Köhler 1999, p. 38).

event or process descriptions. Behaviour modelling of individual entities and events is done with *statecharts and activity diagrams*. We use statecharts predominantly in an early modelling phase to build an *abstract conceptual model* of the target system’s state transitions (see the example in figure 3). This state model is then refined to one or more *activity diagrams* that are closer to the respective event- or process-oriented simulation style (see section 5). Due to their Petri-net like token semantic activity diagrams also support the validation of process-models either by manual “token-game” simulations or by mapping to executable code.

- *Interaction diagrams*: Since temporal aspects of interactions are of special importance in simulation, we preferably use *sequence and timing diagrams* in simulation modelling. Both diagram types might also be used in validation as a means to *visualize simulation traces* (see e.g. (Systä 2000)).

In the following we focus on UML activity diagrams and relate their modelling elements to the common DES world-views as they are supported by our simulation framework DESMO-J. Though most elements can be used in their standard form, we also specify some extensions using the stereotype mechanism.

## 5 RELATING ACTIVITY DIAGRAMS TO THE WORLD-VIEWS OF DES

According to (Jeckle et al. 2002, p. 199) *activity diagrams* are the *notation of choice for modelling processes*. While in software engineering the main application of this diagram type is the description of operations, use cases or business processes (Jeckle et al. 2002, p. 199), in DES activity diagrams are well suited for *modelling event routines in event-scheduling and life cycles of simulation processes in the process-interaction world view*. Due to features like concurrency, object flow and message-passing they are particularly appropriate to display the synchronization of two or more simulation processes using advanced modelling constructs related to the activity-scanning or transaction-oriented world view.

Activity diagrams have experienced strong changes at the transition from version 1.x to 2.0. In UML 1.x, they were a special case of statecharts emphasizing synchronous control flow instead of asynchronous event handling. In the new version, the statechart-like event-handling semantic of activity diagrams has been replaced by a *Petri-net-like token semantic* making this diagram type even more suitable for modelling concurrent processes.

The most straightforward application of activity diagrams in simulation is the description of event-routines in *event-scheduling simulation*. This world view is characterised by a *top down modelling perspective*. The model’s dynamic is specified in event routines of certain event classes that *change the state of the model’s entities at discrete points in time* (Page 1991, p. 30).

While event-graphs (Schruben 1983) display causal relations between event classes, the *detailed description of event routines* can be done with activity diagrams used basically as flowcharts. Figure 4 shows a simple example drawn from the “Gravel-Pit” simulation model. The event routine “Truck Arrival” is composed of elementary actions like creation and modification of entities or scheduling of further events, that we introduce stereotypes for. *Object nodes* (see below) might be employed to indicate *causal relations to other event classes*.

An execution of the event routine can be visualised as a token-flow similar to Petri-nets. A control-token is initially located at the initial node. Each action node fires when there is a token present on all of its incoming edges, consumes the tokens and produces new tokens on its outgoing edges.

Different from event-scheduling, the *process-interaction world-view* is characterised by a *bottom-up modelling perspective*, where the model dynamic

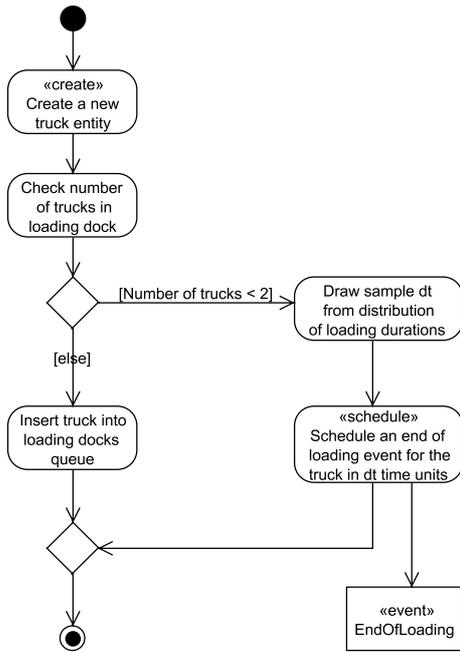


Figure 4: An example activity diagram describing the actions performed on the event “arrival of a truck in the gravel pit”.

is specified in terms of simulation processes and their life-cycles (Page 1991, p. 30). There is an alternation between active process phases where processes change the model state, and passive phases representing either wait states or *time-consuming activities*. The main operations in process interaction are

- *passivation* of processes to enter an unconstrained wait-state,
- *activation* of waiting processes, and
- the *hold* operation to enter a time-constrained wait-state.

These operations map quite obviously to *send- and receive-signal actions from UML 2 activity diagrams* (Jeckle et al. 2002, p. 214). The hold operation corresponds to the reception of a time signal that is depicted by an hourglass symbol (Jeckle et al. 2002, p. 215). Generally any time consumption is modelled using receive-signal actions, whereas normal action nodes correspond to active process phases without passing of simulation time.

Figure 5 shows two process classes from the “Gravel Pit” example, that synchronize via sending and reception of activation signals. When a truck arrives at the loading dock, it queues up in a “truck queue”. If an idle dock is available, the truck activates it to start loading. This can be denoted in UML by a send-signal action with the stereotype

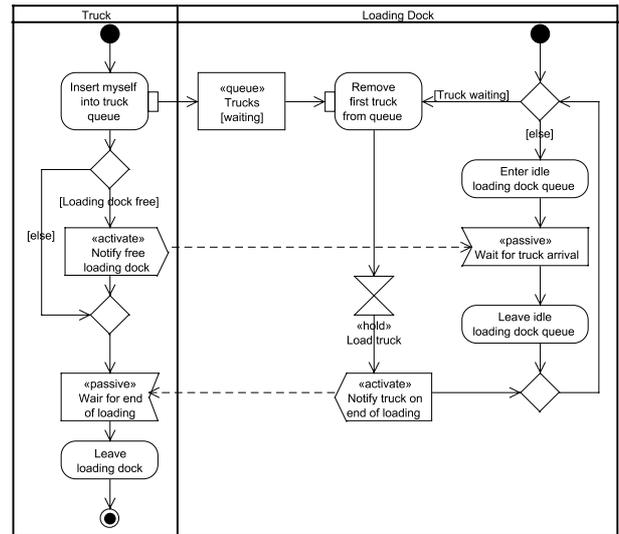


Figure 5: Synchronisation of trucks and loading docks in the “Gravel-Pit” model via sending and reception of signals.

**«activate»**. The truck then remains passive until loading is finished. The passive state is indicated by a receive-signal action with the stereotype **«passive»**. The notion of an action is somewhat misleading in this context, because token flow in the activity diagram is explicitly delayed (thereby forcing the process to wait) until the reception of a matching signal.

As mentioned before, *delays with predefined duration* are modelled using a *time signal reception node* depicted by an hour glass symbol with the stereotype **«hold»**. This node delays incoming tokens for a specified duration. One might also imagine that the modelled process schedules itself a future time signal when the node is reached and then waits for its reception. While in the example the service duration of the loading dock is stated implicitly by naming the corresponding time consuming activity (“Load truck”) UML also provides keywords *after* and *when* for relative and absolute time specifications. In stochastic simulation it should be possible to specify such durations in terms of the underlying random distributions (e.g. *after(Exp(3.0))* for exponentially distributed durations with a mean of 3 time units).

Note that the dashed arrows between the send and receive actions are not part of the standard UML notation, but only serve to clarify the direction of signal flow between the processes here. In larger simulation models it might not be sensible to display all communicating processes in a single diagram. In this case one would prefer to draw a detailed activity diagram for each process and use sequence or timing diagrams to specify exemplary interactions.

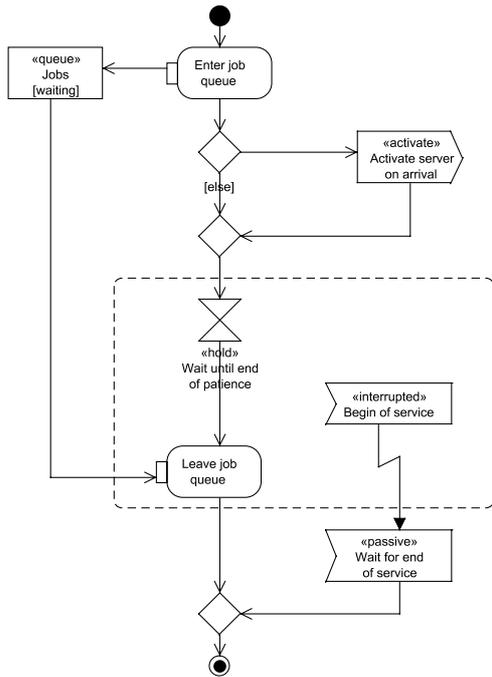


Figure 6: A customer with limited patience modelled with an interruptible activity region (Model adopted from (Page 1991, pp. 42)).

Another important requirement in process-interaction is the ability to model *preemption* and *interrupts* in queueing systems. Figure 6 shows an example of a customer process with limited waiting patience. After a certain waiting period the customer leaves the queue without being serviced. This can be represented using *interruptible activity regions* (Jeckle et al. 2002, pp. 241) in activity diagrams.

The interruptible activity region is depicted by a rounded rectangle with a dashed outline containing a “blizzard”-shaped interrupt edge (Jeckle et al. 2002, pp. 242). In the example, the customer process receives a *begin of service* signal that triggers the corresponding signal reception action at the beginning of service. This causes the interruptible activity region to be left via the interrupt edge. Different from a standard edge this step removes *all* tokens from the region. Thus no further actions within the region can be executed.

The ability of activity diagrams to display data flow is also interesting in the context of process-interaction. The main construct for *modelling data flow* are *object nodes* depicted by rectangles (Jeckle et al. 2002, pp. 218). When the outgoing edge of an action node is connected to an object node, execution of the action produces a so called *data token* that contains the result object of the execution. The data token is *stored in the object node* and might serve as input to another action with its incoming

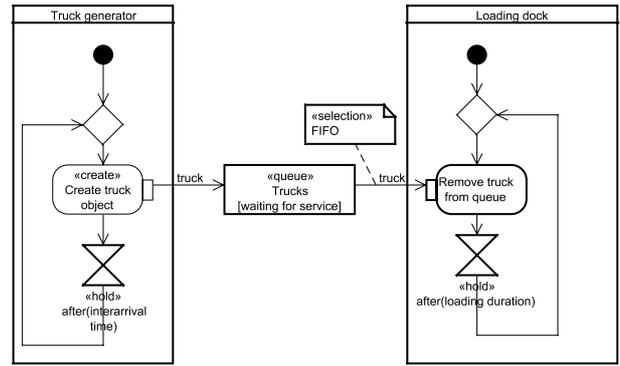


Figure 7: A queue modelled as an object node.

edge connected to the object node.

A useful property of object nodes is their ability to *buffer incoming data tokens*. This allows them to be used as synchronisation constructs in queueing and resource models. Figure 7 shows the interaction between trucks and the gravel pit’s loading dock as a simple *single server queueing system*. A truck generator process creates truck objects at a certain arrival rate and inserts them into a *queue modelled as an object node*. We use the stereotype `«queue»` to indicate that an object node has a queue semantic. The inscription names the type of entities to be stored in the queue. The optional inscription in square brackets is a condition constraining the state of all entities in the object node.

In the example, trucks are dequeued by the loading dock’s “remove from queue” action. The semantic of the edge connecting this action node to the queue is similar to standard edges in activity diagrams. When the queue node provides a data token (i. e. there is a truck waiting) and the standard edge above the action nodes provides a control token, the action node fires and removes one truck object from the queue. According to the UML 2 standard, there is a *note symbol* with the stereotype `«selection»` (Jeckle et al. 2002, p. 227) attached to the edge that indicates the selection strategy FIFO (or any other queueing strategy).

If no truck is present, the loading dock process waits for the arrival of the next truck. Note that queues used in process oriented simulation software (e.g. in the DESMO-J framework) often do not have these synchronization capabilities. In this case, one has to check the presence of objects in queues and synchronize processes explicitly via signals.

Object nodes also map directly to advanced synchronisation constructs used in the transaction-oriented world-view, that is especially suited for modelling production systems. Transaction-oriented models consist of static *blocks* (e.g. machines) and dynamic *transactions* (e.g. jobs) that change their

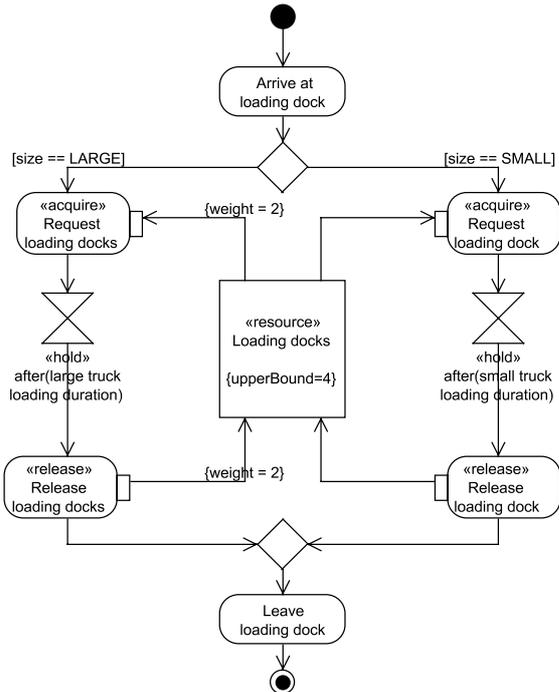


Figure 8: Interaction between trucks and loading docks modelled with a resource.

state while flowing through a network of blocks. Such models can easily be reduced to process-oriented models by mapping processes to transactions and blocks to *resources with internal wait queues* (Page et al. 2000, p. 17, p. 98).

An example of transaction oriented modelling with UML 2 activity diagrams is shown in figure 8. It displays the life cycle of a refined truck process from the “Gravel Pit” model. Here trucks might have two different sizes LARGE and SMALL. To become loaded, large trucks require two of 4 available loading docks while small trucks require only one. The loading docks are represented as a *resource with a certain capacity*.

The resource semantic of the corresponding object node is indicated by the stereotype `«resource»`. As before the truck process waits in its life cycle when there are too few resources available for service. In UML 2 the *maximum capacity of an object node* can be specified using the attribute `upperBound` written in curly braces. Similarly, the number of tokens an edge consumes or produces is specified using the attribute `weight` where the default weight of 1 is not stated explicitly.

In the activity scanning world view the model dynamic is described in a declarative way by specifying activities that are executed whenever their pre-conditions hold (Page et al. 2000, p. 17). Nevertheless this world view can be reduced to process-interaction by introducing conditional wait queues

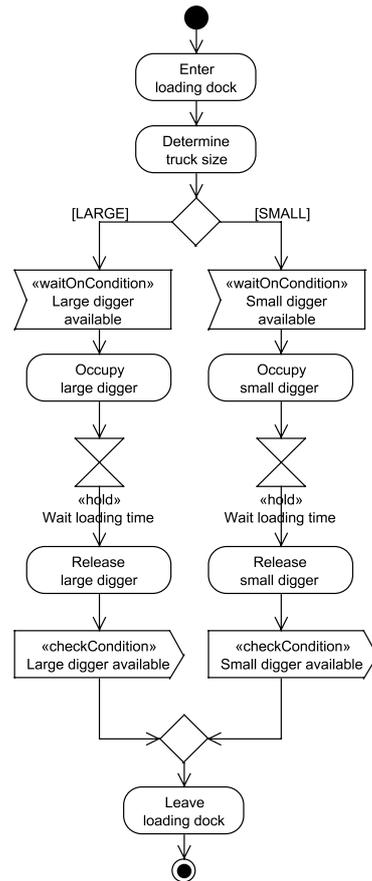


Figure 9: Interaction between trucks and loading docks modelled with conditional waiting.

(Page et al. 2000, pp. 102). Each condition is represented by a wait queue where processes wait in their life-cycle until the expected condition holds. To improve performance of such simulations, it is advisable to explicitly re-check these conditions only on changes of the model state.

We represent conditional waiting in UML 2 activity diagrams by receive-signal actions with the stereotype `«waitOnCondition»`. Re-checking a condition is requested by a send-signal action with stereotype `«checkCondition»`. Figure 9 shows an example from the “Gravel Pit” model where large trucks arriving at the loading dock wait for large diggers to become available while small trucks are only served by small diggers (adopted with modifications from (Page et al. 2000, pp. 145)).

## 6 CONCLUSIONS AND FUTURE WORK

In this paper we have discussed the benefits of the new UML version 2.0 for the domain of discrete event simulation. Due to enhanced conciseness and expressiveness of the behaviour diagrams, the new version

has become an even more promising candidate for a standard modelling language in DES. Therefore it is necessary to systematically evaluate for what purpose different UML diagram types can be applied best in simulation modelling. We include a proposal based on a literature review and our practical and teaching experience in section 4 of this paper.

To our impression the enhanced UML 2.0 activity diagrams relate very well to the dominant world-views of DES. While in event-scheduling, simple activity diagrams are used to specify event-routines, their main application domain is in our opinion process-interaction modelling based on the UML event model. Object nodes can be employed to represent synchronisation constructs in transaction-oriented or activity-scanning models. Though most elements of activity diagrams map directly to concepts from DES, some simulation-specific extensions (e.g. resources) were defined using UML's stereotype mechanism.

In our further work we are going to evaluate the UML 2 notation in our simulation courses based on a simulation textbook that is currently in preparation. Moreover we are developing a code generator that maps the activity diagram notation shown in section 5 to code for our Java-based simulation framework DESMO-J (Lechler and Page 1999). An extended version of the lightweight UML drawing tool *UMLet*<sup>3</sup> is used as a diagram editor that can also be integrated as a plugin into the popular *Eclipse*<sup>4</sup> development platform.

For code generation we employ a template-based approach similar to (De Wet and Kritzing 2004) using the *Velocity* template engine.<sup>5</sup> Template-based code generation provides the advantage that the destination language or framework can easily be exchanged due to a layered architecture in conformance with the "viewpoints" of the OMG's Model Driven Architecture (Born et al. 2004, p. 279). Different from (De Wet and Kritzing 2004) we use Java instead of SDL as inscription language, thus making activity diagrams more generally applicable at the cost of some formal precision.

At the conceptual level, a further evaluation of the OMG's *UML profile for schedulability, performance and time* (Object Management Group 2005) and its relation to the DES domain might be of interest (see also (Marzolla and Balsamo 2004)). Work in this area could possibly lead to a standard UML profile for discrete event modelling in the future.

## REFERENCES

- Arief, L. and Speirs, N.: 2000, A UML tool for an automatic generation of simulation programs, *WOSP 2000*, Ontario, Canada.
- Barton, R. R., Fishwick, P. A., Henriksen, J. O., Sargent, R. G. and Twomey, J. M.: 2003, Panel: Simulation: Past, present and future, in S. Chick, P. Sanchez and D. F. and D.J. Morrice (eds), *Proceedings of the 2003 Winter Simulation Conference*, pp. 2044 – 2050.
- Bause, F. and Kritzing, P.: 1996, *Stochastic Petri Nets - An Introduction to the Theory*, Advanced Studies in Computer Science, Vieweg, Braunschweig.
- Birtwistle, G.: 1979, *DEMOS, a System for Discrete Event Modelling on Simula*, Macmillan, London.
- Booch, G., Rumbaugh, J. and Jacobson, I.: 1999, *The Unified Modeling Language User Guide*, Object Technology Series, Addison-Wesley, Reading (MA).
- Born, M., Holz, E. and Kath, O.: 2004, *Softwareentwicklung mit UML 2 - Die "neuen" Entwurfstechniken UML 2, MOF 2 und MDA*, Programmer's Choice, Addison-Wesley, M"unchen.
- De Wet, N. and Kritzing, P.: 2004, Using UML models for the performance analysis of network systems, *Proceedings of the Workshop on Integrated-reliability with Telecommunications and UML Languages (WITUL)*, Rennes, Brittany, France.
- Garrido, J. M.: 2001, *Object-Oriented Discrete-Event Simulation with Java - A Practical Introduction*, Series in Computer Systems, Kluwer Academic / Plenum, New York.
- Jeckle, M. et al.: 2002, *UML 2 glasklar*, Carl Hanser Verlag, M"unchen.
- Jeckle, M. et al.: 2004, UML 2.0: Evolution oder Degeneration, *Objekt Spektrum* (3), 12–19.
- Kl"ugl, F.: 2001, *Multiagentensimulation - Konzepte, Werkzeuge, Anwendung*, Agententechnologie, Addison-Wesley, M"unchen.
- K"ohler, H.: 1999, *Code-Generierung f"ur UML Kollaborations-, Sequenz- und Statechart-Diagramme*, Diploma thesis, University of Paderborn.
- K"ohler, H., Nickel, U., Niere, J. and Z"undorf, A.: 2000, Integrating uml diagrams for production control systems, *Proc. of the 22nd International Conference on Software Engineering (ICSE)*, ACM Press, Limerick, Ireland, pp. pp. 241–251.
- Lechler, T. and Page, B.: 1999, Desmo-j – an object-oriented discrete simulation framework in java, in G. H. D. M"oller (ed.), *Simulation in Industry 099 – 11th European Simulation Symposium 1999 in Erlangen (Germany)*, SCS, Delft (Netherlands), pp. pp. 46 – 50.

<sup>3</sup><http://www.umlet.com>

<sup>4</sup><http://www.eclipse.org>

<sup>5</sup><http://jakarta.apache.org/velocity>

Marzolla, M. and Balsamo, S.: 2004, UML-psi: The UML performance simulator, *Technical report*, Universita Ca Foscari di Venezia, Dipartimento di Informatica.

Object Management Group: 2005, UML Profile for Schedulability, Performance and Time v1.1, *Full specification formal/05-01-02*, OMG.

Oechslein, C., Klügl, F., Herrler, R. and Puppe, F.: 2001, UML for behaviour-oriented multi-agent simulations, in B. Dunin-Keplicz and E. Nawarecki (eds), *Proceedings of the CEEMAS*, number 2296 in *Lecture Notes in Artificial Intelligence*, Springer, Berlin, pp. 217 – 226.

Page, B.: 1991, *Diskrete Simulation: Eine Einführung mit Modula II*, Springer, Berlin.

Page, B., Lechler, T. and Claassen, S.: 2000, *Objektorientierte Simulation in Java mit dem Framework Desmo-J*, Libri Books on Demand, Hamburg.

Richter, H. and März, L.: 2000, Toward a standard process: The use of UML for designing simulation models, in J. Joines, R. Barton, K. Kang and P. Fishwick (eds), *Proceedings of the 2000 Winter Simulation Conference*, pp. 394 – 398.

Rumbaugh, J., Jacobson, I. and Booch, G.: 1999, *The Unified Modeling Language Reference Manual*, Object Technology Series, Addison-Wesley, Reading (MA).

Schruben, L.: 1983, Simulation modeling with event graphs, *Communications of the ACM* **13**, 265–275.

Selic, B.: 2003, Brass bubbles: An overview of UML 2.0 and MDA, *Object Technology Slovakia*.

Spaniol, O. and Hoff, S.: 1995, *Ereignisorientierte Simulation - Konzepte und Systemrealisierung*, Internationale Thompson Publ., Bonn.

Systä, T.: 2000, Understanding the behavior of java programs, *Proc. of the 7th Working Conference on Reverse Engineering (WCRE 2000)*, Brisbane, Australia, pp. 214 – 223.

Prof. Page was cofounder and long year chairman of the working group “Informatics for Environmental Protection” in the German Computer Society (GI). One of his main interests is the development of Java tools for discrete event simulation. Prof. Page has been granted a significant number of research and development funds for public as well as industrial projects. Currently he is leading a funded research project on E-Learning in the context of computer simulation.

His E-Mail address is [page@informatik.uni-hamburg.de](mailto:page@informatik.uni-hamburg.de), and his personal webpage can be found at <http://asi-www.informatik.uni-hamburg.de/personen/page>.

**NICOLAS KNAAK** obtained his diploma degree in Computer Science from the University of Hamburg, Germany in 2002. Since then he has been working as a scientific assistant and PhD student in the simulation group led by Prof. Page at the University of Hamburg. His research interests are UML modelling and the verification and validation of Multi-Agent Systems. He is currently preparing his PhD thesis on the validation of Multi-Agent Systems by means of simulation and data-mining techniques.

His E-Mail address is [knaak@informatik.uni-hamburg.de](mailto:knaak@informatik.uni-hamburg.de), and his personal webpage can be found at <http://asi-www.informatik.uni-hamburg.de/personen/knaak>.

## AUTHOR BIOGRAPHIES

**BERND PAGE** holds degrees in Applied Computer Science from the Technical University (TU) of Berlin, Germany and from Stanford University, USA. Prof. Page acted as a scientific associate in the Environmental System Group at the German Federal Environmental Agency in Berlin before he was appointed professor for Applied Computer Science at the University of Hamburg. As permanent academic staff member at the Hamburg Department of Computer Science he is carrying out research and teaching in the field of computer simulation methods, tools and applications as well as in Environmental Informatics.

# WIND SPEED MODELLING AND SHORT-TERM PREDICTION USING WAVELETS

Piers R.J. Campbell and Ken Adamson  
Faculty of Engineering  
University of Ulster  
Shore Road, Belfast  
Northern Ireland  
Email: pr.campbell@ulster.ac.uk

## KEYWORDS

Wavelets, Wind Energy, Prediction

## ABSTRACT

The mathematical method of wavelets is explained and used to forecast wind conditions using short-term data collected at a site and referred to long term data from meteorological stations. We model the response time-series in terms of a multi-scale wavelet decomposition of the explanatory time-series. Preliminary results of this method, using hourly 10 minute averaged data from six locations in the British Isles, allow comparison with a linear regression method in terms of prediction errors over 21 days.

## INTRODUCTION

This paper examines the applicability of the mathematical method wavelets to the forecasting of wind speed for wind parks in the UK and Ireland. Wind energy is the fastest growing alternative fuel in the world today. Coupled with solar, hydro and biomass fuel the UK government hopes that 15% of the UK's energy will be generated by renewable energy by 2015.

A number of technologies have been applied to the forecasting or prediction of wind regime; these have included statistical approaches, Artificial Intelligence approaches and parallel computing. Of these a hybrid approach has to date produced the most satisfactory results, at least in terms of international application. Implementations of this technology such as WaSP have performed well when producing forecasts for specific geographical areas but have underachieved when applied to the UK and Irish markets. The reasons for the poor performance of these technologies may have to do with the different orographic conditions in the UK/RoI region. In addition the systems implemented have been designed specifically for the weather patterns of the Scandinavian countries and may require redevelopment to achieve satisfactory results in the UK.

The wavelet approach does not require the significant computing power of the NWP technique but does require some historical data from the target location and also a more substantial quantity of historical data from a near by location. This is normally achieved as follows; a small amount of wind speed data is collected at a site (this is normal procedure when assessing a locations suitability as a wind park installation) and that a nearby reference location, for example, a meteorological station, provides long term data. The Wavelet approach then models the data at the target site, known as the *response time-series* ( $y_t$ ), in terms of the data at the reference location, known as the *explanatory times series* ( $x_t$ ).

The forecasting or predicting of future wind speeds at a target site using data from a reference location is generally known in the wind energy industry as "measure-correlate-predict", MCP.

Linear regression is a popular industry method for constructing the statistical model which will predict future values of a response time-series, (Derrick 1992). Other approaches such as Artificial Neural Networks also rely on historical data in order to produce forecasts. As an alternative, we propose a wavelet method which takes a multi-resolution approach. We model the response time-series in terms of a multi-scale wavelet decomposition of the explanatory time-series. We provide some preliminary results of this method using hourly, 10-minute averaged data from six locations in the British Isles and show how our model compares with a linear regression method in terms of prediction errors over 21 days. Typically the industry will require accurate predictions across a range of time horizons; short term (between 2 and 48 hours) for energy trading (due for implementation during 2005) and for longer term forecasts (2 to 10 days) for maintenance scheduling. For a more detailed discussion of the wavelet methodology refer to (Nason *et al* 2001).

## A BRIEF INTRODUCTION TO WAVELETS

The following section is intended to give only a brief introduction the concept of wavelets, for a more detailed description we would suggest Burrus, Gopinath and Guo (Burrus *et al* 1998). Multi-resolution analysis provides the framework for examining functions at different scales. In a multi-resolution analysis a father wavelet,  $\Phi(x)$ , is a function constructed to approximate general functions to a certain scale by using shifted copies of itself. The mother wavelets,  $\nu(x)$  derived from the father wavelets, represent the difference between father wavelet approximations at two different scales. A mother wavelet is a localised oscillating function from which a family of wavelets,  $\nu_{j,k}(x)$ , can be constructed by dilation and translation, i.e.

$$\nu_{j,k}(x) = 2^{j/2} \nu(2^j x - k) \quad (1)$$

for integers  $j, k$ .

The dilation parameter  $j$  controls the scale (or size) of the wavelet and the translation parameter  $k$  controls the location of the wavelet. For suitable mother wavelets,  $\nu(x)$ , the set  $\{\nu_{j,k}(x)\}_{j,k}$  provides a basis that can be used to approximate functions, i.e.

$$f(x) = \sum_j \sum_k d_{j,k} \nu_{j,k}(x) \quad (2)$$

where  $d_{j,k}$  are the wavelet coefficients.

Wavelets have enjoyed particular success in representing complex various types of complex signals and as a result have been implemented in domains such as image compression. Wavelets are particularly useful for representing signals with discontinuities due to their excellent localisation ability.

For some time-series wavelet packets may be of more use as they provide a wider choice of decompositions of the frequency domain. A wavelet packet is a particular linear combination of wavelets that retains many of the orthogonality, smoothness and localisation properties of wavelets (Wickerhauser, 1994).

Using wavelets (or wavelet packets) in our model allows us to attach a physical interpretation to the model. For example, a wavelet packet is given in Figure 1 which could be used to represent daily variation in wind speeds over the past 2.75 days. The

wavelet packet provides valuable information about which components in the explanatory time series drive the response time series, i.e. which types of oscillatory behaviour in  $x_t$  influence  $y_t$ .

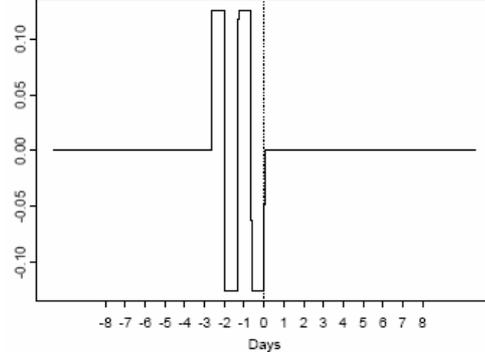


Figure 1: A Haar Wavelet Packet

## A WAVELET MCP MODEL

To represent the explanatory time-series in terms of wavelet packets we compute a time ordered non-decimated wavelet packet transform. This calculates wavelet packet coefficients for the  $j$  scales and  $k$  locations in the analysis. These coefficients are stacked together to form a multivariate time-series matrix. Each variable in the multivariate time-series matrix corresponds to a particular wavelet packet and quantifies how similar the time-series is to the wavelet packet at each time point.

Principal Components Analysis (PCA) is performed on the multivariate time-series matrix because of some high correlations between the wavelet packets. The 3 resulting linear combinations of wavelet packets, the principal components (pc), are uncorrelated and are such that a few usually will explain most of the variation in the explanatory time-series.

We model the response time-series in terms of the principal components (pc). We assume that the residuals of the fitted model follow the normal distribution and this is reasonable in each example we have seen. We use the routines available in S-Plus for fitting a standard multivariate linear regression.

## TEST LOCATIONS

In order to prove the validity of this approach three test sites have been selected on which to test the model. Each of the sites is an existing

wind park installation which will record the data required for result validation. The test data (response time-series) is a subset of the historical data recorded for that site and the reference data (explanatory time-series) is recorded by a meteorological mast or station at a variety of locations throughout the United Kingdom.

### Site A; Met Mast 1

For our first example we use data from a wind park installation in northwestern England as the response time series and a meteorological mast in northern Wales as the explanatory time-series. The locations are approximately 85 miles (136 km) apart and the correlation between the time-series was 0.702. A graphical representation of the time-series data is given in Figure 2.

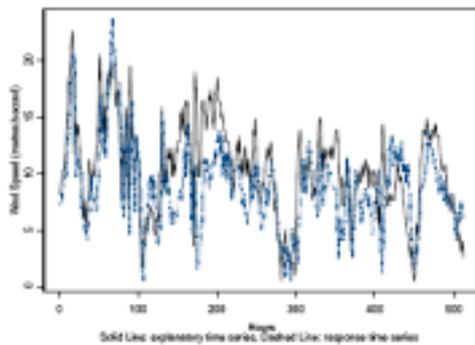


Figure 2: Time Series Plot of Site A

We perform the modelling procedure described in Section 3 to fit the model;

$$\text{Site A} \sim \text{pc 1} + \text{pc 5} + \text{direction factor} \quad (3)$$

The 1<sup>st</sup> and 5<sup>th</sup> principle components (PC1 and PC5) were chosen for inclusion in the model.

We have chosen to include the first principal component (pc1) and the fifth principal component (pc5) in the model. The principal components represent linear combinations of non-decimated wavelet packet coefficients – so (3) expresses a statistical relationship between the response time series to the coefficients through the principal components. The direction factor is added to permit the statistical model to take account of the variability in wind direction.

The factor itself simply records the direction “bin” (the process of binning is the division of 360° into 12 distinct divisions of wind direction). To help us choose which components to include in the model we used a stepwise analysis to determine which components are most useful for the prediction of wind speeds at the target site. We adjust the estimate of the wind speed at the target site by a given direction factor. This is simply a constant that is determined by the direction of the wind at the reference location.

The first principal component is a mixture of father wavelets and wavelet packets. Father wavelets average wind speeds over a given time determined by the dilation level and translation index of the father wavelet. Wavelet packets capture a wide variety of high and low frequency oscillations in the data, the exact frequency determined by the dilation and

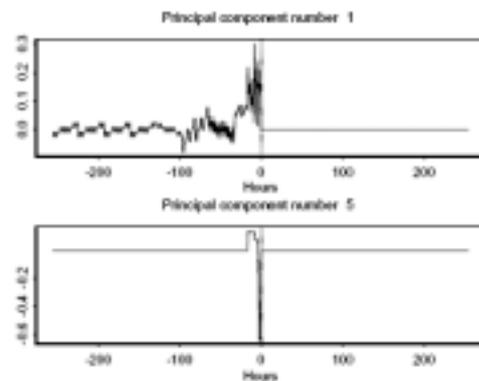


Figure 3: Principal Components from Site A Mast 1 model

translation index of the packet. The second principal component is made up of mother wavelets. Mother wavelets capture up to one oscillation over a given time determined by the dilation and translation index of the mother wavelet. The direction factor can have significant consequences in terms of swirl and wake effects for installations.

In Figure 3 we have plotted a graphical representation of the principal components. As an example, we suggest an approximate physical interpretation for principal component 5. In the first two hours there is a negative relationship between the two sites. As the sites are 136 km apart, unless the winds were extremely strong, we would not initially expect to see the same behaviour at the two sites. After two hours there is a gradual build up of the positive relationship between the two sites as wind speeds from the reference location

reach the target site. This continues up until 16 hours, by which time the weather system from Valley will have passed over St Bees Head.

### Site B; Met Mast 2

For our second example we use data from a Wind Park in southern Wales as the response time-series and data from meteorological station (Met Mast 2) as the explanatory time-series. Met Mast 2 is approximately 87 miles (136 Kms) to the North of Site B and the correlation between the time-series is 0.762. A time-series plot is given in Figure 4. We explored various model relationships and the following model was found to fit well;

$$\text{Site B} \sim \text{pc } 10 + \text{pc } 41 + \text{direction factor} \quad (4)$$

The forty-first component is similar to the first component picked out in the Site A example which represents a mixture of father wavelets and wavelet packets. The tenth principal component is a mixture of wavelet packets. These components are picking up more high frequency characteristics in the data than the components in the previous model.

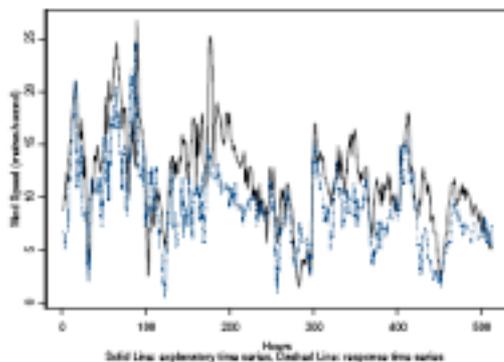


Figure 4: Time Series Plot of Site B

The direction factor is significant when the wind direction is from the Northwest, which we can again relate to site alignment, or from the Southwest, the direction of prevailing winds.

### Site C & Met Mast 3

For the third test example we take Wind Park C on the Northwest coast of England as the response time-series and Met Mast 3 on the Southwest coast of the Isle of Man as the explanatory time-series. The two

meteorological stations are approximately 45 miles (64 km) apart. The correlation between the two time-series is 0.697 and a time-series plot is shown in Figure 5. For this data we fit the model;

$$\text{Site C} \sim \text{pc } 1 + \text{pc } 4 + \text{direction factor} \quad (5)$$

The principal components are similar to those illustrated in the first example. They pick out similar wavelets and wavelet packets that represent the major characteristics of the wind speed data but with some variation that may be specific to local conditions. The direction factor is significant for winds from a Southern to a Northwest direction encompassing both prevailing wind and site alignment direction.

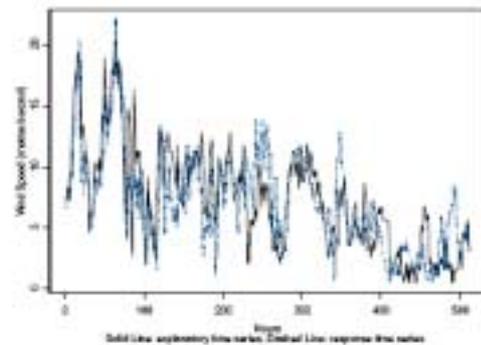


Figure 5: Time Series Plot of Site C

## PREDICTING FUTURE WIND SPEEDS

We measure the accuracy of our method by looking at the mean residual sum of squares (MRSS) error of the predictions generated by our model. The models were fitted using a training dataset of 512 observations and a consecutive data run on 512 observations was used for the prediction. To evaluate our model we compare our predictions with those generated by the simple linear regression model.

It can be seen from Table 1 that the wavelets perform significantly better over the early prediction period. The results are closer for the longer prediction period as only a small quantity of data is used to formulate the models. We believe these preliminary results suggest our wavelet method is worth investigating as an alternative method for the short-term prediction of wind speeds.

## CONCLUSIONS AND FUTURE WORK

We have proposed an alternative method for the prediction of wind speeds at a target site using wind speeds from a reference location. The preliminary results on relatively short time-series have been encouraging. Wavelet methods have been shown to provide more reliable estimates than a prediction method using simple linear regression over 21 days.

Our wavelet models also have the added bonus of often being physically interpretable. While it may be sensible to continue the study over a longer period of time it is important to remember the initial scope of the investigation; to assess wavelets ability to forecast wind speeds over a short term horizon, to satisfy trading requirements and over a longer term horizon for maintenance scheduling.

Table 1: Mean Residual Sum of Squares (MRSS) of the Predictions

Location	Forecast Horizon	Wavelets	LR
Site A & Mast 1	5 days	6.20	8.08
	10 days	5.40	6.89
	21 days	5.46	6.48
Site B & Mast 2	5 days	4.80	16.69
	10 days	4.36	12.74
	21 days	6.95	9.09
Site C & Mast 3	5 days	3.29	7.00
	10 days	3.59	6.57
	21 days	4.39	5.51

Wavelets show that they offer a significant improvement over linear regression and similar statistical approaches to wind spectra forecasting. It is intended to conduct further tests to assess their performance when compared to other leading techniques such as ARIMA, Moving Averages and Neural Networks, which have produced impressive results in recent studies (Campbell and Adamson 2004).

The authors would like to thank B9 Energy Ltd, Northern Ireland for their support of this project.

## REFERENCES

- Derrick, A. 1992. "The development of the measure-correlate-predict strategy 9 for site assessment". *Proc. 14th British wind energy association conference*, 259-265.
- Nason, G.P., Sapatinas, T. , Sawczenko, A. 2001. "Wavelet packet modelling of infant sleep state using heart rate data". *Sankhya*, Series B, Vol. 63, 199-217
- Burrus, C. S., R. A. Gopinath, & Guo, H. 1998. *Introduction to wavelets and wavelet transforms: A primer*. Upper Saddle River, NJ: Prentice Hall.
- Wickerhauser, M. V. 1994. *Adapted wavelet analysis from theory to software*. Wellesley, Massachusetts.
- Nason, G.P. & Sapatinas, T. (2002). "Wavelet packet transfer function modelling of nonstationary time series". *Statistics and Computing*, Vol. 12, 45-56.

Campbell PRJ, Adamson K, 2004. "A Comparison of Forecasting Methodologies for Wind Speed Time Series", *In Proc. Fourth International Conference on Modelling, Simulation and Optimization*, Kauai, USA, 211-218.

# MODELLING FORCES ACTING ON THE PLOUGH BODY

Adolfs Rucins, Arvids Vilde  
 Latvia University of Agriculture, Research Institute of Agricultural Machinery  
 1 Instituta Street, Ulbroka, LV-2130, Latvia  
 e-mail: [arucins@delfi.lv](mailto:arucins@delfi.lv)

## KEY WORDS

Modelling forces acting on the plough body, draft resistance, analytic correlations, optimisation of parameters.

## ABSTRACT

By using analytical correlations derived as a result of theoretical research, a computer algorithm has been worked out for simulating the functions of the plough body and the forces exerted by soil upon the operating parts, as well as its draft resistance. These correlations allow to determinate the forces acting on the plough body and its draft resistance depending on the body parameters, as well as to evaluate the impact of the physical and mechanical properties of soil upon it. They considerably influence the draft resistance of the body and, respectively, the energy capacity of ploughing and fuel consumption. The greatest influence upon the draft resistance is exerted by soil hardness, density and slip resistance along the surfaces of the operating parts. The speed increasing, the optimum inclination value of the horizontal generatrix for the minimum draft resistance decreases. In loamy soils, when the operating speed is  $1...3 \text{ m s}^{-1}$ , its optimum value is correspondingly  $50...25^\circ$ .

The draft resistance of the supporting surfaces can reach 25...30% of total plough body draft resistance or 42...54% of its share-mouldboard drafts resistance. The friction resistance constitutes 50 – 60 % of the total resistance including the resistance of the supporting surfaces (25...30%).

## INTRODUCTION

It is known from our previous investigation (Vilde 1999, 2001) that the draft resistance of ploughs depends on such soil properties as its hardness, density, friction and adhesion. However, there were no analytical correlations that would enable to determine the draft resistance of the share-mouldboard surface and the plough body, as a whole, depending on their properties.

The purpose of the investigation is to estimate the forces acting upon the surfaces of the plough body and the impact of the physical and mechanical properties of soil on its draft resistance.

## MATERIALS AND METHODS

The objects of the research are the forces acting on the plough body and its draft resistance depending on the body design parameters, as well as the physical and mechanical properties of soil. On the basis of the previous investigations (Vilde 1999) a computer algorithm has been worked out (Rucins and Vilde

2003a) for the simulation of the forces exerted by soil upon the operating (lifting and supporting) surfaces of the plough body, and the draft resistance caused by these forces (Fig. 1).

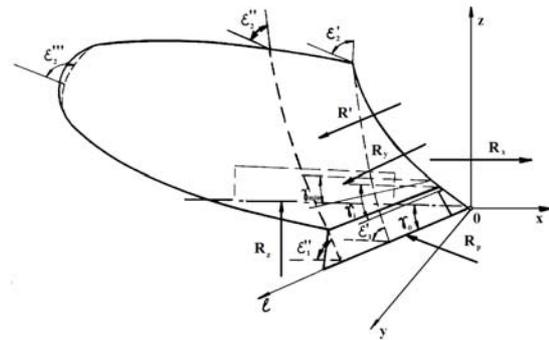


Fig.1. Scheme of the plough body, its parameters and acting forces.

According to our previous investigations (Vilde 1999), the draft resistance  $R_x$  of the plough body is determined by the share cutting resistance  $R_{Px}$ , the resistance caused by weight  $R_{Gx}$  of the strip lifted, by the inertia forces  $R_{Jx}$ , by soil adhesion  $R_{Ax}$  and by weight  $R_{Qx}$  of the plough body itself (including a part of the weight of the plough). However, the latter is not dependent on the plough parameters.

$$R_x = \sum R_{ix} = R_{Px} + R_{Gx} + R_{Jx} + R_{Ax} + R_{Qx} \quad (1)$$

The vertical reaction  $R_z$  and the lateral reaction  $R_y$  of the operating part are defined by corresponding partial reactions:

$$R_z = \sum R_{iz}; \quad R_y = \sum R_{iy} \quad (2; 3)$$

The total draft resistance  $R_x$  of the operating part is composed of the resistance of the working surface  $R'_x$  and the resistance of the supporting (lower and lateral) surfaces  $R''_x$ :

$$R_x = R'_x + R''_x = \sum R'_{ix} + f_0 (\sum R_{iz} + \sum R_{iy} + p_{Axy} S_{xy} + p_{Axz} S_{xz}) \quad (4)$$

where:  $f_0$  is the coefficient of soil friction along the working and supporting surfaces of the operating part;

$p_{Axy}$  and  $p_{Axz}$  - specific adhesion force, respectively, to the lower and the lateral supporting surfaces of the operating part;

$S_{xy}$  and  $S_{xz}$  - the surface area, respectively, of the lower and the lateral supporting surfaces of the operating part.

The friction resistance  $F_x$  is a constituent part of these reactions and their components (Rucins and Vilde 2003b), and by analogy we can write that

$$F'_x = \sum F'_{ix} = F'_{Px} + F'_{Gx} + F'_{Jx} + F'_{Ax} + F'_{Qx} =$$

$$= R'_x - R'_{xo} \quad (5)$$

$$F''_x = f_0 (R_z + R_y + p_{Axy} S_{xy} + p_{Axz} S_{xz}) = R''_x, \quad (6)$$

$$F_x = F'_x + F''_x. \quad (7)$$

The friction resistance of the share-mouldboard surface is defined as the difference between the total resistance (general value of the partial resistance) and the resistance  $R_{xo}$  in operation without friction ( $f_0 = 0$ ).

$$F_{ix} = R_{ix} - R_{ixo}; F_x = R_x - R_{xo}, \quad (8; 9)$$

The ratio  $\lambda_F$  of the friction resistance in the partial and total resistance (reaction) is determined from their correlations:

$$\lambda_{F_{ix}} = F_{ix} R_{ix}^{-1}, \lambda_{F_x} = F_x R_x^{-1}. \quad (10; 11)$$

The ratio  $\lambda_R$  of the supporting reactions in the partial and total draft resistance is determined from correlation:

$$\lambda_{R_i} = R_i R_{ix}^{-1}. \quad (12)$$

Cutting resistance  $R'_{Px}$  is proportional to soil hardness  $\rho_0$  and the share edge surface area  $\omega$ :

$$R'_{Px} = k_p \rho_0 \omega = k_p \rho_0 ib, \quad (13)$$

where:  $k_p$  is the coefficient involving the impact of the shape of the frontal surface of the ploughshare edge;  
 $i$  and  $b$  - the thickness and width of the edge.

It is evident from formula (13) that the friction of soil along the edge does not influence the cutting resistance of the edge.

At a sharp ploughshare (the rear bevel is absent)

$$R_{pz} = 0. \quad (14)$$

At a blunt (threadbare) ploughshare having rear bevel the vertical reaction  $R_{pz}$  on the hard soils can reach summary value of vertical reactions, this summary value arising from other forces acting on share-mouldboard surface (soil gravity and inertia) and weight of the body  $Q$ .

At an inclined ploughshare a lateral reaction  $R_{py}$  arises, its value being affected by the friction reaction.

$$R_{py} = k_p \rho_0 ib \operatorname{ctg} (\gamma + \varphi_0), \quad (15)$$

where:  $\gamma$  is the inclination angle of the edge towards the direction of movement (the wall of the furrow);

$\varphi_0$  - the angle of friction.

When friction is absent,  $f_0 = 0$ ,  $\varphi_0 = 0$  and

$$R_{py_0} = k_p \rho_0 ib \operatorname{ctg} \gamma. \quad (16)$$

Friction of soil along the ploughshare edge reduces the lateral pressure of the ploughshare (the pressure of the plough body against the wall of the furrow).

The resistance of the supporting surface

$$R''_{Px} = k_p \rho_0 ib f_0 \operatorname{ctg} (\gamma + \varphi_0) = F''_{Px}. \quad (17)$$

The total cutting resistance

$$R_{Px} = k_p \rho_0 ib [1 + f_0 \operatorname{ctg} (\gamma + \varphi_0)]. \quad (18)$$

The lateral cutting resistance of the knife is determined by formulae, similar to those for the cutting resistance from below. Consequently, similar to the above formulae will also be the formulae defining the impact of friction on the total resistance of the knife.

*Forces caused by the weight of the lifting soil strip:*

$$R'_{Gx} \approx q \delta g k_y r \sin^{-1} \gamma *$$

$$* \{ [(\sin \gamma \cos \varepsilon_1 + \cos^2 \gamma \sin^{-1} \gamma) e^{f_0 \sin \gamma (\varepsilon_1 - \varepsilon_2)} -$$

$$- (\sin \gamma \cos \varepsilon_2 + \cos^2 \gamma \sin^{-1} \gamma)] \cos \varepsilon_1 +$$

$$+ (\cos \varepsilon_1 e^{f_0 \sin \gamma (\varepsilon_2 - \varepsilon_1)} - \cos \varepsilon_2) (\cos \varepsilon_1 -$$

$$- f_0 \sin \varepsilon_1 \sin \gamma)^{-1} \sin \varepsilon_1 *$$

$$* [\sin \varepsilon_1 \sin \gamma + f_0 (\sin^2 \gamma \cos \varepsilon_1 + \cos^2 \gamma)] \}; \quad (19)$$

$$R_{G_y} \approx q \delta g r \sin^{-1} \gamma (\varepsilon_2 - \varepsilon_1) (\varepsilon_1 + 0.52) \operatorname{ctg} \gamma; \quad (20)$$

$$R_{G_z} \approx q \delta g r \sin^{-1} \gamma (\varepsilon_2 - \varepsilon_1); \quad (21)$$

$$R''_{Gx} = f_0 (R_{Gz} + R_{Gy}) = F''_{Gx}. \quad (22)$$

*Forces caused by the soil inertia:*

$$R'_{Jx} = q \delta v^2 k_y^{-1} \sin \gamma \{ (\sin \gamma \cos \varepsilon_1 + \cos^2 \gamma \sin^{-1} \gamma) *$$

$$* e^{f_0 \sin \gamma (\varepsilon_1 - \varepsilon_2)} - (\sin \gamma \cos \varepsilon_2 + \cos^2 \gamma \sin^{-1} \gamma) +$$

$$+ (\cos \varepsilon_1 - f_0 \sin \varepsilon_1 \sin \gamma)^{-1} e^{f_0 \sin \gamma (\varepsilon_2 - \varepsilon_1)} *$$

$$* \sin \varepsilon_1 [\sin \varepsilon_1 \sin \gamma + f_0 (\sin^2 \gamma \cos \varepsilon_1 + \cos^2 \gamma)] \}; \quad (23)$$

$$R_{Jz} = q \delta v^2 k_y^{-1} \sin \gamma \sin \varepsilon_2 e^{f_0 \sin \gamma (\varepsilon_2 - \varepsilon_1)}; \quad (24)$$

$$R_{Jy} \approx q \delta v^2 k_y^{-1} \sin \gamma \cos \gamma (1 - \cos \varepsilon_2); \quad (25)$$

$$R''_{Jz} = f_0 (R_{Jz} + R_{Jy}) = F''_{Jz}. \quad (26)$$

*Forces caused by soil adhesion:*

$$R'_{Ax} = p_A b r \sin^{-1} \gamma (e^{f_0 \sin \gamma (\varepsilon_2 - \varepsilon_1)} - 1) *$$

$$* \{ \sin \gamma \cos \varepsilon_1 + \cos^2 \gamma \sin^{-1} \gamma + (\cos \varepsilon_1 - f_0 \sin \varepsilon_1 \sin \gamma)^{-1} *$$

$$* \sin \varepsilon_1 [\sin \varepsilon_1 \sin \gamma + f_0 (\sin^2 \gamma \cos \varepsilon_1 + \cos^2 \gamma)] \}; \quad (27)$$

$$R_{Az} = 0; \quad (28)$$

$$R_{Ay} \approx 0; \quad (29)$$

$$R''_{Ax} = f_0 (p_{Axy} S_{xy} + p_{Axz} S_{xz}) = F''_{Ax}. \quad (30)$$

where:  $q$  - the cross section area of the strip to be lifted;  
 $\delta$  - the density of soil;  
 $k_y$  - the soil compaction coefficient in front of the operating part;  
 $f_0$  - the soil friction coefficient against the surface of the operating element;

- $v$  - the speed of the movement of the plough body;
- $p_A$  - the specific force of soil adhesion to the operating surface;
- $b$  - the surface width of the soil strip;
- $\varepsilon_1$  and  $\varepsilon_2$  are correspondingly the initial and the final angles of the lifting (share-mouldboard) surface;
- $g$  - acceleration caused by gravity ( $g = 9.81$ ).

The soil friction coefficient and the specific force of soil adhesion are not constant values. Their values decrease with the increase in speed (Vilde 2003b). This is considered in calculations.

The resistance of the supporting surfaces of the plough body depends on the values of the reacting forces. Yet their value is dependent, in many respects, on the manner of unification and perfection of the hydraulically mounted implements of the tractor. The vertical reaction of the plough with modern tractors having power regulation is transferred to the body of the tractor, and it affects the plough resistance to a considerably lesser degree. There are also solutions for the reduction of the lateral reaction. In such a way, the dominating component of the draft resistance of the plough body is the resistance of its share-mouldboard surface, to the research of which the present work is mainly devoted.

## RESULTS

The presented work discusses, as an example, the research results of the forces acting on the plough body and the draft resistance caused by the share-mouldboard surface of the plough body at various angles  $\gamma$  of the horizontal generatrices depending on the speed of operation when ploughing loamy soils that predominate in Latvia.

The calculations were carried out with the computer according to the foregoing formulae.

The following values of the basic factors were taken into consideration, which affect the resistance of the share-mouldboard surface and the plough body.

*Parameters of the plough body:*

Thickness of the share blade and knife  $i = 0.004 \text{ m}$

The initial angle of the lifting strip of soil  $\varepsilon_1 = 30^\circ$

The final angle of the lifting strip of soil  $\varepsilon_2 = 100^\circ$

The angle between the horizontal generatrix of the operating surface and the vertical longitudinal plane  $\gamma = 15^\circ \dots 90^\circ$

The radius of the curvature of the lifting surface  $r = 0.5 \text{ m}$

The area of the lower supporting surface  $S_{xy} = 0.0157 \text{ m}^2$

The area of the lateral supporting surface  $S_{xz} = 0.068 \text{ m}^2$

The weight of the plough body  $Q = 200 \text{ kg}$

*Physical and mechanical properties of soil:*

The hardness of soil  $\rho = 4.1 \text{ MPa}$

The density of soil  $\delta = 1600 \text{ kg m}^{-3}$

The coefficient of soil friction against the surface of the operating element  $f_0 = 0.4$

The adhesion force  $p_{A0} = 2.5 \text{ kPa}$

*The mode and status of work:*

The ploughing depth  $a = 0.20 \text{ m}$

The cross section area of the lifted soil strip  $q = 0.07 \text{ m}^2$

The soil compaction coefficient in front of the operating part  $k_y = 1.1$

The working speed  $v = 1 \dots 5 \text{ m s}^{-1}$ .

The inclination angle  $\gamma$  of the horizontal generatrix of the real share-mouldboard surfaces of plough bodies lies between  $26^\circ \dots 50^\circ$ . Steeper surfaces ( $\gamma > 50^\circ$ ) refer to the slanting blades of bulldozers.

The calculation results of the draft resistance of the lifting surface and its components are presented in Fig. 2 – 5, the reacting forces on the supporting surfaces – in Fig. 6 – 8, the draft resistances of the share-mouldboard and supporting surfaces - in Fig. 9 – 10 and the total draft resistance of the plough body – in Fig. 11.

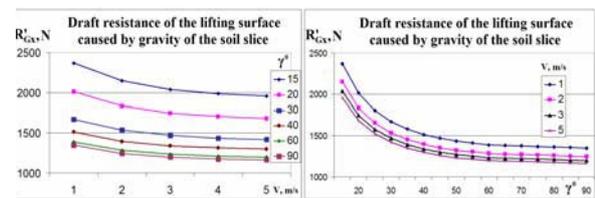


Fig. 2. Draft resistance of the lifting surface caused by the gravity of the soil slice depending on speed  $v$  and the inclination angle  $\gamma$  of the horizontal generatrix.

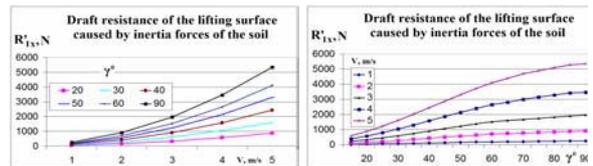


Fig. 3. Draft resistance of the lifting surface caused by the soil inertia forces of the soil slice depending on speed  $v$  and the inclination angle  $\gamma$  of the horizontal generatrix.

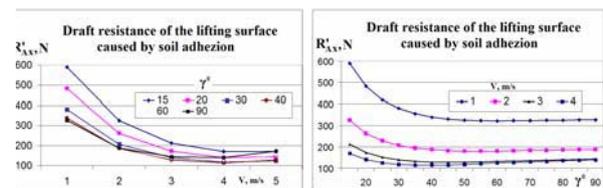
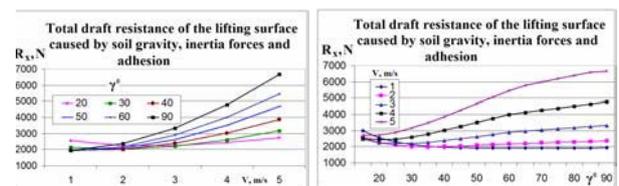


Fig. 4. Draft resistance of the lifting surface caused by soil adhesion depending on speed  $v$  and the inclination angle  $\gamma$  of the horizontal generatrix.

Fig. 5. Total draft resistance of the lifting surface caused



by soil gravity, inertia forces and adhesion depending on speed  $v$  and the inclination angle  $\gamma$  of the horizontal generatrix.

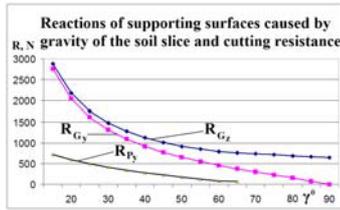


Fig. 6. Reactions of the lower and lateral supporting surfaces caused by gravity of the soil slice and share cutting resistance depending on the inclination angle  $\gamma$  of the horizontal generatrix.

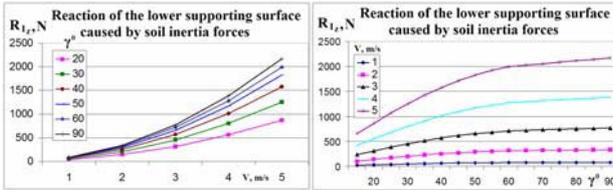


Fig. 7. Reaction of the lower supporting surface caused by soil inertia forces depending on speed  $v$  and the inclination angle  $\gamma$  of the horizontal generatrix.

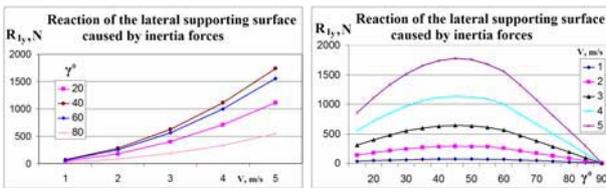


Fig. 8. Reaction of the lateral supporting surfaces caused by soil inertia forces depending on speed  $v$  and the inclination angle  $\gamma$  of the horizontal generatrix.

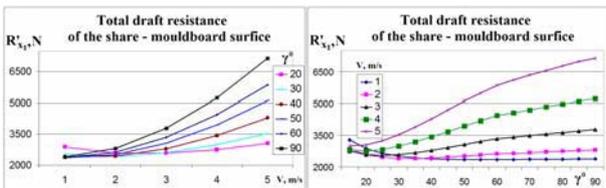
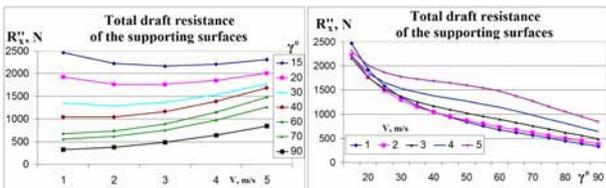


Fig. 9. Total draft resistance of the share-mouldboard surface caused by soil gravity, inertia forces, adhesion and share cutting resistance depending on speed  $v$  and the inclination angle  $\gamma$  of the horizontal generatrix.

Fig. 10. Total draft resistance of the supporting surfaces depending on speed  $v$  and the inclination angle  $\gamma$  of the horizontal generatrix.



depending on speed  $v$  and the inclination angle  $\gamma$  of the horizontal generatrix.

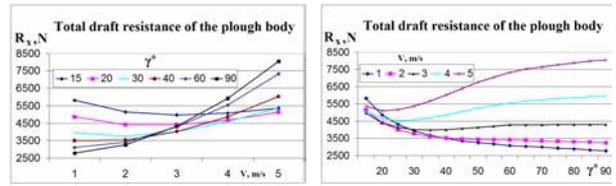


Fig. 11. Total draft resistance of the plough body depending on speed  $v$  and the inclination angle  $\gamma$  of the horizontal generatrix.

The material of the calculations presents the values and correlations of the changes in the forces acting on the share-mouldboard and the supporting surfaces, the draft resistance of the share-mouldboard, and the supporting surfaces, as well as the total resistance of the plough body and its components under working conditions depending on the working speed  $v$  and the inclination angle  $\gamma$  of the horizontal generatrix.

It follows from the figures that the values of resistances caused by the weight and soil adhesion decrease with the increase in the operation speed (Fig. 2 and 4). This can be explained by the reduction of the friction coefficient and the specific adhesion force of soil while the speed of its slipping along the share-mouldboard surface increases. The resistance caused by the soil inertia forces increases when speed increases (Fig. 3), and at speeds over  $3 \dots 4 \text{ m s}^{-1}$  these inertia forces start dominating over all the other components. When speed increases (up to the speed of  $2 \dots 2.5 \text{ m s}^{-1}$ ), the summary draft resistance of the share-mouldboard surface increases insignificantly, then grows faster (Fig. 5 and 9). At a steeper share-mouldboard surface (at great values of angle  $\gamma$ ) this growth is more remarkable and intense.

In wet loamy soils there may be cases (at quite a flat share-mouldboard surface) when the draft resistance does not increase but even decreases whereas speed increases (within the range of  $1 \dots 2 \text{ m s}^{-1}$ ). Such a phenomenon may occur when the decrease in resistance due to the lower friction coefficient and specific soil adhesion proceeds more intensely than the growth in the resistance caused by the soil inertia forces within the given range of speeds.

When the inclination of the generatrix is increased (angle  $\gamma$ ), resistances because of the soil weight and adhesion fall but the resistance due to the inertia forces increases, particularly in operation at higher speeds. The decrease of the first ones can be explained by the fact that at a steeper share-mouldboard surface its length decreases and because of this there is a decrease in the mass of soil slipping along it. Decreasing the area of its surface leads to a lower resistance due to soil adhesion. As a result, the total draft resistance of the share-mouldboard surface shows a marked minimum, which at a greater operating speed moves towards lower inclination values of the horizontal generatrix. Thus, increasing the speed from  $1$  to  $3 \text{ m s}^{-1}$ , the optimum

value of angle  $\gamma$  of the share-mouldboard surface decreases from  $50^0$  to  $25^0$  (Fig. 9).

From the presented example it is evident (Fig. 10 and 11) that the draft resistance of the supporting surfaces is considerable. It can reach 25...30% of the total plough body draft resistance, or 42...54% of its share-mouldboard draft resistance (Fig. 10 and 9).

The impact of the soil-metal friction upon the plough body draft resistance is significant too. It may reach 50...60% of the total draft resistance including the resistance of the supporting surfaces (25...30%).

In such way, the deduced analytical correlations and the developed computer algorithm allow simulation of soil coercion upon the share-mouldboard surface of the plough body, taking into consideration its draft resistance in determining the optimum parameters (the inclination of the horizontal generatrix) at minimum resistance.

## CONCLUSIONS

1. The deduced analytical correlations and the developed computer algorithm allow simulation of the soil coercion forces upon the operating surfaces of the plough body, determination of the draft resistance and the optimal values of parameters.

2. Presentation of the plough body draft resistance as the sum of components – the cutting resistance of the strip, the resistance caused by its weight, the soil inertia forces and adhesion - allows analysing the forces acting upon the share-mouldboard surface, finding out the character of their changes depending on speed and the parameters of the surface, and assessment of their ratio in the total resistance.

3. Increase in the inclination of the horizontal generatrix leads to a decrease in the draft resistance caused by the weight and adhesion of soil but it increases the resistance caused by inertia forces, particularly, when the speed increases. The inclination of the generatrix (the edge of the share) does not affect the cutting resistance of the strip.

4. In loamy soils, when the speed grows from 1 to 3 m s<sup>-1</sup>, the optimum value of the inclination angle between the horizontal generatrix of the share-mouldboard surface and the wall of the furrow decreases from  $50^0$  to  $25^0$ .

5. The draft resistance of the supporting surfaces is considerable. It can reach 25...30% of the total plough body draft resistance, or 42...54% of its share-mouldboard draft resistance.

6. The impact of the soil-metal friction upon the draft resistance of the plough body is significant too. It may reach 50...60% of total draft resistance including the resistances of the supporting surfaces (25...30%).

## References

Vilde A. 1999. "Dynamics of the soil tillage machine operating parts and their elements". In: *Proceedings*

*of the Latvia University of Agriculture*, Vol.1 (295). Jelgava, Latvia, 36-44.

Vilde A. 2001. "Physical and mechanical properties of soil affecting energy capacity of its tillage". In: *Proceedings of the 1<sup>st</sup> International Conference of BSB of ISTRO "Modern ways of soil tillage and assessment of soil compaction and seedbed quality"* – 21-24 August 2001. EAU, Tartu, Estonia, 97-106.

Rucins A., Vilde A. 2003a. "Mathematical modelling of the operation of plough bodies mould-boards to determine their draft resistance and optimal parameters". *Research for rural development 2003. International scientific conference proceedings Jelgava, Latvia 21 -24 May, 2003*. Jelgava, Latvia University of agriculture. 64-67.

Rucins A., Vilde A. 2003b. "Impact of soil-metal friction on the draft resistance of ploughs". *Research for rural development 2003. International scientific conference proceedings Jelgava, Latvia 21 -24 May, 2003*. Jelgava, Latvia University of agriculture, 61-63.

## AUTHOR BIOGRAPHIES



**ADOLFS A. RUCINS** was born in 1962 in Riga, Latvia. After finishing Latvia Industrial Polytechnic school he entered the Latvia Technical University, where he studied Machinery and obtained his engineer degree in 1989. Subsequently he

obtained degree Master of Technical Science in 2002. He worked 3 years as lector in Latvia Technical University. Since 1992 up to this time he is researcher in Latvia Research Institute of Agricultural Machinery.

Research branches: technology, machines and units for soil tillage; combined machines for tillage, fertilizing and grain sowing. Now he takes part in the research works in Precision Agriculture using the GPS, as well as simulation of forces acting on the plough body in order to determine its draft resistance and optimal parameters. He has also written numerous publications.

He enjoys skiing and fishing. With his wife Sarma, he has san.

His e-mail address is [arucins@delfi.lv](mailto:arucins@delfi.lv)



**ARVIDS A. VILDE** was born in 1929 in Bauska, Latvia, in a farmer's family. After finishing a secondary school he entered the Latvia Academy of Agriculture, where he studied Agricultural Machinery and obtained the degree of an

engineer in 1954. Subsequently he obtained the following degrees: Candidate of Technical Sciences (SU) in 1965, Doctor of Technical Sciences (SU) in 1986, Doctor Habil. Sc. Ing. in 1992. He worked for 5 years as the chief engineer in the field of agricultural machinery and production. Since 1960 up to this time he is engaged in the leading research groups in Latvia Research Institute of Agricultural Machinery. The research branches: soil dynamics in tillage, including the dynamic properties of soil; technology, machines and units for soil tillage and sugar beet growing, their

rationalization and perfection; the use of big high-speed tractors and machines; energy requirements for field crop production and ways of it reduction; normative requirements for tractors and machines, their working load and fuel consumption. Now he is leading the research in Precision Agriculture using the GPS and in the plant spacing simulation by sowing seeds at exact intervals, as well as in the simulation of forces acting on the plough body in order to determine its draft resistance and optimal parameters.

A Vilde has received several medals and diplomas at the Exhibition of Economic Achievement in Moscow. In 1985 he received the Latvian State Prize. He was named a Merited Inventor of Latvia in 1990, International Man of the Year for 2000-01 and Latvia State Emeritus Scientist 2001. He is an expert of promotion councils and a publicist who has written more than 700 publications including eighteen monographs.

He enjoys orchards and stenography. With his late wife Velta, he has four children and eleven grandchildren.

His e-mail address is [vilde@delfi.lv](mailto:vilde@delfi.lv)

# MODELLING PLANT SPACING AND YIELDS OF CROPS BY SOWING SEEDS AT EXACT INTERVALS

Arvids Vilde and Aivars Cesnieks  
Latvia University of Agriculture, Research Institute of Agricultural Machinery  
1 Instituta Street, Ulbroka, LV-2130, Latvia  
E-mail: [vilde@delfi.lv](mailto:vilde@delfi.lv)

## KEYWORDS

Modelling plant spacing, plant density, yields, mathematical coherences.

## ABSTRACT

A study of probable plant spacing in field crop and vegetable plantations and their corresponding yields by sowing seeds at exact intervals and growing without thinning have been carried out. By modelling relationships of the plant distribution, spacing density and yields mathematical coherences are obtained. It is stated that the plant spacing density is a function of the seed germinating power in the field. The lower is the germinating power, the higher is the irregularity of the seedlings, i.e. the number of the longer spacing increases. The obtained coherences allow to prognosticate the irregularity of the seedlings depending on the expected seed germinating power and to specify the standard quantity of seeds per hectare. A formula is obtained for the estimation of field crop and vegetable yields depending on plant density.

## INTRODUCTION

To obtain high and qualitative yields in growing field crops and vegetables, it is of great importance to achieve the required density and their uniform spacing. This problem has become particularly urgent due to the transition from growing crops (sugar and fodder beet) and vegetables (cucumbers, red beet) with their thinning to growing them without thinning, by sowing the seeds at extreme intervals.

The purpose of this study is to clarify mathematical coherences between the plant spacing density, their distribution and the yields to be reached.

## APPROACH

Theoretical and experimental research has been carried out to obtain the relationships of plant density, their spacing and yields of crops obtained. Theories of probability and mathematical statistics are used in the investigations. The results of theoretical research carried out to clear up the relationships of plant density and their spacing has been affirmed with experiment data.

## RESULTS

### Plant spacing relationships

It is found out in the previous investigations that at a

great sowing ratio (more than 20 seeds per metre of the row) the plant spacing in the row approaches the binomial distribution, but at thin sowing of the seeds at certain intervals it forms a series of normal spacing with a decreasing mode frequency. A similar picture is observed in the later investigations too (Vilde and Cesnieks 1999).

To reach the desired plants distribution density  $N_{opt}$ , the number of the seeds  $N_s$  to be sown per unit of the area (ha) is determined when plant distribution density is divided by the germinating power  $q$  of the seeds:

$$N_s = N_{opt} q^{-1} \quad (1)$$

The number of the seeds  $n_s$  to be sown out per metre of the row at the distance  $b$  between the rows:

$$n_s = 10^4 N_s b \quad (2)$$

and the distance  $l_s$  between the seeds sown:

$$l_s = n_s^{-1} \quad (3)$$

The plant spacing uniformity in the plantations can be characterised by the frequency of intervals (distances) between the plants.

If each seed has sprouted, the distance between the plants corresponds to the distance between the seeds sown  $l_s$ . But there is a probability that one, two, three, etc. seeds that follow each other have not sprouted. In this case the intervals between the plants will increase correspondingly two, three etc. times (see Fig.1).

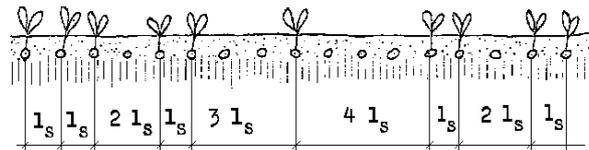


Fig.1. Spacing of seeds and seedlings in the row.  
 $l_s$  - the interval between the seeds

By applying: the probability theory and the methods of mathematical statistics a mathematical coherence is obtained to be used to determine the distribution of intervals between the plants and their spacing. It is found that the intervals between the seedlings and their frequency are dependent on the distance between the seeds sown at the interval  $l_s$  and their germinating power in the field. With the decrease of the germinating power the irregularity of the sprout spacing increases, that is, the number of the greater intervals increases.

The expected spacing frequency  $p(l_s)$  between the

seedlings corresponding to the distance between the seeds  $l_s$  is equal to the germinating power of the seeds in the field  $q$ . For double distance  $2 l_s$  the distribution of the corresponding intervals  $p(2 l_s)$  will correspond to  $(1-q)q$ , and so on.

As a result, a series of formulae is obtained for the calculation of the interval frequency between the crop seedlings depending on their germinating power  $q$  in the field:

$$\begin{aligned}
 p(l_s) &= q \\
 p(2 l_s) &= q(1-q) \\
 p(3 l_s) &= q(1-q)^2 \\
 &\dots\dots\dots \\
 p(n l_s) &= q(1-q)^{n-1}
 \end{aligned}
 \tag{4}$$

In the set of formulae (4) the given coherence is in force if the seeds are sown out one by one, the deviation of the seeds and the sprouts from the pre-set sowing interval will not exceed its half and in the given part of the field the germinating power of the seeds is approximately the same.

The interval frequency between the seedlings at various germinating power is presented in the Table 1 and the diagram (Figure 2).

It is obvious from Table 1 and Figure 2 that the decrease in the germinating power of the seeds is followed by increased irregularity of spacings between the seedlings. Therefore, in order to obtain uniform plant distribution, not only exact spacing of the seeds is important but also their high germinating power in the field, which can be achieved by using high-quality seeds and ensuring optimum conditions for their germination (qualitative soil preparation, sowing at optimum depth with shares of a correct shape, protection against diseases and pests etc.).

### Correlations of plant distribution on non-homogeneous fields.

Experience shows that the large fields formed as a result of joining smaller fields together often have non-uniform physical and mechanical soil composition, which considerably affects the germinating power of the seeds on the field and the density of sprouts. This circumstance should be taken into particular consideration when the plants are grown without thinning by sowing them out in rows at extreme intervals. At such a technology, areas (spots) often appear on the fields with insufficient plant density, which has an adverse effect on the yield obtained. In this connection theoretical studies have been carried out and mathematical coherences are derived on the plant distribution on such non-homogeneous fields.

On non-homogeneous fields the medium interval frequency between the plants is determined on areas with different germinating power of the seeds on the field.

Table 1. The interval frequency between the seedlings, %

The interval		The germinating power in the field, %					
General case	Particular case, cm	100	80	60	40	20	10
$1 l_s$	16.7	100	80	60	40	20	10
$2 l_s$	33.4	0	16	24	24	16	9.0
$3 l_s$	50.1		3.6	14	14	13	8.1
$4 l_s$	66.8		0.2	3.8	8.6	10	7.3
$5 l_s$	83.5			1.5	5.2	8.2	6.6
$6 l_s$	100.2			0.6	3.1	6.6	5.9
$7 l_s$	116.9			0.2	1.9	5.2	5.3
$8 l_s$	133.6			0.1	1.1	4.2	4.8
$9 l_s$	150.3				0.7	3.4	4.3
$10 l_s$	167				0.4	2.7	3.9
$11 l_s$	183.7				0.2	2.1	3.5
$12 l_s$	200.4				0.1	1.7	3.1
$13 l_s$	217.1					1.4	2.8
$14 l_s$	233.8					1.1	2.5
$15 l_s$	250.5					0.1	2.3
$16 l_s$	267.2						2.1
$17 l_s$	283.9						1.9
$18 l_s$	300.6						1.7
$19 l_s$	317.3						1.5
$20 l_s$	334						1.4

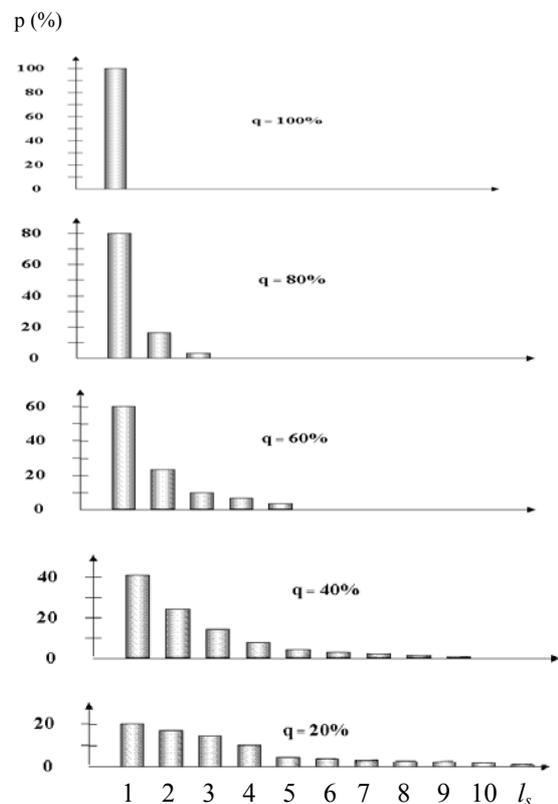


Fig. 2. Frequency  $p$  (in %) of intervals between the seedlings for different germinating power  $q$  of the seeds in the field.

$l_s$  - the interval between the sowed seeds.

The expected interval frequency between the plants  $p(l_s)$  corresponding to the interval between the seeds  $l_s$  is dependent on the interval frequency on these areas:

$$p(l_s) = (c_1 q_1 + c_2 q_2 + \dots + c_m q_m), \tag{5}$$

where  $q_1, q_2 \dots q_m$  - the germinating power of seeds on the corresponding area of the field;  
 $c_1, c_2 \dots c_m$  - the ratio of the area (spot) in the total area of the field.

The frequency corresponding to a double distance  $2 l_s$ :

$$p(2l_s) = [c_1 q_1(1-q_1) + c_2 q_2(1-q_2) + \dots + c_m q_m(1-q_m)] \quad (6)$$

A series of formulae is obtained in a similar way for the calculation of frequencies for the intervals that are three, four and more times larger.

### Yields.

The plant distribution density and its non-uniformity affect the expected yield of the crops (beets) (Figure 3).

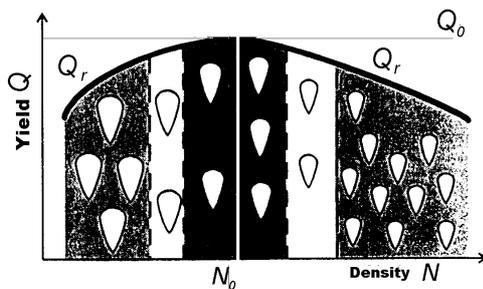


Fig.3. Variations in the crop productivity  $Q$  depending on plants distribution density  $N$ .

- desired density
- acceptable density
- insufficient density

A formula is obtained for the estimation of yields  $Q$  that are dependent on the plant distribution density  $N$ :

$$Q = Q_0 e^{-c(N_0 - N)^2} \quad (7)$$

and the fall in the yield  $Q_r$  is determined by the deviation from the optimum density  $N_0$  and the corresponding yield  $Q_0$ :

$$Q_r = Q_0 [1 - e^{-c(N_0 - N)^2}] \quad (8)$$

where  $c$  – a coefficient that characterises the rate of the yield decrease due the deviation of the plant distribution density from their optimum. The value of coefficient  $c$  depends mainly on soil fertility.

For sugar beet:

$N_0 = 80\ 000 \dots 90\ 000$  plants at the harvesting time;

$c = 2 \cdot 10^{-10} \dots 3 \cdot 10^{-10}$  at the density of 40...80 thousand plants per hectare;

$c = 1.1 \cdot 10^{-10} \dots 1.4 \cdot 10^{-10}$  at the density of 80...120 thousand

In determining the productivity of plants when their density varies from one area of the field to another, one cannot be guided by the average density indices. Their productivity should be evaluated for each area of the field having a particular plant distribution density and the calculated average crop productivity  $Q_{av}$ :

$$Q_{av} = \sum c_i Q_i \quad (9)$$

where:  $Q_i$  - the yield of crops on an area of the field;  
 $c_i$  - the ratio of the area (spot) in the total area of the field.

### Data of experimental studies.

As an example, the germinating power of sugar beet, their spacing, beet and sugar yields of more than 20 various sorts were studied for nine years. The results show that, when the seeds are sown at extreme distances (16-20cm) to avoid plant thinning, the field germination by years, as well as of separate sorts or seed batches vary within a very wide range. The average field germination is 40-70% but under bad conditions it fell to 20%, while under favourable conditions it reached 85% and, in individual cases, even 95%.

At the corresponding germinating power the plant interval distribution is, on the average, adequate to that calculated according to formulas (4). The lower is the seed germinating power, the less uniform is the plant spacing (the frequency of the longer intervals is greater). For instance, when the field germination is 20%, some plant intervals may surpass 2 metres.

Increasing the sowing rates does not improve the uniformity of plant spacing. It only decreases the frequency of the long intervals with a simultaneous increase in the frequency of the short intervals.

The fall in the sugar beet yield at low seed germination in the field is caused not only by a decreased plant distribution density but also by their lesser uniformity and weeds on the area not covered by the plants.

### CONCLUSIONS

1. By modelling of seed germination process mathematical coherences are obtained for the determination of plant spacing density, their distribution as well as yields. They enable to prognosticate plant spacing irregularity in the rows depending on the expected seed germinating power and to specify their sowing ratios.
2. Plant spacing frequency is a function of the seed germinating power in the field. The lower is the germinating power of seeds, the lower is the regularity of sprouts, i.e., the number of the longer intervals increases.
3. The desired plant distribution density and yield is ensured by quality soil tillage, precise sowing and embedding of high-quality seeds, good plant protection from pests and diseases.

### REFERENCES

Vilde, A., Cesnieks A. 1999. "Plant Spacing Relationships and Yields in Growing Sugar Beet without Thinning". *Trends in Agricultural Engineering TAE 1999*. Czech University of Agriculture. Prague, 309-313.

## AUTHOR BIOGRAPHIES



**ARVIDS A. VILDE** was born in 1929 in Bauska, Latvia, in a farmer's family. After finishing a secondary school he entered the Latvia Academy of Agriculture, where he studied Agricultural Machinery and obtained the degree of an engineer in 1954. Subsequently he obtained the following degrees: Candidate of Technical Sciences (SU) in 1965, Doctor of Technical Sciences (SU) in 1986, Doctor Habil. Sc. Ing. in 1992. He worked for 5 years as the chief engineer in the field of agricultural machinery and production. Since 1960 up to this time he is engaged in the leading research groups in Latvia Research Institute of Agricultural Machinery. The research branches: soil dynamics in tillage, including the dynamic properties of soil; technology, machines and units for soil tillage and sugar beet growing, their rationalization and perfection; the use of big high-speed tractors and machines; energy requirements for field crop production and ways of its reduction; normative requirements for tractors and machines, their working load and fuel consumption. Now he is leading the research in Precision Agriculture using the GPS and in the plant spacing simulation by sowing seeds at exact intervals, as well as in the simulation of forces acting on the plough body in order to determine its draft resistance and optimal parameters. A. Vilde has received several medals and diplomas at the Exhibition of Economic Achievement in Moscow. In 1985 he received the Latvian State Prize. He was named a Merited Inventor of Latvia in 1990, International Man of the Year for 2000-01 and Latvia State Emeritus Scientist 2001. He is an expert of promotion councils and a publicist who has written more than 700 publications including eighteen monographs. He enjoys orchards and stenography. With his late wife Velta, he has four children and eleven grandchildren. His e-mail address is [vilde@delfi.lv](mailto:vilde@delfi.lv)



**AIVARS H. CESNIEKS** was born in 1938 in Jelgava, Latvia in a farmer's family. After finishing a secondary school he entered the Latvia Academy of Agriculture, where he studied Agricultural Machinery and obtained the degree of an engineer in 1966. Subsequently he obtained the following degrees: Candidate of Technical Science (SU) in 1983, Doctor Sc. Ing. in 1992. He worked for 3 years as an engineer-technologist in repair shop. From 1969 till 2002 he has been working in Latvia Research Institute of Agricultural Machinery as a leading researcher and department leader. Now he is a farmer in Dobeles District, simultaneous taking part in the research activities of the Institute. The research branches: soil tillage technology, machines and units, their rationalization and perfection; the use of big high-speed tractors and machines; technologies and machinery for field crop growing, grain drying and conditioning, energy requirements for the field crop production and ways of its reduction, simulation plant spacing by sowing seeds at exact intervals. A Cesnieks has received medals and diplomas at the Exhibition of Economic Achievement in Moscow. In 1985 he received the Latvian State Prize. He has also written numerous publications. He enjoys bee-keeping and justice. With his wife Ilga he has two sons and two grandchildren.

# THE MODEL FOR SOFTWARE QUALITY MEASUREMENT, USING THE “GENETIC” FEATURE OF THE SOFTWARE

Jekaterina Kokina  
Department of Transport Electronics and Telematics  
Riga Technical University  
LV-1019, Lomonosova 1,  
e-mail: jkokina@yahoo.com

**KEYWORD:** *Software quality. “Genetic” feature. Mathematical model.*

## ABSTRACT

The author of this paper believe, that software system by itself is a complex object, which could be described via the Linear Mixed Model ANOVA. It includes different parameters, which could have fixed (non-random) or stochastic nature. There is produced an idea about software quality prediction by defect number using “genetic” feature of the software (Kokina 2003). It is believed that the term of “genetic” feature is an initial software feature of the software affecting the software from the moment it was initially developed. There are presented a mathematical model for “genetic” quality of software evaluation by Best Linear Prediction (BLP) method and some results of the experiment using simulation of this mathematical model in MATLAB here (Kokina and Petersons 2004).

Finally it is presented an idea of real complex software system simulation and a reference to the produced mathematical model for evaluation of the real software “genetic” feature in this paper. Currently the author is working at this idea and software model simulation.

This paper is a report of the current stage of the exploration work in the field of the software quality evaluation using the term of the “genetic” feature and simulation of produced model and the real application system. The previous report was presented in (Kokina 2004).

## INTRODUCTION

The narrowest sense of software product quality is commonly recognized as lack of “bugs” in the product. Software reliability and quality fundamentals were discussed in one on the previous publications (Kokina 2003). The software quality (SQ) could be determined as defect rate or number of defects (NOD) per unit of source code, for example, per million lines or per function point, or other units. Despite developers’ attempts at checking every detail, errors are made during development and very often cause end-release software failures. Some errors could be found in the software even after very long and good testing.

The author of this paper supposed that software „quality”, measured in NOD, is in high dependence of software birth conditions: main system requirements and system complexity, human factor of development

team - number of developers and testers, their skills, development place and tools and so on. This set of factors stipulates for the “genetic” feature of program, which characterizes initial quality of the system. Then it makes sense to make the assumption, that more trained and more equipped fathers using the best development tools develop more qualitative product, i.e. product with the minimum NOD.

The aim of the exploration work is to predict and evaluate software quality using a term of “genetic” feature of the software. We present Software Genetic evaluation model, based upon Best Linear Prediction (BLP) method. This method was proposed by Henderson and was used in biology, i.e. biometrics, originally and then in many other fields.

The target of this exploration work is try to predict and estimate the software “genetic” feature, making the analysis of a potential way to get better software reliability and quality.

## THE PROBLEM DESCRIPTION

The software system under testing itself is a complex object, which may be described by a set of different parameters, which may have a fixed (non-random) or stochastic nature.

Suppose the software under testing at the time moment  $t$  is characterized by the NOD  $y(t)$ : the less is the NOD, the higher SQ is (Kokina 2003). Further testing allows to reveal program errors, which number become less and less during the testing process

The number of fixed and random factors affects number of defects raised during testing. The number of fixed and random factors affects NOD raised during testing. The fixed factors for instance are financial resources for development, programming language specificity of the development sphere, development deadlines and other. The random factors are software quality and the noise, representing the not considered factors

The stochastic SQ is like the „genetic” feature, affecting the software from the moment it was initially developed. We can see, that it is very difficult to generally determine, what the „genetic software quality“ is, but we strongly believe, that this parameter must be taken into account in the planning and design of the software.

## MATHEMATICAL MODEL

The distribution of time relation is indirect to SQ. We suppose the tie between the controlled and measurable parameter  $\mathbf{y}(\mathbf{t})$ , the NOD found during the testing process in time moment  $\mathbf{t}$ , and uncontrolled random parameter  $\mathbf{s}$ , which defines the “genetic” feature of the software in our case.

This relation may be expressed via the linear mixed ANOVA model (Henderson 1984):

$$\mathbf{y} = \mathbf{X} \mathbf{h} + \mathbf{Z} \mathbf{s} + \mathbf{e} \quad (1)$$

where:

$\mathbf{y}$  - the array of controlled parameters (NOD) with the dimension  $\mathbf{n} \times \mathbf{1}$ ;

$\mathbf{h}$  - the array of fixed parameters with the dimension  $\mathbf{q} \times \mathbf{1}$ ;

$\mathbf{X}$  - incidence matrix, consisting from  $\mathbf{1}$  and  $\mathbf{0}$ , dimension  $\mathbf{n} \times \mathbf{q}$ ; specifies which components of  $\mathbf{h}$  is taken into account;

$\mathbf{s}$  - the array of uncontrolled random parameters with the dimension  $\mathbf{r} \times \mathbf{1}$ , having  $\mathbf{m}_s$  expectation for BLP case and zero expectation for BLUP and the variance  $\mathbf{G} \times \sigma_s$ .

$\mathbf{Z}$  - incidence matrix, consisting from  $\mathbf{1}$  and  $\mathbf{0}$ , dimension  $\mathbf{n} \times \mathbf{r}$ ; specifies which components of  $\mathbf{s}$  is taken into account;

$\mathbf{e}$  - the noise, having zero expectation and the variance  $\mathbf{R} \times \sigma_e$ .

Our task is to predict the numerical values of  $\mathbf{s}$ , knowing the values of  $\mathbf{y}$ . This is the reverse task to the classical regression problem.

It is known from the literature (Henderson 1984), that the Mixed Model Equations (MME), based on BLP and BLUP techniques should be used to calculate the values of  $\mathbf{s}$ . If we assume, that there is no correlation between  $\mathbf{s}$  and  $\mathbf{e}$ , than MME will look as follows:

$$\begin{bmatrix} \mathbf{X}^T \mathbf{X} & \mathbf{X}^T \mathbf{Z}_1 \\ \mathbf{Z}_1^T \mathbf{X} & \mathbf{Z}_1^T \mathbf{Z}_1 + \mathbf{I} \gamma \end{bmatrix} \begin{bmatrix} \mathbf{h} \\ \mathbf{s} \end{bmatrix} = \begin{bmatrix} \mathbf{X}^T \mathbf{y} \\ \mathbf{Z}_1^T \mathbf{y} \end{bmatrix} \quad (2)$$

where  $\gamma = \sigma_e^2 / \sigma_s^2$ ,

SQ values  $\mathbf{s}$  for BLP technique could be derived from the equation below:

$$\mathbf{s} = \mathbf{m}(\mathbf{s}) + (\mathbf{Z}^T \mathbf{Z} + \mathbf{I} \times \gamma)^{-1} \mathbf{Z}^T (\mathbf{y} - \mathbf{X} \mathbf{h}) \quad (3)$$

for BLUP technique

$$\mathbf{s} = (\mathbf{Z}^T \mathbf{Z} + \mathbf{I} \times \gamma)^{-1} \mathbf{Z}^T (\mathbf{y} - \mathbf{X} \mathbf{h}) \quad (4)$$

where

$$\mathbf{h} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y} \quad (5)$$

## RESULTS OF THE MATHEMATICAL MODEL SIMULATION

It was decided to approve stated above theoretical theses as applied to the problem of program «genetic» parameters definition (Kokina and Petersons 2004). We put some experiments with known beforehand results, which should give a model. Besides of that it was revealed the rate of the algorithm convergence to the known results.

During the experiment we assume, that the  $\mathbf{y}$  array consists of random values with normal distribution, which determines NOD in every of  $\mathbf{t}_i$  error revisions. The mean value of  $\mathbf{y}(\mathbf{t}_i)$  decreasing with exponential distribution, as in the most of software reliability models:

$$\mathbf{y}(\mathbf{t}_i) = e^{-\mathbf{t}_i} + \mathbf{a}, \quad (6)$$

where  $\mathbf{a}$  determines some “genetic” error level, i.e. NOD, which are appearing in program in birth stage and not found during the testing process, but could appear in production after short or long time.

We believe also the matrix  $\mathbf{X}$  with dimension  $\mathbf{n} = 20$  and  $\mathbf{q} = 3$  as a constant, the matrix  $\mathbf{Z}$  with dimension  $\mathbf{n} = 20$  and  $\mathbf{r} = 4$  as a variable value. The evaluation of  $\mathbf{s}$  with dimension  $4 \times 1$  were made for following cases of  $\mathbf{Z}$  matrix: 1) one parameter of  $\mathbf{Z}$  array is taken into account, i.e. one component of matrix  $\mathbf{Z}$  has value  $\mathbf{1}$ , but others components of matrix are  $\mathbf{0}$ , 2) more, than one parameters of  $\mathbf{Z}$  array are taken into account, i.e. more, than one component of matrix  $\mathbf{Z}$  have value equal to  $\mathbf{1}$ , but not all, others components of matrix  $\mathbf{Z}$  are  $\mathbf{0}$ , 3) all parameters  $\mathbf{Z}$  array are taken into account, i.e. all components of matrix  $\mathbf{Z}$  have value equal to  $\mathbf{1}$ . We assume various values for  $\gamma$  also: 1)  $\sigma_e^2 = \sigma_s^2$ ; 2)  $\sigma_e^2 < \sigma_s^2$ ; 3)  $\sigma_e^2 > \sigma_s^2$ .

During the experiment of the mathematical model simulation the only one «genetic» parameter was evaluated, witch is NOD in the software concerned with its «genetic» feature. We assume, that the array of the  $\mathbf{s}$  parameter also consists of random values with normal distribution and known expectation mean value  $\mathbf{m}(\mathbf{s}) = \mathbf{a}$ , we get some estimation results, shown on Figure 1 and Figure 2. For model simulation the tool Matlab 5.0 was used.

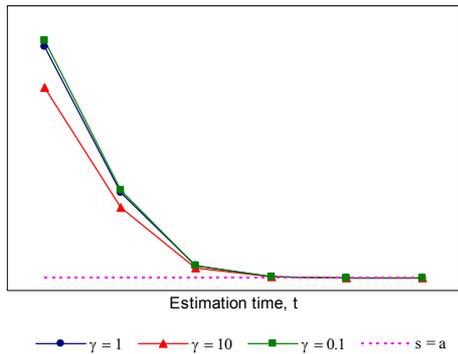


Figure 1. Uncontrolled random parameters (SQ) distribution as function of time by BLP method. One parameter of Z array is taken into account.

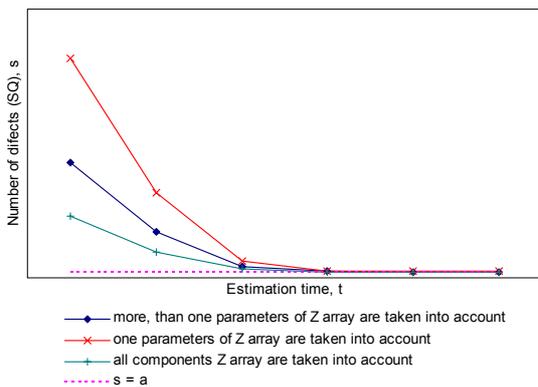


Figure 2. Uncontrolled random parameters (SQ) distribution as function of time by BLP method.  $g = 1$ . Z matrix has different values for array parameters

Figure 1 and Figure 2 demonstrate evaluation results of simulation model, which we get using BLP method and the formula (3). The Figure 1 is  $s(t)$  - uncontrolled random parameters (SQ) distribution for case, when Z is constant, but parameter  $\gamma$  is a variable value. The Figure 2 is  $s(t)$  SQ distribution for case, when  $\gamma$  is constant, but Z a variable value. We can see that both curves in the beginning of the testing process have more defects (NOD), than in the end of testing. The NOD as parameter of SQ is decreasing as an exponent and strives to a constant value  $s = a$ , meanwhile the actual quality of software is increasing. So we reveal, that value  $s = a$  actually is the “genetic” software feature in our experiments.

The given graphical results represent the rate of the of the NOD algorithm convergence to the known

beforehand result. The more is  $\gamma = \sigma_e^2 / \sigma_s^2$ , relation of noise to the uncontrolled random parameters, the more is the rate of the algorithm convergence (Figure 1.). The less is the number of the  $s$  components, which has influence to SQ, i.e. “genetic” parameter, the smaller rate of NOD distribution is (Figure 2.).

In the real life software development process we do not know the expectation  $m(s)$  of uncontrolled random (SQ) beforehand, that is why is could be used the other approach for “genetic” software feature’s evaluation using BLUP method, which assumes zero expectation for  $s$  value. Then the analysis of  $s$  distribution for BLUP could be such:  $s$  parameter is negative for almost all time of testing, which means bad SQ, meanwhile positive values of  $s$  define good SQ, and the value, for which  $s$  is streaming in time by exponent could be accepted as “genetic” software feature.

## REAL SYSTEM SIMULATION

Two previous paragraphs discussed the mathematical model of software system for initial quality of software evaluation using system parameters in the output of the software system. This part of the paper produces a model description for real application system simulation by traditional way, i.e. using input parameters:

1. Suppose we have three urns, which simulate three real application systems under test.
2. Let’s put a random number of red balls N, which means the number of program code lines with errors, into each urn with the known expectation and the known variance.
3. Let’s put the known number of white balls M to the every urn. The white ball means the line of application code without errors. The real number of program code lines could be taken from the real software systems.
4. When red and white balls (wrong and right code lines) are distributed among the urns (software) let’s take out balls from each urn. This step corresponds to the software testing and error prevention process: if a chosen ball is white, it should be returned back into the urn, where it was initially placed; if the chosen ball is red, it should be taken out from the urn and replaced by a white ball – wrong code line is repaired to a right one. Here we believe, that: a) color of the ball is defined correctly; b) there is an error in color of ball definition: probability  $P_1$  – color of the ball is defined rightly,  $P_2$  - not rightly.
5. Make a prediction of number of red balls in each urn according to the BLUP method (see previous paragraphs).
6. Do the next selection (testing iteration) of balls from urns: repeat steps 4 and 5 i times.
7. Compare results of BLUP method evaluation (from 5<sup>th</sup> step) with number of red balls after i iteration.

We suppose, that after realization of this scenario number of defects in the software is very small and it corresponds to results received by BLUP before and called by software “genetic” feature.

There are many examples of software reliability research works, which show the similar NOD distributions in real life system development and testing (Wood 2003).

The described scenario realization for software system simulation is developed in C++. It shows that our supposition was right, but the task of theoretical results comparison with simulated system is not finished yet. This is the reason, why these results are not published here.

## CONCLUSIONS

This exploration work shows that software quality could be predicted and evaluated using a term of “genetic” feature of the software and stated above theoretical theses. Currently the author is working in research of the theoretical approach application to the real software system using the simulation model described overhead. We believe that our SQ estimation approach of software “genetic” feature is a potential way in research of good reliability and quality for application system.

## REFERENCES

- Kokina J. 2003. “Software Reliability And The “Genetic” Feature Of The Software”. *Riga Technical University Scientific Proceedings 2003.*
- Kokina J., Petersons E. 2004. “The Mathematical Model For Measurement Quality, Using The “Genetic” Feature Of the Software”. *Riga Technical University Scientific Proceedings 2004.*
- Henderson Charles R. 1984. “Application of Linear Models in Animal Breeding. University of Guelph”.
- Wood Allan P. 2003. “Software Reliability From the Customer View”. Computer, August.

## AUTHOR BIOGRAPHIE



**JEKATERINA KOKINA** was born in Riga, Latvia and went to the Riga Aviation University, where she studied telecommunications and obtained Master degree in 1999. She worked 6 years for DataPro company and then 2 years for the MicrolinkLatvia company as a software testing specialist. From the 2002 she is a post-graduate student in Riga Technical University, working in the field of the software testing and quality with the assistance of the professor Ernests Petersons. J.Kokina participated in international RTU conferences in 2003 with a report “Software Reliability and The “Genetic” Feature of The Software” and in and 2004 with report “The Mathematical Model For Measurement Quality, Using The “Genetic” Feature Of the Software” E-mail address is: jkokina@yahoo.com.



# **Simulation and Education**



# REVISION OF MATHEMATICAL APPROACH TO ELECTRICAL CIRCUIT MODELLING

Mirko Dozet  
Predrag Valozic  
Department of Electrical Engineering  
Polytechnic of Zagreb  
Konavoska 2, 10000, Zagreb, Croatia  
E-mail: pvalozic@tesla.vtszg.hr

## KEYWORDS

Model, Circuit, Analysis, Numerical, Computer.

## ABSTRACT

Discussion of two possible approaches to the same problem: linear electrical circuits modelling is used to define the mathematical background which students need to master before they start with the circuit analysis. Classical, analytical approach is compared to numerical one, and the better choice is proposed. Consequences to the teaching, for the moment, are unpredictable but intriguing.

An RLC circuit is a frequent example in the most of textbooks (Carter and Richardson 1972, Desoer and Kuh 1969, Nilsson and Riedel 1996) about electrical engineering theory fundamentals. Although the circuit is as simple as possible, it is of great educational value. Mathematics needed for the circuit analysis is a core of mathematic modelling used in all other linear electrical circuits analysis. The difference is in the number of components and equations complexity, but the principles are the same.

## INTRODUCTION

It is customary to analyse the series RLC circuit (Fig. 1) with the initial circuit current  $i(0)$  and electrical charge  $q(0)$ . The circuit's history is of no importance; the future is important! The R, L and C values are known, voltages and current are oriented and marked as shown in circuits diagram. Due to the lack of independent voltage or current sources, the circuit response on the initial conditions is the free one (zero-input response). The circuit energy is dissipated gradually on the resistance R in the form of heat, so it is to be expected that current and charge, sooner or later, are reduced to zero.

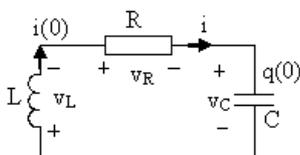


Figure 1: Series RLC Circuit Diagram

## CIRCUIT ANALYSIS

For the clockwise referent loop direction, Kirchhoff's voltage law equation is:

$$v_L + v_R + v_C = 0 \quad (1)$$

Voltages are calculated from the current and charge:

$$v_L = L \frac{di}{dt} \quad (2a)$$

$$v_R = R \cdot i \quad (2b)$$

$$v_C = \frac{q}{C} \quad (2c)$$

$$i = \frac{dq}{dt} \quad (2d)$$

By inserting 2a, 2b and 2c into 1, differentiating and dividing by L, one could get:

$$\frac{d^2i}{dt^2} + \frac{R}{L} \frac{di}{dt} + \frac{i}{LC} = 0 \quad (3)$$

Equation (3) is the second order homogenous differential equation with constant coefficients. Equation solution is started by the **assumption**, that it is in the following form:

$$i = Ae^{-rt} \quad (4)$$

The constants A and r are to be calculated from the circuit initial values and elements values. After (4) is inserted in (3), the so called characteristic equation is:

$$r^2 + \frac{R}{L}r + \frac{1}{LC} = 0 \quad (5)$$

Two new values: the damping factor  $\delta$  and free oscillations angular frequency  $\omega_0$  are defined as follows:

$$\delta = \frac{R}{2L} \quad (6a)$$

$$\omega_0 = \frac{1}{\sqrt{LC}} \quad (6b)$$

With (6a) and (6b), (5) may be transformed to:

$$r^2 + 2\delta \cdot r + \omega_0^2 = 0 \quad (7)$$

By **well known procedure**, two solutions are:

$$r_{1,2} = -\delta \pm \sqrt{\delta^2 - \omega_0^2} \quad (8)$$

Generally, characteristic equation (5) has two solutions. It is **expected** that the circuit current is:

$$i = Ae^{-r_1 t} + Be^{-r_2 t} \quad (9)$$

Constants A and B are to be calculated from the initial condition:

$$A + B = i(0) \quad (10)$$

$$r_1 A + r_2 B = -\frac{i(0)R}{L} - \frac{q(0)}{LC} \quad (11)$$

With **a little bit longer procedure**, one could get a general solution for the circuit current  $i(t)$ :

$$i(t) = i(0) \cdot e^{-\delta t} \left[ ch\sqrt{\delta^2 - \omega_0^2} t - \frac{\delta}{\sqrt{\delta^2 - \omega_0^2}} sh\sqrt{\delta^2 - \omega_0^2} t \right] - \frac{q(0)\omega_0^2}{\sqrt{\delta^2 - \omega_0^2}} e^{-\delta t} sh\sqrt{\delta^2 - \omega_0^2} t \quad (12)$$

By making use of:

$$q(t) = \int_0^t i(t) dt + q(0) \quad (13)$$

it is obtained:

$$q(t) = \frac{i(0)e^{-\delta t} sh\sqrt{\delta^2 - \omega_0^2} t}{\sqrt{\delta^2 - \omega_0^2}} + i(0) \cdot e^{-\delta t} \left[ \frac{\delta}{\sqrt{\delta^2 - \omega_0^2}} sh\sqrt{\delta^2 - \omega_0^2} t + ch\sqrt{\delta^2 - \omega_0^2} t \right] \quad (14)$$

From (12), (14) and (2) it is possible to calculate voltages  $v_L$ ,  $v_R$  and  $v_C$ . Then, it is possible to determine zero and extreme values, as well as powers and energies.

In science, it is customary that starting from general case, one get certain special cases. Equations (12) and (14) are generators of next examples.

## Special Cases

In order to draw graphs of the circuit current  $i(t)$  and capacitor charge  $q(t)$ , equations are to be slightly modified. Instead of real values of R, L and C, and initial values  $i(0)$  and  $q(0)$  qualitative waveforms depend on their ratio. Normalized time  $t_n$  and relative damping factor  $k$  are defined as:

$$\omega_0 t = t_n \quad \frac{\delta}{\omega_0} = k \quad (15)$$

Value K is:

$$K = \frac{q(0)\omega_0}{i(0)} \quad (16)$$

There are four possible characteristic cases, dependent on the  $k$  value:  $k > 1$ ,  $k = 1$ ,  $k < 1$  and  $k = 0$ . Two of them are presented.

### Aperiodic Case, $k > 1$

If  $k > 1$  then  $\delta > \omega_0$  and  $R > 2\sqrt{\frac{L}{C}}$  so (12) and (14) are:

$$i_n = \frac{i(t_n)}{i(0)} = e^{-kt_n} \left[ ch\sqrt{k^2 - 1}t_n - \frac{k + K}{\sqrt{k^2 - 1}} sh\sqrt{k^2 - 1}t_n \right] \quad (17)$$

$$q_n = \frac{q(t_n)}{q(0)} = e^{-kt_n} \left[ \left(k + \frac{1}{K}\right) \frac{sh\sqrt{k^2 - 1}t_n}{\sqrt{k^2 - 1}} + ch\sqrt{k^2 - 1}t_n \right] \quad (18)$$

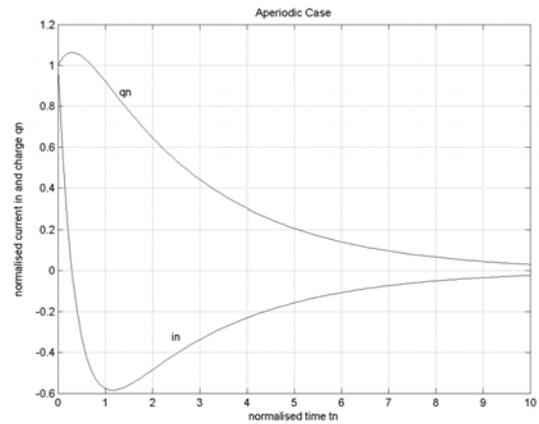


Figure 2: Aperiodic Case,  $K=2$ ;  $k=1,5$

Graphs in figure 2 are performed by MatLab 6. From Fig. 2 it is evident that the charge  $q(t)$  gains maximum value at the moment when the current is zero. The same conclusion comes from the energy conservation principle: the circuit energy is oscillating between the only two energy conservative elements: L and C.

## Damped Oscillations, $k < 1$

If  $k < 1$  then  $\delta < \omega_0$  then  $R < 2\sqrt{\frac{L}{C}}$

Instead of hyperbolic functions in (17) and (18), due to negative values under the square roots, one can gain trigonometric functions in the **well known manner**:

$$i_n = e^{-kt_n} \left[ \cos \sqrt{1-k^2} t_n - \frac{k+K}{\sqrt{1-k^2}} \sin \sqrt{1-k^2} t_n \right] \quad (19)$$

$$q_n = e^{-kt_n} \left[ \left(k + \frac{1}{K}\right) \frac{\sin \sqrt{1-k^2} t_n}{\sqrt{1-k^2}} + \cos \sqrt{1-k^2} t_n \right] \quad (20)$$

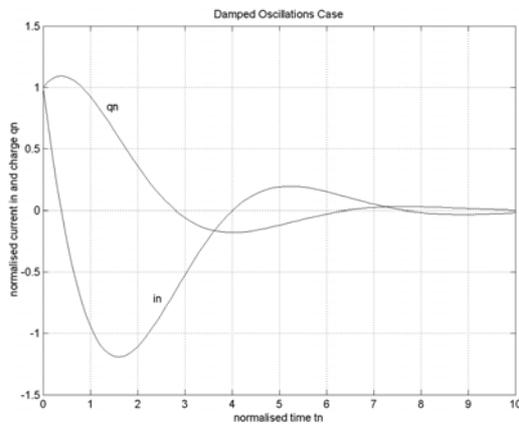


Figure 3: Damped Oscillations,  $K=2$ ;  $k=0,5$

Generally, starting from the given values of  $R$ ,  $L$ ,  $C$ ,  $i(0)$  and  $q(0)$ ,  $\delta$ ,  $\omega_0$ ,  $k$  and  $K$  are to be calculated. One of the four possible cases is determined by the  $k$ -value. Adequate equations are picked up,  $K$  and  $k$  values inserted and normalized values calculated. Real values  $i(t)$  and  $q(t)$  are gained by  $i(0)$  and  $q(0)$  multiplying, respectively. Real time oscillations are calculated by scaling with the  $\omega_0$  value.

## NUMERICAL, COMPUTER BASED APPROACH

Starting from the same initial equation (1) we get:

$$v_L = -v_R - v_C \quad (21)$$

Then, equations (2) are modified in the difference equation form instead of differential form.

$$v_{Li} = L \frac{\Delta i_i}{\Delta t} \quad (22a)$$

$$v_{Ri} = R \cdot i_i \quad (22b)$$

$$v_{Ci} = \frac{q_i}{C} \quad (22c)$$

$$i_i = i_{i-1} + \Delta i_{i-1} \quad (22d)$$

By inserting (22a), (22b) i (22c) into (21), one gets:

$$\Delta i_i = -\frac{R}{L} \cdot i_i \cdot \Delta t - \frac{1}{L \cdot C} q_i \cdot \Delta t \quad (23)$$

Equation (23) in spreadsheet MS Excel is programmed (Fig. 4). In A and B columns there are circuit values and calculated values. In C column is time index, in column D is discrete time  $t_i$ , in E is discrete current difference  $\Delta i_i$  and in the F column, the current is calculated based on (22d) equation. Formula (23) written in Excel programming syntax is presented at fig. 6. too.

Figure 4: MS Excel spreadsheet table

Instead of equation (13) for the time function of the charge  $q(t)$  time-discrete charge values are calculated by recursive addition:

$$q_i = i_i \cdot \Delta t + q_{i-1} \quad (24)$$

Therefore, problems with numerical integration are, for the moment, avoided. Calculated values are in the G-columns.

In B1 cell there is  $R$  value, in B2 is  $L$ ,  $C$  is in B3. Initial charge value is in B4 cell and initial current in B5 is placed. Time-step value  $\Delta t$  is in B6 cell. Calculated free-oscillations angular frequency is in B8, frequency is B9 and period in B10. Values  $\delta$ ,  $k$  and  $K$  are in cells B12, B13 and B14, respectively.

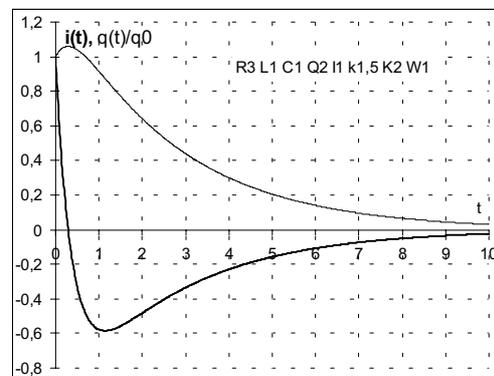


Figure 5: Aperiodic Case,  $K=2$ ;  $k=1,5$

Each of four characteristic cases are calculated and graphically presented, Figs. 5 and 6.

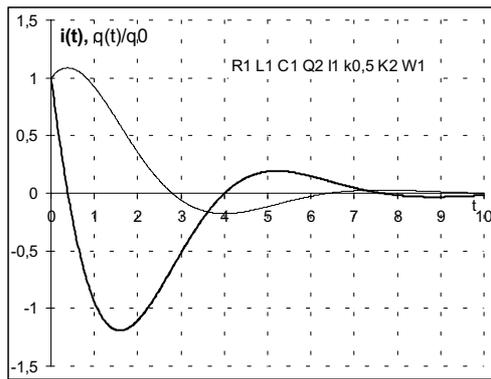


Figure 6: Damped Oscillations,  $K=2$ ;  $k=0,5$

## CONCLUSION

Illustrated by a well known simple example from electrical networks, the intention of this paper is to mark mathematical knowledge and skill needed for teaching and learning of electrical circuits analysis. Together with fundamental laws of physics and electricity, the student has to implement trigonometric and hyperbolic functions, algebraic equations solving and their roots discussion, linear differential equations solving, and limits, derivatives and integrals calculating. Higher semester students should be able to solve those problems by Laplace transformation, too.

Expected previous knowledge is bolded in the paper above:

- Waveform assumption, equation (4).
- Well known differential equations (4) to (7) solving procedure.
- Routine procedure from (10, 11) gives A and B...

Only a few students can understand processes dynamics directly from equations (12) or (14) even in the case of simple circuit such as series RLC. The saying "Picture is thousand words worth!" is confirmed here, i.e. it is obvious why visualization and graphs are so frequently used. So, if graphs are a suitable way to present electrical circuits processes, it is reasonable to pose a question:

Is it possible to make a bypass, and generate graphs by simpler, better or more accessible mathematical procedure? From the point of a student, of course!

Or:

Could it be, that computer based approach will reduce and simplify mathematical background the student needs to understand processes to be learned?

Today, there is broad spectrum of software tools for circuits analysis even without any higher mathematics knowledge. All one needs is to understand the problem, know how to insert data and how to interpret results obtained.

But, it is doubtful for the knowledge level if lecturer's theoretical explanation is illustrated and supported by simulation programs or virtual laboratories only. What is student's knowledge benefit if he or she has to type data in marked fields, press "ENTER" and look to or print data table or graph? And do it tens or hundreds of time!

The middle way seems to be more productive: Computer modelling in general purpose programming language, based on well known fundamentals of physics (charge and energy conservation, Ohm's and Kirchhoff's law) and mathematics which students have already mastered.

In the previous chapter such an approach is described. Outcomes of numerical and analytical models are quite satisfactory and close to each other. Mathematics used in numerical modelling vs. analytical is quite simple and MS Excel programming is a basic of 21-th century literacy!

Consequently, teaching subjects like Electrical Engineering Fundamentals does not have to wait for Mathematics to be completed. It is even more important now, when all European countries are faced with education system crisis and Bologna initiative as possible solution. Shortened first study level (three years) gives no time for a traditional sequence (Math I, Math II, Physics...). Each reasonable way to start with core teaching subjects without any introduction is to be considered.

The example presented is well known just for the reason to be widely recognized: RLC circuit and its mathematical modelling, comparison of the analytical and numerical one. It is shown that the starting point: physical laws of a serial RLC electrical circuit and final result to be discussed: current, voltage and charge diagrams are the same! But, in the case of numerical approach, the "path" is much shorter (just few equations) and simpler (multiplication and addition instead of differential equations solving). Instead of detailed mathematical analysis, numerical mathematics is to be preferred! If needed, deeper analysis based on higher mathematics may be postponed for the second, graduated study level.

## REFERENCES

- Carter, G. W. and A. Richardson. 1972. "Techniques of Circuit Analysis", University Press, Cambridge.
- Desoer, C. A. and E. S. Kuh. 1969. "Basic Circuit Theory", Mc Graw Hill-Koga Kusha, Tokyo.
- Nilsson, J. W. and S. A. Riedel. 1996. "Electric Circuits", Addison - Wesley Publ. Comp., Reading, Massachusetts.
- Valozic, P. 2003. "Some Experience on Learning via a Computer", in Proceedings of the Eurocon 2003, The International Conference on Computer as a Tool, 22-24-September 2003, Ljubljana, Slovenia
- Valozic, P. 2002. "Signal Processing at an Electronic Workplace", in Proceedings of the IASTED International Conference Applied Simulation and Modelling, June 25 - 28, 2002, Crete, Greece.

# ALGORITHMIC AUTONOMY ARCHITECTURE (AAA) – THE PRINCIPLES OF BUILDING INFORMATION SYSTEMS WITH APPLICATIONS IN SIMULATION AND EDUCATION

Dr. Janis Sedols  
Research Institute of Mathematics and Computer Science, University of Latvia  
Rainis boulvd. 29, Riga LV-1459, Latvia  
E-mail: [sedols@cclu.lv](mailto:sedols@cclu.lv)

Sniedze Sedola  
SWH SETS  
Antonijas str. 8, Riga LV-1010, Latvia  
E-mail: [Sniedze.Sedola@sets.lv](mailto:Sniedze.Sedola@sets.lv)

## KEYWORDS

Algorithm, autonomy, architecture, information system, education, program, model, simulation, geography, game, center.

## ABSTRACT

The principles of building information systems are considered which has emerged from practical experience developing applications for simulation and education. The principles are directed to increase the to increase stability of the IS, to improve data exchange between users and to rise creativity. The short report is presented on already completed systems based on these principles – simulation system SITA and geographical game EARTH. The brief description is given of the theory called centristics which originated from game EARTH and deals with centers of geographical areas.

## INTRODUCTION

The main use of a computer is working with or building an information system (IS) consisting of programs, data and documents. Thus, to solve a problem using simulation one takes a simulation program, builds some models, gets results (data) from them and includes these in the text of a document, for example, a report or a scientific paper. The entirety of all this can be called information system involved in the particular work.

The principles of building IS called Algorithmic Autonomy Architecture (AAA) which will be discussed

below are directed to increase stability of the IS, to improve data exchange between users and to motivate creative work with a computer. The principles emerged from practical experience working with computers and by summing up methods of building software from different sources. The most of the principles are not very original by themselves, but they have not been integrated in one common system.

These principles of AAA are practically applied and tried in building IS in areas of simulation and education. The examples are simulation system SITA and geographical game EARTH

## PROPOSALS

**The structure of blocks.** An IS consists of relatively independent parts called blocks. Blocks can be combined in different ways making optimal IS for the particular situation. Every block belongs to one of two categories – the programs or P-blocks and the data or D-blocks. The P-blocks contains programs and their descriptions, the D-blocks – the data sets for specific needs created and processed by core programs. The stability of the IS is heavily based on interaction of two categories – a solid core produces a vast number of D-blocks, which in turn makes it necessary to improve the P-blocks regularly.

**The link between programs and textual information.** The textual material of an IS can belong to either P-blocks (the user guides for programs) or the D-blocks (the information about specific application). Programs and texts can be separated or subject to one

another both ways – you can invoke texts from a program (help) or contribute texts with an option to run the programs mentioned within (executable hyperlinks). AAA provides both opportunities. Programs can access help, D-blocks can contain texts with executable hyperlinks, but the program Prologue described next gives opportunity to run programs or to read texts. Besides AAA has its own text formatting which makes them easy to use everywhere as well as publish in the Internet.

**Manager program Prologue.** The manager program Prologue makes IS very easy to use. It shows the structure of the IS and provides access to all its components. In Prologue you can run programs, open documents for reading and choose the necessary blocks from lists. The Prologue contains language options mentioned below. The Prologue is common for all types of IS that belong to AAA.

**The language options.** AAA makes localization of existing IS for use in different languages rather simple. You have to translate documents and program text resources which are extracted in separate files. If IS has more than one language, you can choose a working language from list in Prologue.

**Raising creativity.** The principles of AAA encourage the user not only to use the IS but also to participate in development. The simplest way of doing this would be building an IS from readymade blocks. The next step is making new D-blocks or expanding existing ones. Rather simple is changing documents in P-blocks, which can be improved and enhanced as well as translated to different language. Qualified users can also make new P-block for existing or new work areas.

## IMPLEMENTATIONS

Currently there are two classes or spheres of IS based on principles of AAA – **simulation system SITA** and **geographical game EARTH**. The third sphere is AAA itself which encapsulate other spheres. Further we will briefly discuss each sphere separately.

### SPHERE AAA

AAA includes programs for common usage – aforementioned program Prolog, launching program Start and installer program which sets up AAA from a CD to user's PC. Program Start is necessary to launch Prolog

from different locations in different modes. Start also registers blocks which is necessary when the block is added or relocated to ensure the interaction between blocks.

The two variations of the Prologue window for respective spheres SITA and EARTH are shown in Figure1. On the left side of the window you can see lists where you can choose the language, the D-block you want to work with called group and subordinate D-block called field. In our case group SITA2000 contains only one field with the same name, on the other hand group "EARTH, World-Europe" has two fields WORLD and EUROPE. Every group with all its fields belongs to one of spheres.

On the right side of the Prologue window there are pushbuttons for executing programs and reading documents. The former are located on the right side within the frame with the current field name in the title. The left sides of pushbuttons for document reading make a vertical line almost in the middle of the window. A button with title "What is..." gives an abstract of the given sphere, Help gives information about the usage of Prologue, buttons with character "i" next to the program icons – about the usage of respective programs. The button at the bottom on the frame gives information about the chosen field. In the case of SITA it opens the book where the teaching aid and scientific research concerning this system are assembled. The solitary button "Go" on the left side of the window is a convenient shortcut for the top program button which is most commonly used. There is also checkbox "AAA info" in the Prologue windows. It switches the program to the state where the information is available about the AAA itself and the spheres in it.

### SIMULATION SYSTEM SITA

Statistical and analytical simulation system SITA (Janis Sedols and Madars Rikards, 2003) is intended for making and using graphic models a.k.a. charts. The example of a chart – model of the telecommunication system with repeated calls caused by the busy subscribers (Gerards Jonins and Janis Sedols, 2001) – is shown in Figure2. The full information of the model is displayed including short textual description. Using chart you can do statistical and analytical simulation and obtain the results from given data.

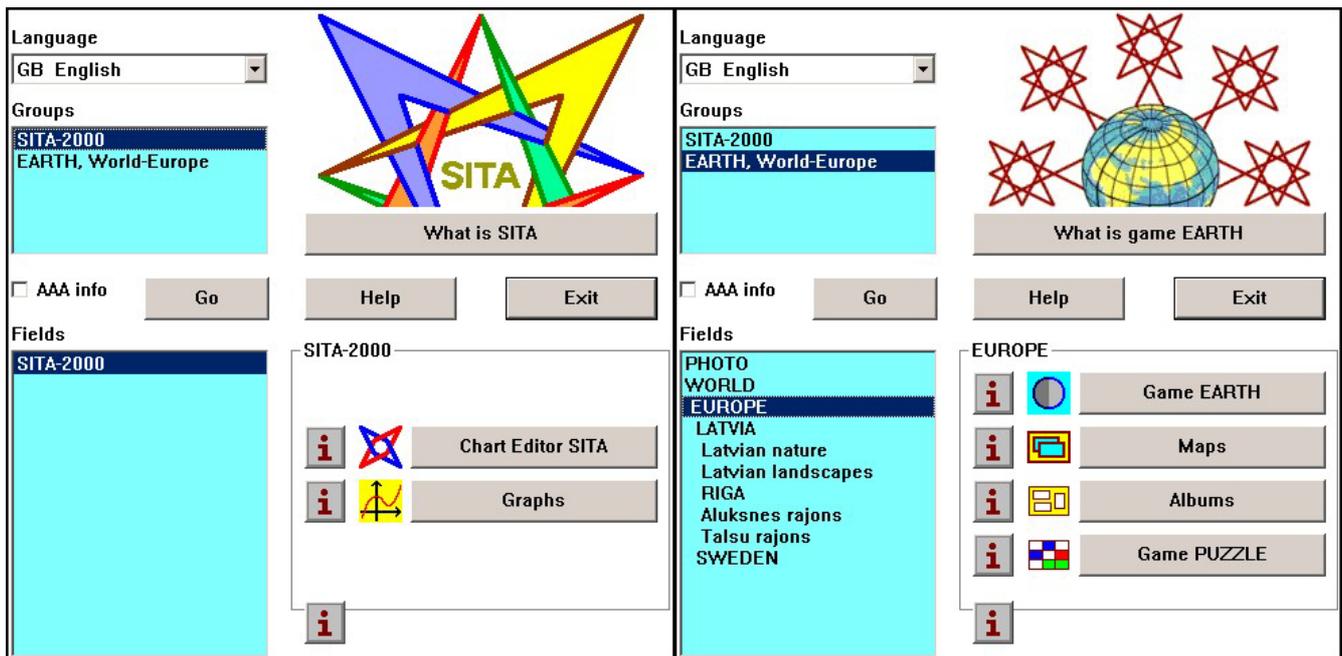


Figure 1. Prologue

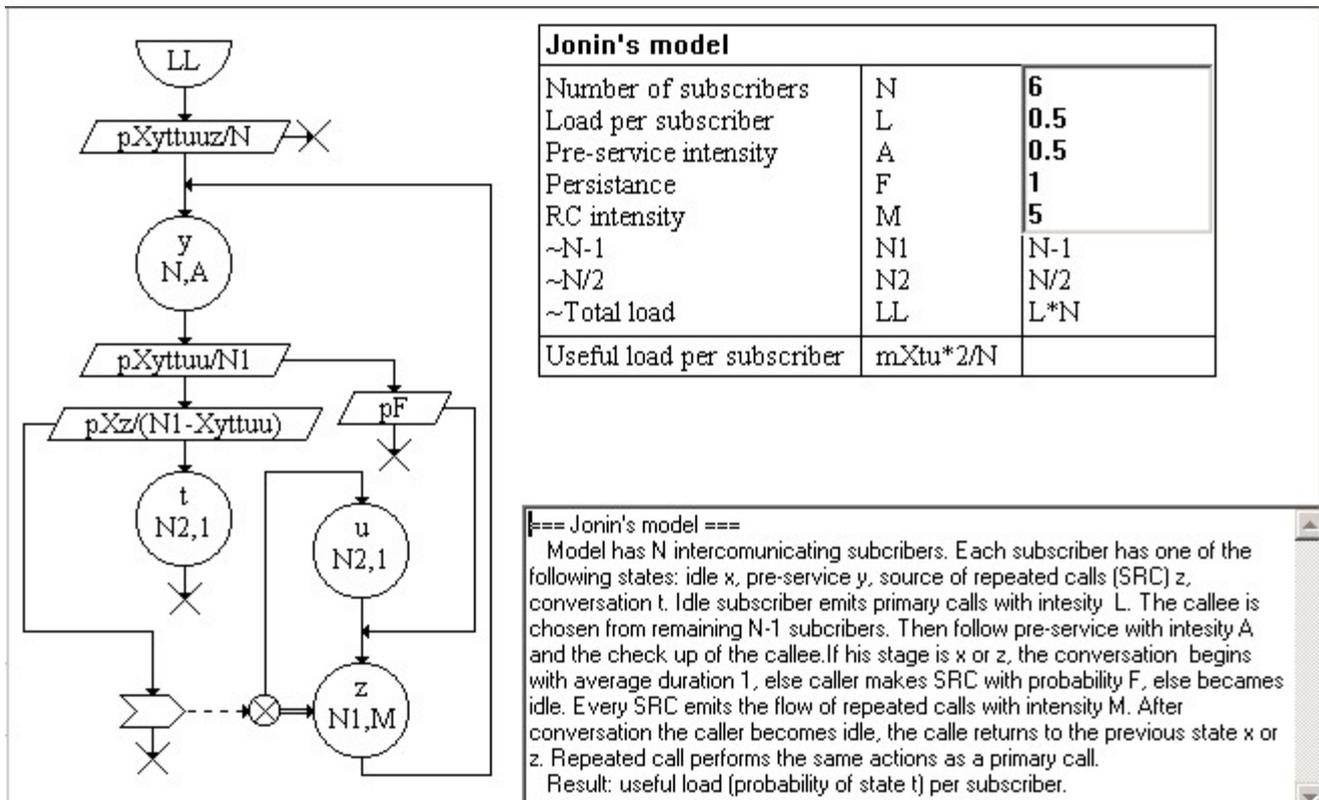


Figure 2. Example of model in system SITA

To make the example more comprehensible let us inspect it more closely. The model of communication system with finite number of subscribers and repeated calls is an example of scientific research using SITA. The model is named after one of its authors Gerards Jonins (1938 – 2001). The model is interesting because the overload phenomena can be observed – when increasing intensity of incoming flow of calls the useful or serviced load at first increases but then starts to decrease. The cause of overload is useless load which originates from pre-service needed to identify the subscriber to call. When the load is great almost all the subscribers are engaged in this phase and cannot reach the communication phase. The result of the research is effective and rather simple method to avoid overload, which briefly is this – the optimum load is calculated and it is not allowed to overreach it by denying new calls at the moments of overload.

SITA is very suitable for educational purposes due to simplicity and demonstrability. It can be used not only for modeling programs but also to make **flowcharts of numerical algorithms**. The both kinds of charts can be executed in **trace mode** which shows step by step the work of the model or running the algorithm. SITA contains the program for drawing diagrams which presents the results of modeling or calculation in the form of a graph.

## GEOGRAPHICAL GAME EARTH

**The game EARTH** is the second example of educational IS. It contains instruments to build and use a game field for any geographical territory.

A game about a geographical territory includes a set of information regarding it which can be used to learn and to check your knowledge in entertaining way as well as a reference material. The game has different modes – you can find a place in the map given its name or picture or some symbol (armorial bearings for example), you can guess the distance between two places, guess the year of historical event. The place can be a point (town or community) or a two-dimensional object – river, lake or administrative unit.

Geographical game has two main sections – **Games**, where you can choose one of 6 game modes and **Guide** which contains information necessary for games and reference material on communities, tourist objects, sights and historical events. You can consider the Guide as encyclopedia, where the data about geographical object can be found easily and quickly.

The games *LATVIA* and *RIGA* included as fields in game *EARTH* have high degree of detailing. The games *WORLD* and *EUROPE* also contain plenty of information about countries of the world but they have few pictures. There are a few drafts of games about other

territories (Sweden and some of regions of Latvia) with small amount of information.

You can obtain and verify your knowledge in Latvian geography an history using game *LATVIA*. The vast collection of photos (currently 2000) with short descriptions is added to the game. It contains information about most of the castles, manors, monuments, churches, rivers and lakes of Latvia.

Game *EARTH* includes satellite programs **Albums** and **Maps** and another game **Puzzle**.

Program **Albums** is used to browse and create collections of photos viewable on computer screen. Its purpose is to order photos by some criteria, to add illustrative texts (names and descriptions) and provide means to view them on the screen. It is necessary aid to game *EARTH*. It also can be used separately – to view photos included in the game and to make independent photo collections.

There is additional feature in program **Albums** – the tool which draws peculiar ornaments – the ANV figures invented by its author and creates gratings which can be used as decorative framing for the pictures shown by the program.

Program **Maps** is a modification of the program **Albums** for work with geographical maps. This program lets you work with pictures of maps in a similar way as the program **Albums** with photos. Game *EARTH* needs this program to prepare maps used in the game, to get coordinates for places and to create and modify two-dimensional objects (rivers, lakes, borders). The program can be used separately from game *EARTH*. You can use it to determine coordinates and distances when planning a touring route as well as to calculate area and mathematical centers for geographical territories.

## CENTRISTICS

The game *EARTH* comprises original theory called centristics about the centers of geometrical figures and geographical areas.

It is often heard about attempts to find and mark centers of geographical territories, but usually the precise definition of the center is not given. In planar geometry the only known definition for the center of the figure that is not central-symmetrical is **center of gravity**. For non-planar figures the center of gravity is outside the figure and the question about the center is left without a clear answer.

In the process of making the game *EARTH* the centristics is created – the theory that defines and lets you compute geometrical centers for planar and spherical figures. There is no limit to such centers because the **power of center  $p$**  can be any floating-point number, for example,  $p=2$  gives you center of gravity. Center with  $p=1$  is interesting where distances to all points of territory are minimized. This is the point where one

should gather all the hay from the territory to minimize transportation costs. Despite the simple definition the **haystack center** is hard to calculate even for a non-regular triangle.

The interpretation of haystack is valid for other centers too if the costs of transporting the hay are considered proportional to the distance in power  $p$ . All the centers form a continuous line – **centroid** with ends

in the centers of inscribed and circumscribed circles (the centers with  $p = -\infty$  and  $p = \infty$ ).

The game EARTH includes a tool for calculating centers of geographical territories. The centroid of Latvia is obtained using this tool and is shown in Figure 3 with the haystack center expressly marked.

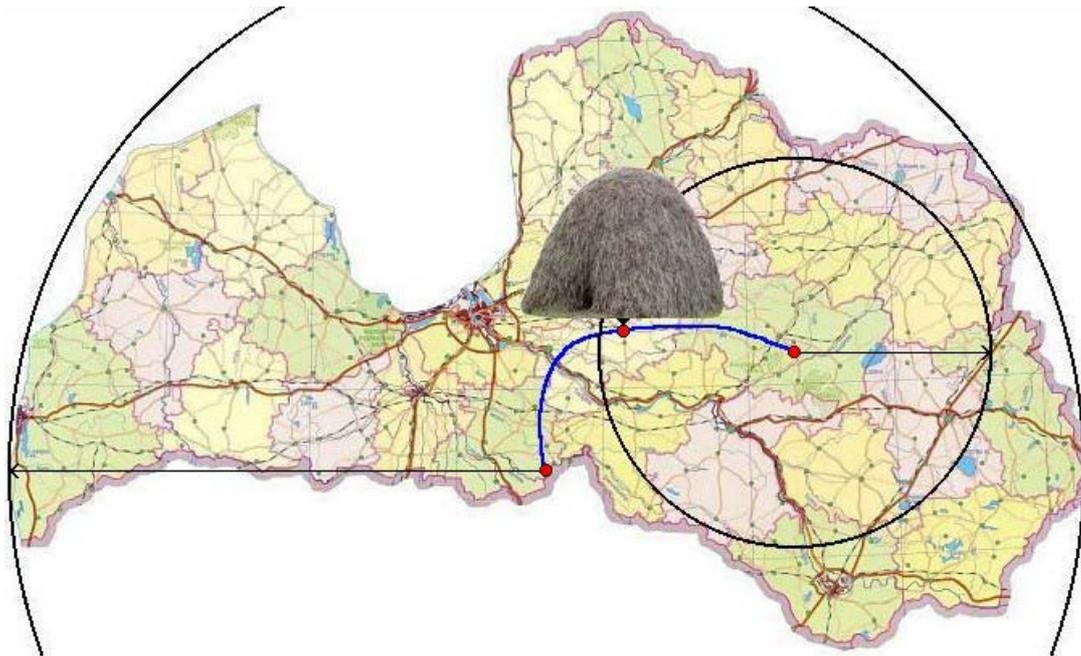


Figure 3. Centroid of Latvia

## REFERENCES

Gerards Jonins and Janis Sedols. “Analysis of Overload in the Telecommunication Systems with Repeated Calls Caused by the Busy Subscribers”. In *Proceedings of 42-nd Riga Technical University Conference*, Riga, Latvia, October 12, 2001. pp 49-50.

Janis Sedols and Madars Rikards. “SITA2000 – Research System of Models with Discrete Events for Education and Practical Purposes“. In *Proceedings of the International Workshop on Harbour, Maritime and Multimodal Logistics Modelling & Simulation HMS 2003*, September 18-20, 2003, Riga, Latvia, pp 284 - 286

## BIOGRAPHIES

**Jānis Sedols** graduated from the University of Latvia in 1961 as a mathematician and received a Dr. sc. comp. degree in 1993 at the same university. Leading

researcher of the Research Institute of Mathematics and Computer Science, University of Latvia. His professional interests include research and practical implementation of systems with discrete events. Developed and implemented simulation and analytic modeling system SITA. Jānis Sedols has about 60 scientific publications. He is involved in The Latvian Education Informatization System (LIIS) project, most notable work is intellectual computer game Earth (about geography and history).

**Sniedze Sedola** graduated from the University of Latvia in 1989 specialized in applied mathematics. Since then she worked as a programmer, developer, system analyst and consultant for various software companies in Latvia, Germany, Finland and Sweden. Her professional interests lie in the field of most recent technologies of software development and reengineering.

# TEACHING SIMULATION WITH SPREADSHEETS

Jelena Pecherska and Yuri Merkurjev  
Department of Modelling and Simulation  
Riga Technical University  
1, Kalku Street, LV-1658 Riga, Latvia  
E-mail: merkur@itl.rtu.lv, jelena@itl.rtu.lv

## KEYWORDS

Spreadsheets, discrete-event simulation, System Dynamics, education

## ABSTRACT

This paper focuses on a spreadsheet-based approach to teaching simulation. The objective is to introduce spreadsheets as a powerful simulation tool for educational purposes. The spreadsheet simulation facilitates the familiarisation with basic simulation concepts apart from simulation software. The paper deals with basic types of simulation models suitable for spreadsheets. The static stochastic or Monte Carlo simulation models are introduced through a sample model for lead-time demand distribution evaluation. The two main types of dynamic simulation spreadsheets models are overviewed as well. The discrete-event system simulation basic concepts are introduced by simulating a two-server system with a single queue. A population dynamics simulation model presents the System Dynamics concepts. The basic simulation model types that are available on spreadsheets make it possible to study simulation concepts, methodology, and techniques as well as to use spreadsheets for on-hand simulation in business games, case studies, engineering, etc. The overviewed experiences in using spreadsheets for teaching simulation result in indication of spreadsheets as an alternative simulation software.

## INTRODUCTION

Simulation studies as well as simulation techniques studies can be complicated by unreasonable usage of professional simulation software. This statement is especially valid for overview or introductory courses. Simulation tools vary from universal programming languages to special simulation software. Model creation using simulation software requires experiences and/or training as well as a good knowledge of simulation theory. However, there is an alternative approach to learn the basic simulation concepts and methodology. Spreadsheets provide almost complete set of tools for simulation. Students' prior experience using spreadsheets is both advantage and disadvantage. The advantage of using a familiar tool is obvious. The disadvantage is due to the use of spreadsheets mostly as an illustrative and calculating tool. Most users are not acquainted with spreadsheets as rather powerful software. Of course, the use of spreadsheets as a simulation tool requires specific skills and the

effectiveness of simulation depends upon the developer's efficiency. Curriculum simulation studies can be carried out by using spreadsheets. Such a possibility is due to the development of electronic spreadsheets in recent years. Spreadsheets can also facilitate on-hand simulations. This paper describes the usage of spreadsheets at various stages of simulation studies such as input data analysis, random variate generation, and simulation model creation. Microsoft Excel is used throughout the paper, although most functions and models can be transformed into other spreadsheet formats.

## TYPES OF SIMULATION MODELS ON SPREADSHEETS

After formulating a simulation task, an appropriate logical model should be developed. The choice of simulation model type can be prescribed by characteristics of the system and an objective of the study. One can create either a static or dynamic simulation model. The created model can be deterministic or stochastic as well as discrete or continuous. This paper is related to some types of simulation models such as Monte Carlo, discrete-event simulation, and System Dynamics models.

## MONTE CARLO SIMULATION ON SPREADSHEETS

Monte Carlo methods generally are stochastic ones. Simulation models that contain uncertainties and are static are called Monte Carlo simulations (Banks et al. 1996). Monte Carlo simulation is based on the use of random numbers and probability statistics to investigate a variety of problems. Monte Carlo simulations are often used in such areas as economics, logistics, management, etc. Sometimes Monte Carlo simulation is defined as a sampling experiment (Evans and Olson, 1998). Actually, to call something a "Monte Carlo" experiment or simulation, all you need to do is to use random numbers to examine a problem.

Spreadsheets provide a variety of built-in statistical functions. For example, in Excel such functions as BETADIST, CHIDIST, EXPONDIST and other popular distribution functions are available. Built-in functions facilitate Monte Carlo simulation on spreadsheets, but an experienced user is not restricted by built-in distributions.

As an example of Monte Carlo simulation on spreadsheets let's consider solving of the lead-time

demand evaluation (Banks et al. 1996). In this case bulk rolls of newsprint are the inventory items. The daily demand for rolls and the lead-time are described by the probability distributions. Both distributions are used for definition of uncertain inputs of the model. In this simple example a Monte Carlo simulation represents 100 random trials. In each trial one value of a random number is used for lead-time value generation. The obtained lead-time value is used for demand during lead-time generation using different random numbers. Performing a predefined number of trials gives a possibility to summarize results and assess lead time demand. The problem is solved as the lead-time demand distribution is determined. The histogram of lead-time demand after 100 trials is shown in Figure 1.

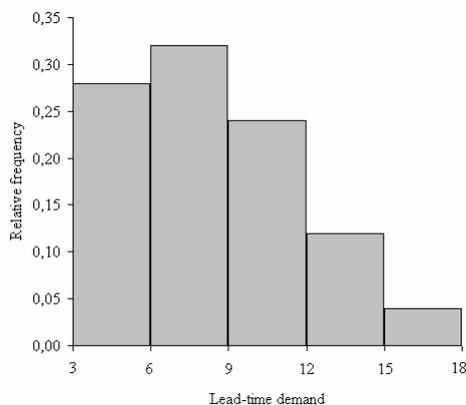


Figure 1: Histogram of Lead-Time Demand for Monte Carlo Simulation.

Monte Carlo simulation allows building in the uncertainty (present in most real-life problems) into spreadsheet models. This example is developed using only standard Excel statistical, lookup, and reference functions. A spreadsheet model contains cells which have random values generated from the given probability distributions. Every time the spreadsheet is recalculated, these random values change. After recalculation the different set of simulation results is drawn. Replications using the same parameters help to figure out the most likely behaviour of the unknown random lead-time demand. The use of Monte Carlo method for teaching simulation results in better understanding of the probabilistic nature of real-life occurrences as well as better insight into different problem solving with uncertainties.

### SYSTEMS SIMULATION ON SPREADSHEETS

Unlike Monte Carlo simulation, systems simulation models sequences of events over time. This section introduces a common approach to dynamic systems simulation on spreadsheets. Actually, the range of systems under consideration is restricted to discrete-event systems and continuous systems.

### Spreadsheet Discrete-Event Simulation Models

In discrete-event systems the state variables change only at discrete time moments. An occurrence that changes the system state is an event. Thus, discrete-event dynamic simulation models, independently of the means of realization, always use some kind of time advance algorithm. The simulation clock value recalculation in spreadsheet models is performed using one of two basic approaches: time advance using fixed increment or time advance using event scheduling. Time advance using fixed increment in discrete-event simulation differs from genuine event scheduling. However, the event planning within both approaches is rather close. The discrete event system simulation with time advance using fixed increment is not considered separately in this paper.

Traditionally, simulation basics of discrete-event systems are introduced via a single-server queuing system. A single server queuing system model on a spreadsheet gives a possibility to introduce almost all basic discrete-event simulation concepts. The sample model under consideration is rather close to the basic one.

The simulated system consists of an independent entity source, a common FIFO queue, and two service devices. The source and devices use random variables during the simulation process. At simulation time  $t=0$  the system is "empty". Random variates for arrival times and cycle times are generated using individual probability mass functions. Entities interarrival time distribution is shown in Table 1.

Table 1: Interarrival Time Distribution of Entities

Time between Arrivals (Minutes)	Probability
1	0.25
2	0.40
3	0.20
4	0.15

The distributions of service times for both service devices are given in the same way. Sample data is used for interarrival time and service time generation as it is described in **Random Variate Generation** section. Arrival times and service times are unique for each replication. Data tables are located on the separate model sheet and can be easily changed without affecting the model logic.

The considered system spreadsheet model is worked out using the event-scheduling approach. The model algorithm concentrates on events and their effect on system state. Any time moments between events are ignored. Each model spreadsheet row keeps a record necessary to advance simulation time and activate future events in time order. Records are put into spreadsheet cells and contain formulas and conditions for future changes. The current system state provides the possibility to produce the next system snapshot after

advancing simulation time up to the moment of the simulation End event.

During simulation the appearance of the model spreadsheet changes automatically. Further we will concentrate on a single replication.

The results of a single replication are available via the Excel spreadsheet. By adding some extra calculations we can obtain necessary statistics. During each simulation trial the system state is updated and statistical performance measures are recalculated according to scheduled events' times. In this case the system status variables are recalculated at discrete time moments  $t_{i+1}=t_i+dt$  ( $dt$  = random time interval between scheduled events). A fragment of the spreadsheet simulation screen is shown in Figure 2. The system simulation model consists of system snapshot series at discrete points of time. A given snapshot includes system state at time  $t_i$ , entity attributes, future event list and necessary cumulative statistics and counters. The future event list contains event notices that have been scheduled to occur at future time. The system snapshot at time 0 is additionally defined by initial conditions as well as by the generation of exogenous events. For this study the initial state is an "empty" queue, free service devices. The exogenous event is the first entity arrival and is scheduled at time 0. Service completion events for both servers are generated and scheduled in the future event list when the entity arrives or upon the system state.

	A	B	C	Event	States	Attributes			
	Customer Arrives	Customer Service	Service F Process	Type	Q	S1	S2	Arr	SS1
5	Initial	0	0	0	0	0	0	0	0
6		1		0	3	1	0	0	1
7		2		2	3	0	1	1	2
8			1	2	4	2	0	0	1
9		3		3	4	3	0	1	4
10			2	3	5	1	0	1	0
11			3	3	6	2	0	0	0
12		4		4	6	3	0	1	0
13			5	5	7	3	0	1	7
14		6		5	8	3	1	1	8
15			4	6	10	1	0	1	1
16		7		6	10	3	1	1	10
17			5	7	11	2	0	1	1

Figure 2: A Fragment of the Single Queue System Model

A stopping event, here called End, defines the length of a simulation run. This event is scheduled at time 0 for a specified future time.

This example introduces discrete-event concepts using Excel lookup and reference functions. Such concepts as entity, entity attributes, system state, event, event notice and event list, simulation clock are dealt with while creating a spreadsheet simulation model. In contradistinction to simulation software application, model building on spreadsheets is practically impossible without clear and deep perception of basic modelling concepts.

## System Dynamics Spreadsheet Models

System dynamics (SD) is another approach to systems analysis using simulation. In system dynamics various complex systems (e.g. population, ecological, and economic, etc.) are simulated. Actually, these simulations mean substitution of a continuous system with a discrete simulation model.

The basic model structure in system dynamics includes a number of elements such as levels and flows, decision functions and information links. Elements of the SD model have specific features and form a complete set of equations for simulation. The simulation trial can be interpreted as a sequence of solutions of these equations calculated with a fixed time interval. Each solution is unambiguously defined by the previous solution and the length of fixed time interval. Actually, the model logical base can be formulated by means of spreadsheets very effectively and briefly. Two main components completely describe the SD spreadsheet model. They are the initial state variable values and the set of equations.

As a sample SD spreadsheet model the simplified fish population system is considered (that corresponds to the well-known Fish Banks resource management game). The system structure is shown in Figure 3.

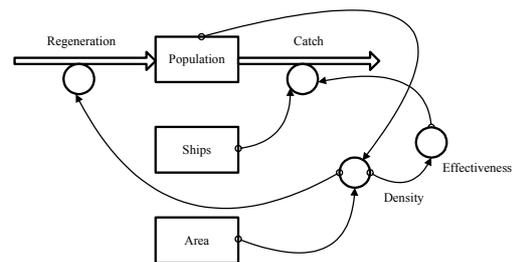


Figure 3: The Structure of Simulated Dynamic System

The structure in Figure 3 is drawn using the symbols that are evident to anyone familiar with system dynamics approach. The rectangles represent levels; the double arrows represent flows; the circles – decision functions; the thin arrows – information links. Model creation assumes fixed initial values for all levels, equations defined for all flows, links and decisions. Some specific information is needed, such as functions defining the relationships between fish density, and both ship effectiveness and fish regeneration. The functions are shown in Figures 4 and 5.

Simulation results of 3 trials are shown in Figure 6. The dynamic simulation results provide information about population dynamics and support decision making in management of renewable resources.

## STEPS OF SIMULATION STUDY USING SPREADSHEETS

Simulation studies independently of means of realization (by hand, using simulation software, etc.) include a typical set of steps. In this section only some of those steps are considered, such as input data

analysis, random variate generation and output data analysis. The steps are selected because they require special techniques in order to be realised on spreadsheets.

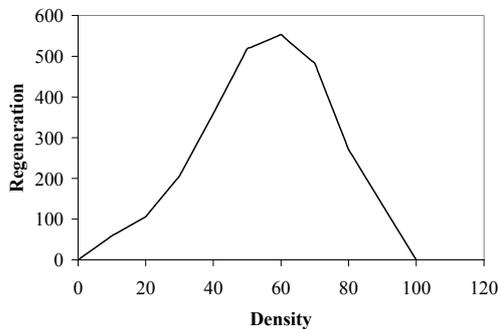


Figure 4: Fish Regeneration Function

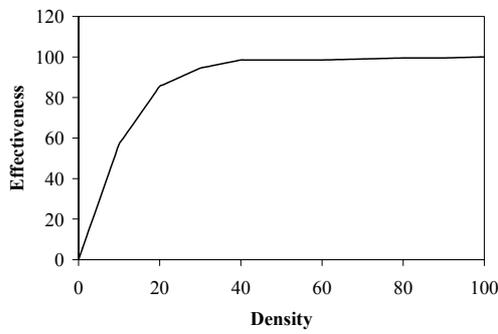


Figure 5: Ship Effectiveness Function

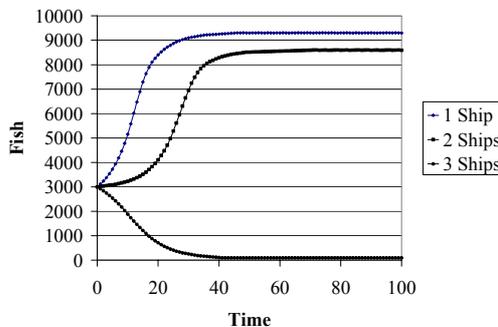


Figure 6: Simulated Population Dynamics

### Input Data Analysis

Input data for simulation models can be obtained from different sources. Measurements, observations and historical data that provide quantitative information for simulation should be “processed” before becoming a driving force of a simulation model. The tabular nature of empirical data nicely fits spreadsheets. There are alternative approaches to data processing. Standard spreadsheet functions, toolpacks, external data processing (e.g. with the help of Arena Input Analyzer or with special software), add-ins such as Crystal Ball or @Risk, and manual data processing provide different

possibilities. The standard spreadsheet functions assume reasonable input data processing.

Data processing includes identifying the distribution, distribution parameter estimation, and goodness-of-fit testing as the main stages. All these activities and calculations can be effectively performed on spreadsheets. It should be mentioned that in all cases the data is discrete.

As an example, an empirical distribution function for customer demand is obtained. The data about demand for mineral water from a small shop was collected. Data collection period was four months. Measurement units are bottles per day. Excel standard functions MIN, MAX, AVERAGE, STDEV and FREQUENCY were used. The resulting histogram is shown in Figure 7, sample parameter estimations are shown in Table 2.

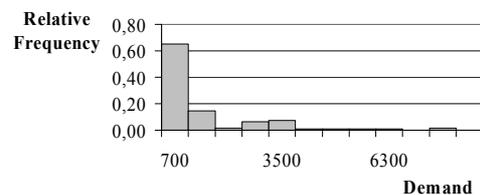


Figure 7: Histogram of the Daily Demand for Mineral Water

Table 2: Parameter Estimation for the Daily Demand for Mineral Water

Minimum	0.00
Maximum	7324.00
Average	907.23
Standard Deviation	1477.58

The chi-square test for distribution fitting for Weibull distribution was performed and the hypothesis about the random variable Weibull distribution was rejected. The Excel functions WEIBULL and CHINV were used for testing. The selection of distribution type based on the shape of the histogram was unsuccessful. Therefore, an empirical distribution was used for simulation.

### Random Variate Generation

A variety of probability distributions is available to generate random data for simulation. Three basic generation approaches in spreadsheets are applied. One can generate random variates using a number of standard distribution functions. The generated values are sampled from continuous or discrete distributions. The second approach is generating random variates from empirical distributions. Using only Excel built-ins gives us a possibility to get a new sample for each replication.

The third approach assumes the usage of Data Analysis Toolpack. Each reference to this tool will provide a sample. For spreadsheet simulation purposes this approach does not suit dynamic simulations.

The most specific for simulation purposes is random variate generation from continuous distributions. In this

case the distribution function should be presented in a tabulated form. In Excel we use the VLOOKUP function to generate random variates from empirical distributions. Using this function we can generate discrete outcomes defined by almost any probability mass function (using inverse transformation method). An example of this method application for generating and analysing data using a discrete triangular distribution function is considered. Figure 8 shows the Excel spreadsheet snapshot with distribution parameters and generation results, while in Figure 9 both appropriate cumulative distribution function and probability mass function are shown.

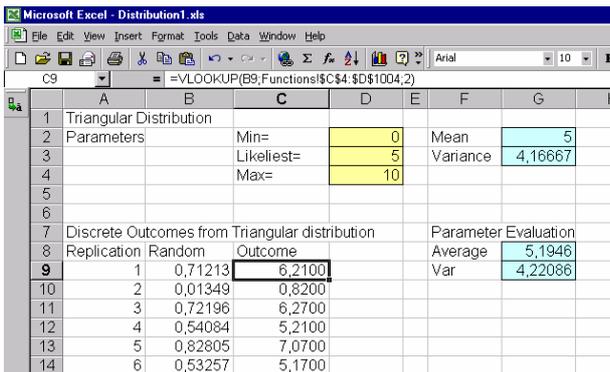


Figure 8: Discrete Distribution Parameters and Generated Outcomes

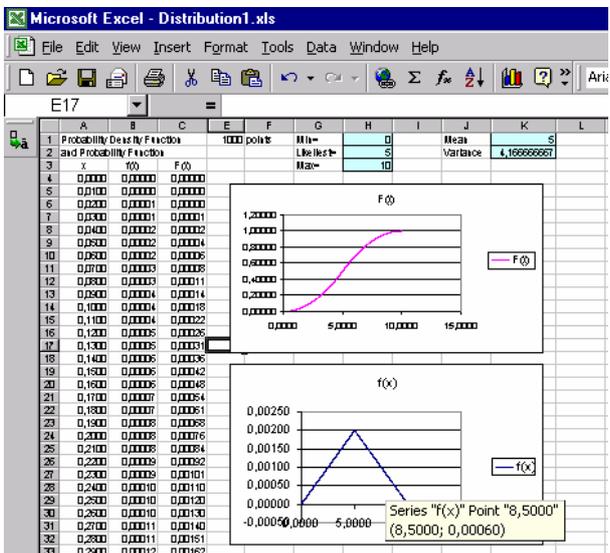


Figure 9: Cumulative Distribution Function F(x) and Probability Mass Function f(x)

The fitting Chi-square test for the obtained thousand points gives a result of  $\chi^2=0.673744$ , while the critical value for this sample is  $\chi_{crit}(0.05,26)=38.88513$ , where probability of rejecting a valid distribution is 0.05 and degrees of freedom correspond to the number of histogram cells. As  $\chi^2$  is less than a critical value, we fail to reject the hypothesis that the data originates from the hypothesized distribution.

The inverse transformation method applied to generate random variates from discrete probability distributions provides usage of empirical distributions for simulation purposes in spreadsheets, as well as gives an insight of using any distribution in simulation.

### Output Data Analysis

The simulation models that use random input data produce random results. Each simulation experiment or trial gives a unique set of random variables. To interpret results correctly the simulation experiments should be replicated, results should be accumulated and after performing a number of replications the statistical performance measures should be calculated. To obtain a necessary set of results the simulation model should be supplemented with an appropriate section. The realization and the appearance of such a section may vary. In Excel it is possible to perform replications not only using macros or add-ins. The choice of Tools\Options\Calculations\Manual option allows calculations using cross-references. Such calculation mode is useful for accumulating simulation results. Actually, the data analysis actions in spreadsheets are almost manual if not using additional tools. The estimation of the basic measures of performance supposes the application of standard functions and goodness-of-fit tests. These calculations are already overviewed in the above sections. The features of simulation model define the performance measure types. For example, the simulation results for simulation model from section **Spreadsheet discrete-event simulation models** shown in Table 3 are obtained after 100 replications.

Table 3: The 55-minute Simulation Results for Queuing System Model

Performance Measure	Resulting value
Total production	21
Average delay in queue	0.5
Server1 Utilization	0.8545
Server2 Utilization	0.6364
Maximum number of entities in queue	1
Maximum delay in queue	2.0
Maximum service time	6.0

Using add-ins for replication of simulation runs can sufficiently facilitate not only experimentation with the model. Add-ins provide convenient input and output data analysis as well as simplify random number generation for different purposes. The only disadvantage for teaching purposes is their “hidden” algorithms for performing replications and performance measure calculations.

## MODEL CONSTRUCTION AND SIMULATION RUNS

Model construction using spreadsheets substantially differs from the same process using simulation software. Actually, it looks more like a kind of programming. The flexibility and complexity of spreadsheet simulation models depends on author's skills. Some stages are typical for spreadsheet calculations:

- assignment of spreadsheet areas for data and parameters;
- programming of cell formulas;
- copying cell formulas to provide a number of trials or a necessary simulation length;
- visualisation by using charts.

The model implementation requires data accumulation for calculations of statistical performance measures. This goal is specific to spreadsheet simulation model construction. Using manual calculations as well as cross-references, it is possible to accumulate the results for further calculations. Advanced spreadsheet users can implement more complicated techniques by creating macros, using Add-ins e.g. Analysis ToolPack, Crystal Ball, etc. Crystal Ball Add-in can be used as an engine of dynamic simulation providing a necessary number of simulation experiments both for static and dynamic models.

## CONCLUSIONS

Spreadsheets provide broad possibilities of creating simulation models for educational purposes. They are useful for on-hand simulations in business games, case studies, engineering, etc. Spreadsheet models are suitable for logistics, business management, operations research and other studies as well as for implementing at small and medium-size enterprises. The basic spreadsheet features facilitate static simulations, and dynamic simulations can be rather effective and helpful in different areas of application. The spreadsheet simulation models can be used as a platform for understanding the mechanisms behind the discrete-event, as well as System Dynamics approaches. Simulation models developed on spreadsheets are not very flexible. But, from a teaching point of view, they have advantages of getting software skills in a short period of time, broad availability, ease of use, ease of validation, and low price (Robinson 2003). It is possible to implement both static and dynamic simulation using spreadsheet standard functions only. Macros and add-ins as simulation engines give an additional support to model construction and replication of simulation experiments. Understanding of simulation concepts by using spreadsheet simulation moulds a basis for application of simulation for different studies, for practical investigations, and for further shifting to special simulation software.

## REFERENCES

- Banks, J.; J. S. Carson; and B. L. Nelson. 1996. *Discrete-Event System Simulation (2nd Edition)*. Prentice Hall, N.J.
- Evans J. and D.L. Olson. 1998. *Introduction to Simulation and Risk Analysis*. Prentice Hall, N.J.
- Pecherska J. and Y. Merkurjev. 2001. "Monte Carlo Simulation on Spreadsheets". In *Scientific Proceedings of Riga Technical University*. RTU, Riga, 122-129.
- Robinson S. 2003. *Simulation. The Practice of Model Development and Use*. John Wiley & Sons, West Sussex, England.

## AUTHOR BIOGRAPHIES

**JELENA PECHERSKA** comes from Riga, Latvia. She works for Riga Technical University from 1979 and obtained her master degree there. At the moment she is working at the Department of Modelling and Simulation of Riga Technical University as a lecturer. Professional interests are in the fields of methodology of discrete-event simulation, supply chains modelling, practical applications of discrete-event simulation and discrete-event simulation in education.

**YURI MERKURYEV** is Habilitated Doctor of Engineering, Professor of the Institute of Information technology at Riga Technical University, Head of the Department of Modelling and Simulation. His professional interests include a methodology and practical implementation of discrete-event simulation, supply chain modelling and management, and education in the areas of modelling, simulation and logistics management. He is Programme Director of the Master-level curriculum "Industrial Logistics Management" at Riga Technical University. Prof Merkurjev has wide experiences in performing research and educational projects in the simulation area, at both national and European levels. He regularly participates in organising international conferences in the simulation area. He is the General Chair of the 2005 European Simulation Multiconference, SCS – ESM2005, to be held in June, 2005 in Riga, Latvia. Prof. Merkurjev has about 180 scientific publications, including 2 books. He is a Board member of the European Council of the Society for Modeling and Simulation International, President of the Latvian Simulation Society, Board member of the Latvian Transport Development and Education Association, and corresponding member of the Latvian Academy of Sciences.

# THE ROLE OF ONTOLOGIES IN AGENT-BASED SIMULATION OF INTELLIGENT TUTORING SYSTEMS

Vita Graudina  
Janis Grundspenkis  
Department of Systems Theory and Design  
Riga Technical University  
1 Kalku str., LV-1658, Latvia  
E-mail: {vita, jgrun}@cs.rtu.lv

## KEYWORDS

Ontologies, Agent-Based Simulation, Intelligent Tutoring Systems, Multi-Agent Systems.

## ABSTRACT

The paper presents a novel architecture of agent-based simulation of teaching and learning process. We propose a conceptual architecture of such system. There are presented several aspects, how the usage of ontologies for the control of students progress can improve the efficiency of intelligent tutoring systems. Main attention is paid to the usage of ontologies for agent communication and formal description of learning content and process.

## INTRODUCTION

The paper presents ongoing research on the role of ontologies in the development of the intelligent tutoring systems. The intelligent tutoring system should be considered as a testbed for agent-based teaching and learning process simulation. Technology based learning stands for all forms of computer supported learning, including distance, on-line, virtual learning, E-learning, etc., (Anohina 2003). A lot of work has been done to improve tutoring process based on E-learning systems (Paulsen 2003; Race 1994), Web-based learning environments (Groeneboer and Stockley 1997), and WebCT (available at <http://www.webct.com/>), etc.

Among wide varieties of applications of technology based learning, one of the most interesting and promising is the development of intelligent tutoring systems. Usually the research of these systems is focused on the development of such modules as Expert Model, Student Model, Course Management System, (Capuano et al. 2002) that are, as a rule, implemented in an intelligent tutoring system's architecture by using multi-agent based architecture (Capuano et al. 2000; Sklar et al. 2004; Dorca et al. 2003; Garro and Palopoli 2002).

Our search for the literature in the field reveals that the main activities in simulation of the teaching and learning process are done mainly in two directions. First, simulation of tutor and student activities is considered. So called pedagogical agents (see, for example, Johnson 2003; Dorca et al. 2003) simulate

tutor, while the student model is used for simulation of students actions (Greer and McCalla 1994). Second, there are attempts to simulate policies of education process (Sklar et al 2004).

To make experiments with real systems, it is necessary to develop a simulator of the system. So, our final goal is the implementation of a simulation system that will enable tutors to pose "what if" questions about the effects of their decisions. Experiments are needed to test different teaching methods, techniques and pedagogical approaches, different course material representation techniques and different sequences of course contents. Multi-agent architecture allows to emulate a human tutor and a student group, i.e., the interaction between agents in the hierarchically organized community, where each agent can only take a role of a tutor or a role of a student, i.e. the roles are mutually exclusive. It is the case when simulation of interactions should be used to evaluate the efficiency of delivering course materials and different ways of their visualisation. Our search for information sources on the usage of ontologies in intelligent tutoring systems shows that their role is underestimated. There are only few papers focused on the usage of ontologies in this field, and mainly they are about management of course materials and learning objects (Brace and Nejdil 2004; Baumann et al. 2002; Tane et al. 2003; Garro and Palopoli 2002).

Ontologies play major role in agent interaction by providing shared representation of the domain and the concepts that agents need to use (Sycara and Paolucci 2004; Esteva et al. 2002). The approach discussed in this paper is based on the assumption that in result of interactions during teaching and learning process agents should reach a common shared ontology. This is not the case at the starting point of tutoring when each student has his/her own domain ontology, which may pretty much differ from the tutor's ontology.

Usually ontologies are used to describe some teaching components of intelligent tutoring systems, like learning objects of the course (Capuano et al. 2002; Garro and Palopoli 2002), theories about learning (Meisel et al. 2003), management of personalized information (Weissenberg et al. 2004; Garro and Palopoli 2002).

The paper is an attempt to highlight the role of ontologies in the intelligent tutoring systems. We propose architecture of the system where course content

as well as interactions between agents are described by ontologies.

It is needed to point out that at the present phase of the research we take a rather simple approach where student's progress is related only with step by step construction of his/her ontology. The ontology represents the student's knowledge base, which includes the particular knowledge domain being learned. At the same time we agree with other researchers that cognitive (internal) factors such as motivation, emotion, and, ability to learn as well as external factors (such as tutor's personality, relationships with class mates, training appliances, environment, etc.) jointly affect teaching and learning process (Sklar et al. 2004). We suppose to include more factors in the simulation system in future.

## ARCHITECTURE

Agents play a dual role in simulation of teaching and learning process: on the one hand, they are goal-directed autonomous problem solvers; on the other hand, they have a social dimension because they interoperate as a part of a multi-agent system (MAS) (Weiss 1999; Wooldridge 2002). Ontologies describe the type of entities that agents encounter, their properties, and the relations between them. Agents in a MAS necessarily interact with other agents. Ontologies provide the basic representation to the agents that allows them to reason about interactions needed for the solution of the problem. Moreover, ontologies provide agents with shared knowledge that they can use to communicate and work together (Sycara and Paolucci 2004). This paper focuses on the modelling of ontologies that help agents in their interaction in the intelligent tutoring systems.

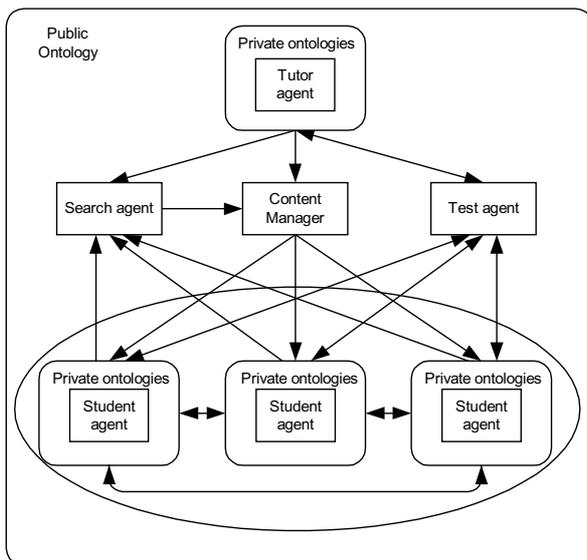


Figure 1: Architecture of the proposed system

The proposed model is a MAS, where agents interact to achieve the goal – perform students tutoring (Student agents' Domain ontology starts to align to one that is intended by Tutor agent). Agents of different types

constitute the system (see Figure 1), while ontologies of different types support interactions between agents.

## Agents

Different types of agents are used in the proposed MAS. First, Student agents, second, tutoring support agents (Search agent, Content Manager and Test agent), and third, the Tutor agent (see Figure 1).

Figure 2 shows all information and data flows between agents, which could be considered and called as agent perceptions. All flows between students' community and other agents are also possible for each student agent individually not only for the whole community.

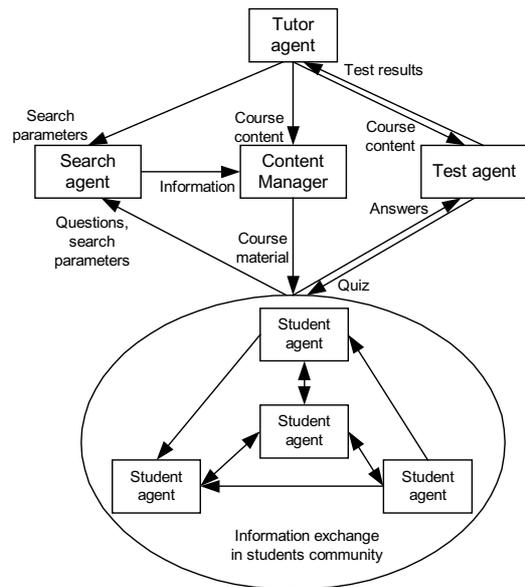


Figure 2: Perceptions of Agents

**Student agents** and the **Tutor agent** simulate the corresponding human beings that are considered to be knowledge workers, whose activities are supported by different agents surrounding them (Grundspenkis 2003). These agents usually are filtering agents, search agents, recommender agents and other personal agents. In our case we consider the Tutor agent and Student agents as knowledge workers, but tutoring support agents as personal agents. All agents have their own ontologies and can use shared ones. Student agents constitute a student community. During communication process Student agents change and update each other's Domain ontologies. Student agents can send some search parameters to Search agent, if they need some additional information about delivered course material.

The Tutor agent is used to simulate the tutor's work. This agent supports a tutoring process in proposed system. The Tutor agent coordinates interactions between other tutoring support agents to reach the tutoring goals by providing information about course contents, pedagogical approaches, etc.

The **Search agent** is needed to support Student agents with appropriate course material. According to given search parameters of Tutor agent and Student agents,

the Search agent derives necessary information to Student agents. In principle, to provide better search results the Search agent could interact with other search agents (like, Web crawlers - also known as Web spiders, robots, or wanderers (Chau and Chen 2003)) or database agents, but it needs more research on it.

The **Content Manager** supports representation of information found by the Search agent in the form, which is understandable and easy to use. The deliverable course material depends on course contents, tutor's teaching style and student's learning style. Only after a course material has been corrected, it is delivered to the Student agent.

The **Test agent** compares Student agent's Domain ontology with Tutor agent's Domain ontology, and returns results of comparison to the Tutor agent. Student agents receive a course material from the Content Manager, tests and quizzes from the Test agent, and give test answers to the Test agent.

### Ontologies

It is worth to underline that there are inherent difficulties encountered in implementing coordinated behaviour in any MAS communication, interaction, coherence and coordination (Capuano et al. 2000). These difficulties may be overcome using ontologies. Ontologies describe communication protocols, provide ways how agents can interact with each other and help agents to find solutions and to achieve their goals. According to dual agents' behaviour (Sycara and Paolucci 2004) ontologies of two types are needed. Private ontologies support the individual problem solving purposes of agents while Public ontologies support social interactions of agents.

Following (Guarino 1998), there are other subtypes of ontologies. Domain ontologies, Task ontologies and Application ontologies are Private ontologies, but Top-level ontologies are Public ontologies. Figure 3 shows the hierarchy of proposed ontologies. A Task ontology and a Domain ontology specify concepts from the Top-level ontology, and an Application ontology specifies the Domain ontology and the Task ontology.

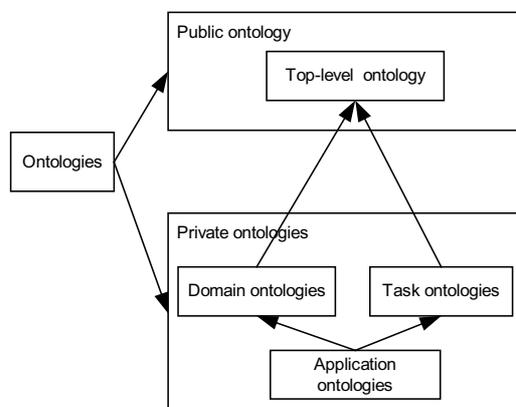


Figure 3: Hierarchy of Ontologies

A **Public ontology** is necessary to support social interactions between agents in the MAS. It supports communication and information exchange. It also provides description of the infrastructure of the whole MAS and involved agents. Involved agents are described in terms of their capabilities, interaction protocol, etc. Public ontology is shared between agents in the MAS. A **Top-level ontology** describes general terms used overall the problem domain, such as time, space, etc. It is possible to use some already created top-level ontologies, like WordNet (Miller 1995), Cyc (Lenat 1995), etc. Top-level ontology as a public ontology is also shared between all agents of the system.

**Private ontologies** are created for each agent individually, depending on the type of an agent, its purpose, goals, acting domain, actions etc. **Domain ontologies** and **Task ontologies** describe vocabulary that is connected with agents' domain (like, education, pedagogical approaches) or task, or actions (like, searching, planning or learning). These ontologies specify concepts from the Top-level ontology. Some agents in our proposal are domain independent, so they need only Task ontologies. For example, for a Content Manager we specify only a Task ontology, because this agent is domain independent and there are no needs to create a special Domain ontology. **Application ontologies** describe concepts depending on both a particular domain and a task (like, order of delivering course material or information about which lessons should be learned before particular one). The Application ontology also defines commands, parameter names and interpretation of them needed to control the system from outside. The Application ontology provides possibility to human intervention in agents' behaviour to make experiments. In some cases it isn't needed to use Application ontologies, it is enough with Domain and Task ontologies, like in case with a Test agent. This agent has only a Task ontology, because it provides concepts about different kinds of quizzes as well as concepts about process of testing.

A **Student agent's Domain ontology** describes the current state of Student agent's knowledge about the domain. This ontology is updated during communication between Student agents or after they have learned a course material provided by the Content Manager. For the Tutor agent's Domain ontology it is possible to use some already created ontologies (ACM Computer Classification System available on <http://www.acm.org/class/1998> can be used for courses in Computer Science or bodies of knowledge, for example, the Software engineering body of knowledge available on <http://www.swebok.org>).

Initially Student agents' Domain ontologies have some concepts associated with the course, but at the beginning of learning they may be different for each Student agent. An initial Student agent's Domain ontology could be empty or may present an initial ontology that is obtained by experience or intuitively. An empty initial ontology simulates those students who have no prerequisite knowledge about particular lecture

course. An empty ontology, as a rule, is for beginners' level courses, as we suppose that students don't have a prior knowledge.

An initial Domain ontology that includes only some concepts simulates students who have already some knowledge about the course, for example, if the offered course is "Advanced Artificial Intelligence", it is more possible that Student agents' Domain ontologies already have concepts from "Fundamentals of Artificial Intelligence" ontology. An initial ontology with some concepts is obtained from real experiments (tests) with real students.

Student agent's Domain ontology is updated after delivering of each portion of course material. The Test agent compares Student agents' Domain ontologies with Domain ontology provided by the Tutor agent. Depending on results of testing the next portion of the course material is delivered.

Candidate concepts are dealt out from the delivered course material by using ontology learning methods and approaches (Maedche and Staab 2004). Candidate concepts are concepts, which could be included in the Student agent's Domain ontology in case of successful learning of delivered material. A Test agent is used to find candidate concepts. By using quizzes it is tested has student learned candidate concepts or not. If he/she has, then candidate concepts are included in the Student agent's Domain ontology in accordance with following rule:  $Ontology^{old} \cup Learned\_concepts \rightarrow Ontology^{new}$ .

The question is still open how to simulate the outcome of student communication that may result in learning new concepts. If it is the case, we somehow need to update the Student agents' Domain ontologies. At the moment it is unclear what kind of agents and methods can be used.

A **Student agent's Task ontology** is used to describe concepts corresponding to a learning process as well as generic concepts about learning styles. A Student agent's Task ontology is a basis for further simulation of students learning styles and preferences.

A **Student agent's Application ontology** can be considered as a complete Student model, because it contains student knowledge level as well as preferred learning style. It is used for the same purpose as a student model in the intelligent tutoring systems (Capuano et al. 2000). A Student agent's Application ontology stores and describes characteristics, preferences, learning style and problem-solving behaviour of the particular student. Concepts of the Student agent's Application ontology are updated after Student agent's action. The Student agent's Application ontology influences the Content Manager agent with concepts about particular student's preferences and learning style.

A **Tutor agent's Domain ontology** contains ontology of the offered course, i.e. it can be considered as the ontology of the course. The aim of whole teaching process is to update Student agents' Domain ontologies until they match the Tutor agent's Domain ontology.

A **Tutor agent's Task ontology** contains concepts

about the order of course material delivering, teaching goals, teaching techniques, methods and styles.

A **Tutor agent's Application ontology** similarly as a Student agent's Application ontology is used to provide capability of making experiments with different pedagogical approaches. A Tutor agent's Task ontology together with a Tutor agent's Domain ontology constitutes complete Tutor model. Therefore, we don't need a specific Tutor agent's Application ontology, as it is in case of a Student's agent.

A **Search agent's Domain ontology** describes the concepts about possible area where an agent can search additional course materials. This ontology contains concepts, like Internet, intranet, available databases, etc.

A **Search agent's Task ontology** describes process and methods of searching for additional course materials.

A **Search agent's Application ontology** describes the interpretation of the received search parameters from Student agents and a Tutor agent. The Application ontology also specifies concepts introduced in the Search agent's Domain ontology and the Search agent's Task ontology.

A **Content Manager's Task ontology** describes concepts about process and techniques (text, slides, diagrams, pictures, etc.) of information visualisation. Forms of visualisation are defined by methodological considerations depending on specific goals and requirements of each course. It is also desirable that at least part of the course material is available in several forms of visualisation because students have different priorities of using them due to their different background knowledge and abilities to perceive materials (somebody prefers full text, somebody prefers "condensed" text presented in slides, while others prefer more diagrams, picture, etc.). Students' preferences are collected during students' actions and they are stored in the Student agents' Application ontologies.

A **Content Manager's Application ontology** describes the interpretation of a Student agent's Application ontology and a Tutor agent's Application ontology. These ontologies have concepts about student's learning style and preferences, tutor's teaching style, and visualisation concepts from the Contents Manager's Task ontology.

A **Test agent's Task ontology** has concepts about different kinds of quizzes (Race 1994), which can be used to test whether a student has learned the delivered course material. It also describes process of making different kinds of quizzes to provide testing of candidate concepts.

## RELATED WORK

Our search for related works confirms that the proposed architecture represents a novel approach because we have not find exactly similar approaches. The previous works can be divided in some groups, which only to the certain extent can be considered to be similar with our conception (some parts of them are similar with our approach). These similar works can be divided into three groups: works related to ontologies, works related

to usage of agent technologies, and works related to simulation in education.

◆ Related works on ontologies can be divided as follows:

- e-Learning, where ontologies are used to support management and retrieval of the course materials (Brace and Nejd1 2004; Baumann et al. 2002; Tane et al. 2003), as well as visualisation of delivered course material (Abecker and van Elst 2004));
- Agent technologies, where ontologies are used to support communication, coordination, interactions and information exchange between agents in MAS (Sycara and Paolucci 2004).

◆ Related works in the field of agents:

- Agents in education (information search, retrieval and representation, and different pedagogical agents for tutoring different learning courses (Zhong et al. 2003, Capuano et al. 2000; Dorca et al. 2003; Garro and Palopoli 2002);
- Agent-based simulation for different purposes (for example, modelling of different decision making (Sklar et al. 2004) or industrial processes).

◆ Related works in the field of simulation in education:

- Agent-based simulation in education (we have found only SimEd for simulation and modelling of search for the optimal educational policy (Sklar et al. 2004);
- Simulators, which are used for training some specific skills (like, simulators in aviation for pilots trainings or manufacture workers (Ho and Vance 1995)).

Figure 4 shows overlapping of four fields (agents, simulation, education and ontologies) to illustrate connections between them. It shows related works and position of our proposal in these areas.

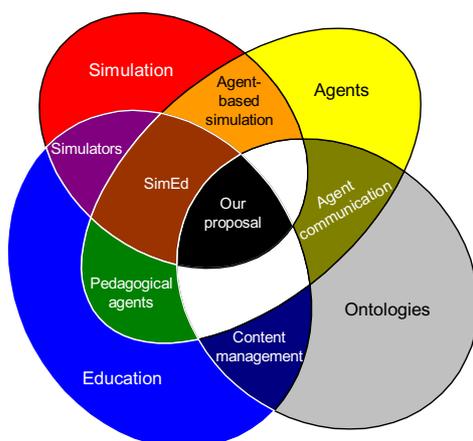


Figure 4: Areas of related works

The presented work differs from related works because specific interactions in specific knowledge domains are simulated using different kinds of ontologies. Our approach integrates these ontologies into multi-agent

system. The long-term goal of this research is to simulate on the individual bases the interactions that take place between the tutor and the group of students. We hope that this approach will allow tutors to experiment with their decisions and their impact on the teaching and learning process considering teaching methods, sequence of topics, course material representation techniques, assessment methods of student progress, etc.

## CONCLUSIONS AND FUTURE WORK

In this paper a novel approach of multi-agent systems architecture based on ontologies is introduced. The proposed system should be considered as a testbed for agent-based simulation of teaching and learning process. The role of ontologies is described and ways how ontologies may be used in intelligent tutoring systems are discussed. This is only a initial phase of the development of the agent-based simulation of intelligent tutoring system in which different ontologies are used. This paper presents a conceptual level of the proposed approach.

Future work is connected with the development of a prototype of the intelligent tutoring system based on the proposed approach. After that the experimental campaign is planned. Further studies will concern the architecture and models of the proposed system. They will be defined in details after the end of the experimentation and analysis of results obtained. We are also planning to make experiments how to obtain students' initial ontologies more effectively and to work on how to prepare tests for establishing concepts and relations between concepts for students' initial ontology.

## REFERENCES

- Abecker, A. and L. van Elst. 2004. "Ontologies for Knowledge management". In *Handbook on Ontologies 2004*, S. Staab and R. Studer (Eds.). Springer-Verlag, 435-454.
- Anohina, A. 2003. "Classification of the Terminology Used in the Field of Virtual Learning". In *Scientific Proceedings of Riga Technical University, 5<sup>th</sup> Series, Computer Science, Applied Computer Systems, V. 17*. Riga, RTU, 94-102.
- Baumann, S.; A. Dengel; M. Junker; and T. Kieninger. 2002. Combining Ontologies and Document Retrieval Techniques: A Case Study for an E-Learning Scenario. In *Proceedings of the 13th International Workshop on Database and Expert Systems Applications (DEXA'02)*.
- Brace, J. and W. Nejd1. 2004. "Ontologies and Metadata for eLearning". In *Handbook on Ontologies 2004*, S. Staab and R. Studer (Eds.). Springer-Verlag, 555-573.
- Capuano, N.; M. De Santo; M. Marsella; M. Molinara; and S. Salerno. 2000. "A Multi-Agent Architecture for Intelligent Tutoring". In *Proceedings of the International Conference on Advances in Infrastructure for Electronic Business, Science, and Education on the Internet SSGRR 2000*.
- Capuano, N.; M. Gaeta; A. Micarelli; and E. Sanginetto. 2002. "An Integrated Architecture for Automatic Course Generation". In *Proceedings of IEEE International Conference on Advanced Learning Technologies (ICALT 2002)*, 322-326.

- Chau, M. and H. Chen. 2003. "Personalized and Focused Web Spiders". In *Web Intelligence 2003*, N. Zhong; J. Liu; and Y.Y. Yao. (Eds.). Springer-Verlag, 197-217.
- Dorca, F.A.; C.R. Lopes; and M.A. Fernandes. 2003. "A Multiagent Architecture for Distance Education Systems". In *Proceedings of the 3rd IEEE International Conference on Advanced Learning Technologies (ICALT'03)*, 368-369.
- Esteva, M.; D. de la Cruz; and C. Sierra. 2002. "ISLANDER: an electronic institutions editor". In *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems (AA-MAS 2002)*. (Bologna, Italy, July). 1045-1052.
- Garro, A. and L. Palopoli. 2002. "An XML Multi-Agent System for e-Learning and Skill Management". In *Proceedings of the 3rd International Symposium on Multi-Agent Systems, Large Complex Systems, and E-Businesses (MALCEB'2002)* (Erfurt, Germany, October), 283-294.
- Greer, J.E. and G.I. McCalla (Eds.). 1994. *Student Models: The Key to Individualized Educational Systems*. Springer Verlag.
- Groeneboer, T.C.C. and D. Stockley. 1997. "Virtual-u: A collaborative model for online learning environments". In *Proceedings of the Second International Conference on Computer Support for Collaborative Learning* (Toronto, Canada, December).
- Grundspenkis, J. 2003. "Development of Hybrid Intelligent Systems: Integration of Structural Modelling, Intelligent Agents and Knowledge Management Techniques". In *Scientific Proceedings of Riga Technical University, 5<sup>th</sup> Series, Computer Science, Applied Computer Systems, V. 17*. Riga, RTU, 7-30.
- Guarino, N. 1998. "Formal Ontology and Information Systems". In *Proceedings of Formal Ontologies in Information Systems 1998* (Trento, Italy, June). N. Guarino (Eds.). IOS Press, 3-15.
- Ho, L.K. and J.M. Vance. 1995. "Virtual Training in Manufacturing Systems Using TELEGRIP". In *Deneb User Group IX Meeting Proceedings*. (Ypsilanti, MI, October).
- Johnson, W.L. 2003. "Using Agent Technology to Improve the Quality of Web-Based Education". In *Web Intelligence 2003*, N. Zhong; J. Liu; and Y.Y. Yao. (Eds.). Springer-Verlag, 77-101.
- Lenat, D.B. 1995. "CYC: a large-scale investment in knowledge infrastructure". *Communications of the ACM* 38 No.11 (Nov), 32-38.
- Maedche, A. and S. Staab. 2004. "Ontology Learning". In *Handbook on Ontologies 2004*, S. Staab and R. Studer (Eds.). Springer-Verlag, 173-190.
- Meisel, H.; E. Compatangelo; and A. Hoerfurter. 2003. "An ontology-based approach to intelligent Instructional Design support". In *Proceedings of the 7th International Conference on Knowledge-Based Intelligent Information & Engineering Systems (KES'2003)*, volume 2773 of LNAI, Springer-Verlag, 898-905.
- Miller, G.A. 1995. "WordNet: A Lexical Database for English". *Communications of ACM* 38 No.11 (Nov), 39-41.
- Paulsen, M.F. 2003. "Online education. Learning Management Systems". NKI Forlaget, Norway.
- Race, P. 1994. "The Open Learning Handbook: Promoting quality in designing and delivering flexible learning", Kogan Page Ltd, London, Nichols Publishing Company, New Jersey.
- Sklar, E.; M. Davies; and M.S.T. Co. 2004. "SimEd: Simulating Education as a Multi Agent System". In *Proceedings of Third International Conference of Autonomous Agents and Multi Agent Systems (AAMAS-2004)*. 998-1005.
- Sycara, K. and M. Paolucci. 2004. "Ontologies in Agent Architectures". In *Handbook on Ontologies 2004*, S. Staab and R. Studer (Eds.). Springer-Verlag, 343-363.
- Tane, J.; C. Schmitz; G. Stumme; S. Staab; and R. Studer. 2003. "The Courseware Watchdog: an Ontology-based tool for finding and organizing learning material". Fachtagung Mobiles Lernen und Forschen, 6.11.2003, Universität Kassel.
- Weiss, G. (Eds.). 1999. *Multiagent Systems: A Modern Approach to Distributed Modern Approach to Artificial Intelligence*. The MIT Press.
- Weissenberg, N.; A. Voisard; and R. Gartmann. 2004. "Using Ontologies in Personalized Mobile Applications". In *Proceedings of the 12th annual ACM international workshop on Geographic information systems* (Washington, DC, USA, Nov. 12-13), 2-11.
- Wooldridge, M.J. 2002. *An Introduction to Multi - Agent Systems*. John Wiley & Sons.

## AUTHOR BIOGRAPHIES

**Vita Graudina** is the Master student at Riga Technical University. She is working on the final version of her Master thesis. The topic of the thesis is the usage of ontologies in different information and knowledge-based systems. Contact her Riga Technical University, 1 Kalku str., LV-1658, Latvia; vita@cs.rtu.lv.

**Janis Grundspenkis** is the professor at Riga Technical University. He also is the dean of the Faculty of Computer Science and Information Technology, the director of the Institute of Applied Computed Systems, and the head of the Department of Systems Theory and Design. His research interests are agent technologies, knowledge engineering and management, and structural modelling. He has published more than 150 scientific papers. Contact him Riga Technical University, 1 Kalku str., LV-1658, Latvia; jgrun@cs.rtu.lv.

# AN ACTIVITY ORIENTED VISUAL MODELLING LANGUAGE WITH AUTOMATIC TRANSLATION TO DIFFERENT PARADIGMS

Luís M. Silva Dias  
A. J. M. Guimarães Rodrigues  
Guilherme A. B. Pereira

Department of Production and Systems  
University of Minho  
Braga, 4710-057, PORTUGAL  
<<http://www.dps.uminho.pt>>  
< {lsd, agr, gui}@dps.uminho.pt >  
Phone: +351253604740/+351936271733 Fax: +351253604741

## KEYWORDS

Visual Simulation Modelling, Activity Cycle Diagrams, Automatic Programs Generation, Arena

## ABSTRACT

The traditional approach for discrete event simulation modelling includes visual support diagrams for modeller-client communication purposes (model interpretation and validation) and also to act as the basis for simulation language program construction.

Although modern simulation packages use powerful graphical interfaces for programming and animation purposes, these packages still require enormous simulation expertise to construct a simulation program. This work suggests the use of the Activity Cycle Diagrams-ACD (activity based philosophy) concepts for modeller-client communication, but also to act as an automatic generator of simulation programs under different paradigms - event scheduling (Basic Simulation Facility – Simulation Library) and Process Flow (ARENA – Simulation Environment) philosophies, thus eliminating any programming effort and expertise.

## 1 INTRODUCTION

The use of visual support diagrams to help the programming step of a simulation project is very common. Even when generic programming languages were replaced by specific purpose simulation languages the use of paper diagrams remained as a previous step to programming (Tocher 1963) (Pidd 1984). These diagrams were conveniently abstracted serving as support to the communication between the simulation client and the modeller (simulation expert), but also helping the construction of the corresponding computational programs (Clementson 1982) (Rodrigues 1987).

Modern simulation languages introduced new powerful graphical interfaces, but these interfaces are clearly programmer oriented, raising the difficulty in communicating with the client and still requiring enormous simulation expertise to use them (Dias and Rodrigues 2002b) (Harrell et al. 2003) (Kelton et al. 2004).

In this paper, we still suggest the use of a (simple) graphical support as a representation of what the client formally needs, but these diagrams will also act as the source to the automatic generation of simulation programs.

The visual language chosen is the Activities Cycle Diagrams (ACD), for its simplicity and efficiency in representing real operating systems. (Dias and Rodrigues 2002b) (Pidd 1990).

This mechanism also implied the construction of translation grammars. These grammars were written according to a modular specification of visual languages, based on attribute grammars (Henriques 1992) in MASOVILa (Modular Attribute-based Specification Of Visual Languages) (Rocha 1995) (Dias 1997) (Varanda et al. 1997). Our translation engine uses a pattern matching rewriting mechanism.

Using Activity philosophy for modelling, then generating simulation programs based on event philosophy and in process philosophy, three major simulation approaches (Bennett 1995) (Sargent 2004) (Overstreet 2004) were explored and linked.

## 2 LAYOUT AND ANIMATION ORIENTED SIMULATION ENVIRONMENTS

As already referred, appropriate diagrams were in use for many years to support the communication among people interested in a particular simulation. The simulation expert would then translate these models into a simulation language or even a general purpose programming language. As far as graphical support became available, an enormous variety of simulation environments emerged (Swain 1991-2003). **Graphical facilities** were then used to essentially represent a **system layout for animation purposes**.

Animation is recognized as an important aspect of simulation. However **when** the modelling process is **focused on animation**, several disadvantages may arise:

1. The **model may be overwhelmed** by many modules and accessory configurations.
2. Such a model, with increased complexity, will be **difficult for the client** to understand.
3. The analysis of a static model (in the first stages) based on the layout will not add much over its photograph or scheme. Thus the **semantic validation** will be left for the **animation** phase.

## 3 PROJECT MAIN STEPS

The main steps of this research work are summarized below:

1. The **choice of an easy** to use and widely spread visual language : Activity Cycle Diagrams (**ACD**).
2. **Formalization of ACDs** (keeping it simple), allocating to each graphical object the information required by the model.
3. Specification of a **file format** to represent the referred graphical objects (XML).
4. Implementation of a **graphical editor** to draw the models.
5. Implementation of a compiler's **compiler** for visual/graphical 2D languages.
6. Specification of a **grammar** for the ACD language.
7. Implementation of **two compilers** - the first compiler generates the simulation program code in **JavaBSF**, and the other one generates the **Arena** program (using both Arena modules and new developed modules in VBA code).

## 3.1 Programming Tools

The main programming language used was **Java** (Martins 1998) (Campione et al. 1999). Several sets of classes were implemented (corresponding to about nine thousand lines of code). Visual Basic for Applications (**VBA**) was also used, both in ARENA and in Microsoft VISIO. A template was built in **ARENA** (Professional Edition) (Kelton et al. 2004) with activities and queues to implement an activity-based executive on a process-oriented environment.

## 3.2 About the use of Graphics in Visual Modelling and Programming

The idea behind this work is to enhance the **utilization of graphical facilities in modelling**, making it a great contribution to **automatic generation of simulation programs**, keeping it simple and portable.

Graphical facilities are more helpful when they **support model semantic** than when it is based on system layout and animation.

Furthermore the utilization of a **simple** axiomatic set may be **accessible** to more potential users and clients.

The use of an **activity-based philosophy** seems to be semantically richer and more simulation oriented than process flow or event scheduling (Pidd 2004).

The strategy was to create a completely **open system**, since the graphical editor creates a text file in XML (eXtended Markup Language). This is compatible with any graphical editor using the established syntax. The compiler uses that XML file and a grammar to build a program in an object simulation language.

## 3.3 Translation

Our first **ACDs compiler** generates a Java program according to the **event scheduling** philosophy. Although this is computationally highly efficient it is harder to program. In other words we could say that it is more computer friendly than programmer friendly. The complex compiler developed allows an easier translation by the computer.

By developing a template with new blocks for **Arena (process oriented)**, we were able to simulate ACDs using Arena objects built according to an activity based philosophy.

Using these two compilers/translators, we explored deeply on three major modelling philosophies (Rodrigues 1987) (Pidd 1990) (Overstreet 2004), making automatic translation mechanisms between them.

#### 4 PROJECT MAIN TASKS

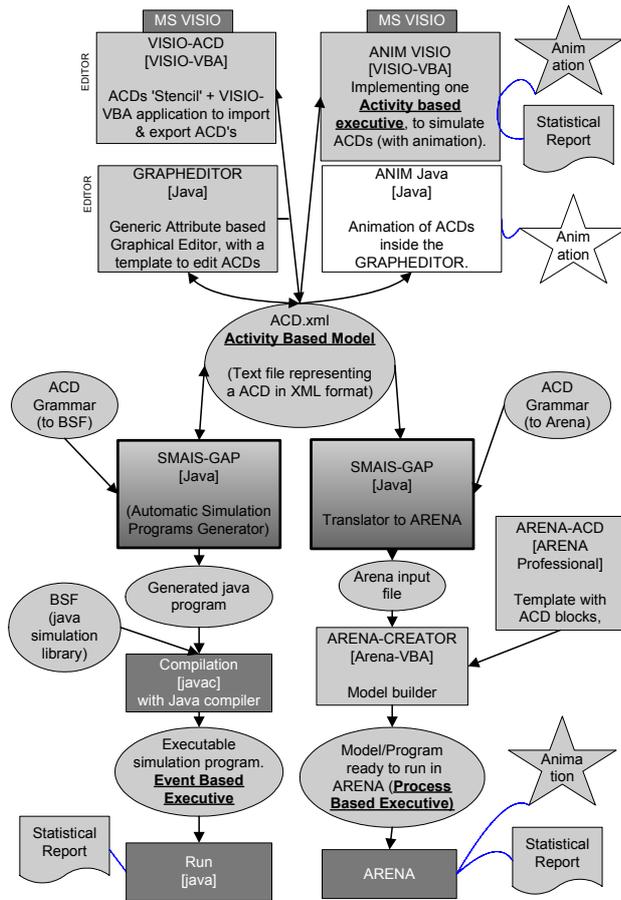


Figure 1: Project Map Diagram

The above diagram illustrates the interdependencies between the main tools and contributions of our project:

■ **ACD.xml** «*in the centre of diagram*» represents the chosen format to physically support the models: a text file in XML. (See also Figure 13 and Figure 14)

☞ **GRAPHEDITOR** «*upper left zone of the diagram*» is the graphical editor that was specifically built for this purpose (see also Figure 9). Microsoft Visio was also customised to deal with ACD XML files – **VISIO-ACD**.

➤ **ANIM VISIO** - «*upper right zone*» simulates and animates ACDs inside MS VISIO. The **ANIM-Java** tool will animate ACDs in the GRAPHEDITOR (is not yet completed, as signalled in the diagram by white background).

☛ **SMAIS-GAP** is the major tool of this project «*lower left section*» allowing the translation of an ACD (activity-based) into a program (event-based) (see also Figure 15, Figure 16 and Figure 17). It uses the **ACD Grammar (to BSF)** (see also Figure 10, Figure 11 and Figure 12).

☛ **SMAIS-GAP** using the **ACD Grammar (to Arena)** «*lower right section*» refers to the translation to **Arena input file**. **ARENA-CREATOR**, using that input file, builds a model with blocks from **ARENA-ACD** template (see also Figure 18, Figure 19, Figure 20 and Figure 21).

#### 5 ACTIVITY CYCLE DIAGRAMS (ACD)

ACDs were widely used mainly with older languages (e.g. HOCUS, ECSL (Clementson 1982)) to schematically specify the system's behaviour, through each assumed entity cycle diagram. These entity cycles explicitly refer the active states into **activities** and passive states into **queues**. This graphical language just requires the use of three types of graphical objects: **Rectangles** (activities), **Circles** (queues) and **Arrows** (links).

**For an activity to start**, it is necessary that entities exist in the preceding queues in the required number and with the adequate attributes. When these conditions hold, it is possible to start the activity. When the activity ends, the entities involved are moved to consequent queues.

The **complete model** consists of the Activity Cycle Diagrams of all the classes of entities, together. Interactions between entities take place at activities. Figure 2 represents the basic activity concept, with one activity in the middle that starts when each precedent queue has one required entity. When the activity ends, the entities are moved to the consequent queues.

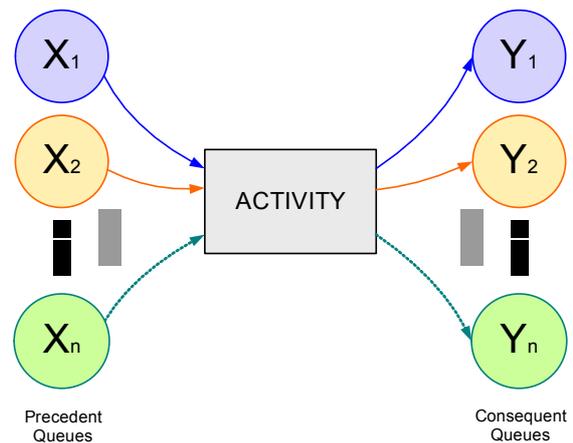


Figure 2: Activity concept

The simplicity inherent to the activity concept and to the ACD facilitates its easy understanding for **validation and teaching purposes**. It has been advocated that the ACD is useful for **research** discrete event simulation studies.

### 5.1 ACD Example: the Bartender Problem

In this illustrative example we have a barman that serves customers in a bartender (Clementson 1982) (Rodrigues 1987).

Entities of class **CUSTOMER**, are initially OUTSIDE. They ARRIVE and then WAIT for activity POUR. When served they are READY to DRINK. After that, if they NEED to drink more they go to queue WAIT, otherwise they leave to OUTSIDE. The ACD of CUSTOMER is described below (Figure 3).

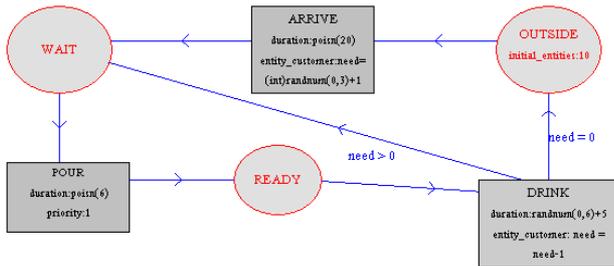


Figure 3: CUSTOMER ACD

Entities of class **BARMAN**, are waiting in the queue IDLE and they can either participate in the activity POUR or WASH:

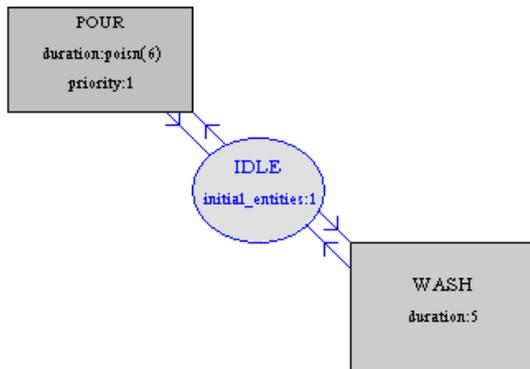


Figure 4: BARMAN ACD

Entities of class **GLASS** are initially in the queue DIRTY. After being WASHed (in batches of size 3), they wait in the queue CLEAN. When there are 1 customer waiting, 1 barman idle and 1 glass clean, then POUR activity begins. Once FULL, the glass goes to activity DRINK. The ACD of GLASS is in Figure 5.

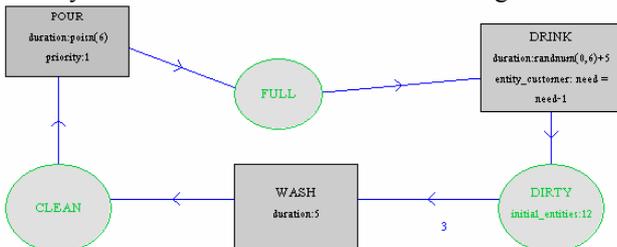


Figure 5: GLASS ACD

*This problem instance parameters are:*

*Activities duration:*

- ARRIVE: Poisson distribution, average=20.
- POUR: Poisson, average=6 [ poisn(6) ].
- WASH: Fixed = 5.
- DRINK: 5 + Uniform distribution between 0 and 6 [5+randnum(0,6)].

*Entities initial allocation:*

- CUSTOMER: 20 in queue OUTSIDE.
- BARMAN: 1 in queue IDLE.
- GLASS: 12 in queue DIRTY.

*Entities class setup:*



Figure 6: ENTITIES setup

- CUSTOMER: have one attribute: NEED.

*Attributes:*

- ARRIVE: Customer attribute NEED is initialized with : (int)randnum(0,3) + 1 → {1,2,3,4}
- DRINK: Customer attribute NEED is decremented. After this activity, customer attribute NEED is evaluated to decide customers destination.

*Simulation setup:*

- DURATION: 1000 time units.
- WARM\_UP: 120 time units.
- SEED: 123543.



Figure 7: Simulation setup

The following ACD (Figure 8) includes all system:

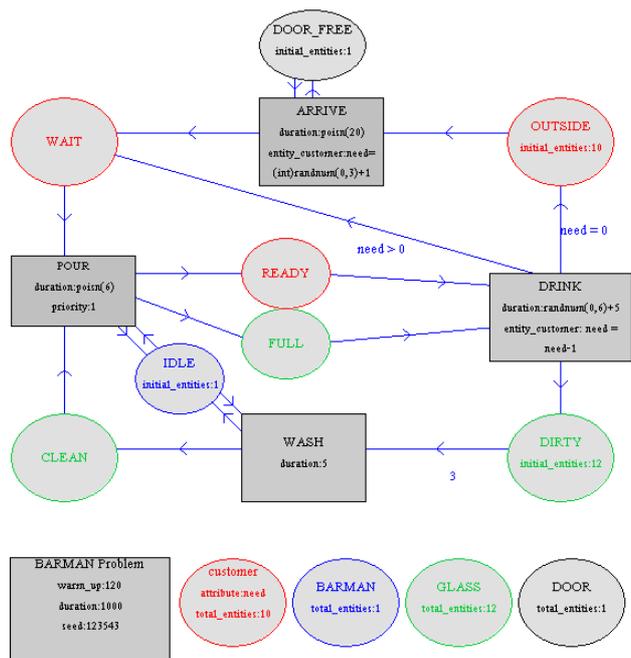


Figure 8: Global ACD

DOOR is an auxiliary entity used to control the Customer's arrives (one at a time). Described in (Rodrigues 1987).

The following image (Figure 9) shows a screenshot of the GRAPHEDITOR, editing the bartender problem.

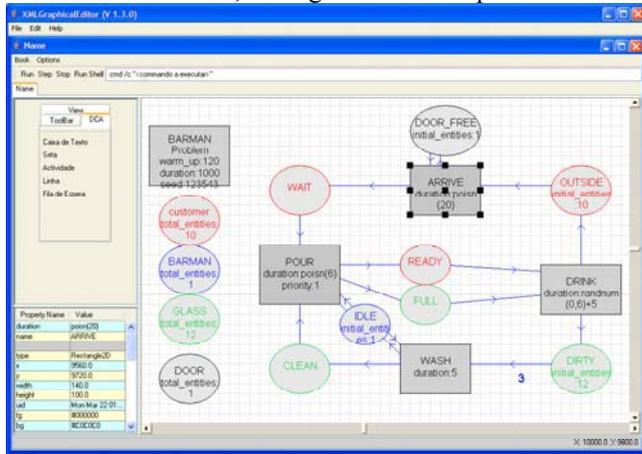


Figure 9 GRAPHEDITOR Screenshot

## 5.2 ACD Language Formalization

Some authors suggest modifications in ACD language, e.g. (Filho and Hirata 2004), but in our opinion that compromises the ACD simplicity.

A significant contribution of our work consisted on the development of the ACD language formalization embedding in the diagrams all the information required for the simulation.

1) We added attributes to the visual objects:

- ‘**duration**’ and [‘**priority**’] in **activities**. (‘priority’ is a value setting the activity priority over other activities. Higher value means higher priority).

- Entities **attributes** changing in activities. (EX: In the DRINK activity of the example model, the customer ‘need’ attribute is decremented by one. (activity attribute ‘entity\_customer’ = “need=need-1”).

- [‘**initial\_entities**’] in **queues**.

2) In order to include global information on the simulation a rectangle alone (called **simulation setup**) is used with the following attributes :

- ‘**model\_name**’ – Model\_Name\_String
- ‘**duration**’ - **Simulation time**
- ‘**warm-up**’ - initialization time.

3) For global attributes and features of each **entity**, one ellipse alone is created with:

- ‘**entity\_name**’ – name of the entity
- ‘**total\_entities**’ – number of total entities of this kind that will exist in the simulation.

- [‘**attribute**’]\* - used zero or more times to declare entity attributes. (EX: customers have one attribute: ‘need’).

- [‘**sort**’] –expression establishing the queue sorting rule. Value is defaulted to 1. If defined, the queue will not be FIFO, this attribute must have an expression, elements are sorted in ascending order based on the evaluated expression over each arriving entity. May be used to create virtually, multi-queues.

## 4) Arrows

- Arrow ending on activities:

‘**label**’ have the number of need entities (entering throw this arrow in the activity) to start the activity (this implements batches). Default value is 1 (if omitted). (EX: three is the number of ‘glasses’ to start ‘WASH’).

- Arrow leaving from activities:

‘**label**’ have conditions to decide which destination queue will be chosen (usually based on attributes values). (EX: ‘need’ attribute is used to decide if the ‘customer’ go OUTSIDE or WAITING after drinking).

## 6 THE GRAMMAR AND TRANSLATION

To create the AIMS compiler, we wrote a set of rules in Visual MASOVila notation (Dias 1997). Each rule synthesizes one new symbol. We developed rules for *queue* (3), for *activity* (9), for *entity* (3), for *input\_link* (1), for *output\_link* (1), for *simulation* (5), and also for syntactical and semantic *error* detection (10) – see graph in Figure 12.

We include bellow one expression example, with textual explanation, visual representation and the generated Java code (portion of the compiler code).

The next graph (Figure 10) is the rule that transforms an *arrow* in an *output\_link*, when it is connected from an *activity* to a *queue*. The new *output\_link* symbol, receives all attributes from *arrow* symbol (\*). Furthermore it synthesizes the *output\_link*'s attribute *origin* from the *activity*'s attribute *name*, the attribute *destination* from the *queue*'s attribute *name* and also the *condition* attribute from *arrow*'s *label*.

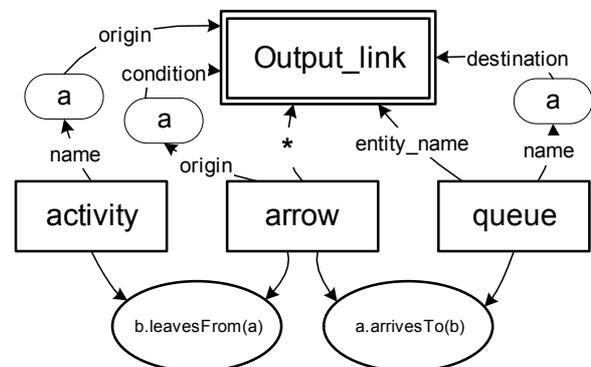


Figure 10: Output\_link rule written in Visual MASOVila

```
public static Symbol output_link(Vector args){
    Symbol activity = (Symbol) args.get(0);
    Symbol arrow = (Symbol) args.get(1);
    Symbol queue = (Symbol) args.get(2);
    if(((Arrow) (arrow.get("container")))
        .leavesFrom((Container)
            (activity.get("container"))) ) &&
        ((Arrow) (arrow.get("container")))
            .arrivesTo((Container)
                (queue.get("container"))))
        {Symbol res = new Symbol(arrow);
        res.put("origin", activity.get("name"));
        res.put("destination", queue.get("name"));
        res.put("condition", queue.get("label"));
        res.put("entity_name", queue.get("entity_name"));
        return res;
        }else return null;
} //& output_link
```

Figure 11: Java code for Output\_link rule.

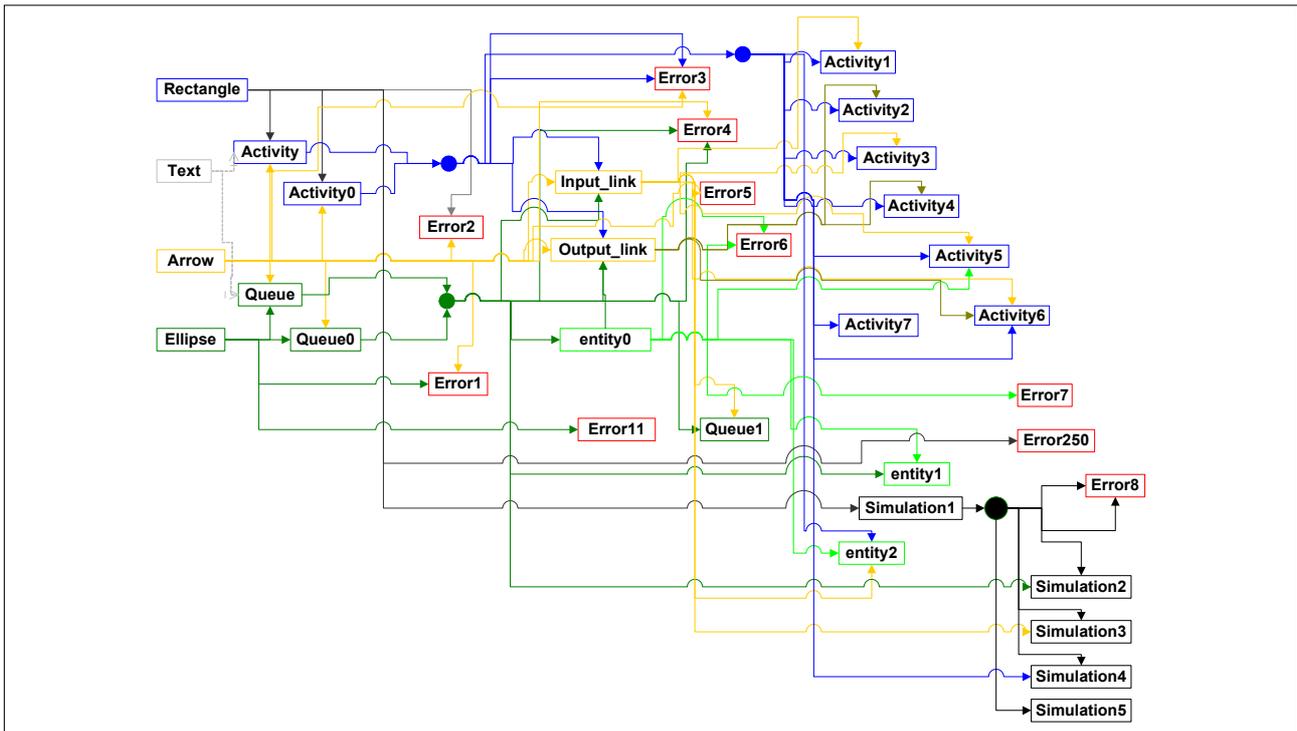


Figure 12: Graph with the grammar rules interfaces, and interdependencies

The Output\_link rule is just one example that can be found in the central region of Figure 12:

The translation is archived by an engine (the SMAIS-GAP) that successively searches patterns in the Diagram under analysis to try to apply each grammar rule (in a specific order). Applying a rule, means rewriting or creating one symbol in the Diagram (some rules may also remove symbols from the diagram).

To the Output\_link rule, the engine must know that this rule needs: one activity, one arrow and one queue. The engine then picks all combinations of symbols from the diagram under analysis, submitting different sequences with (activity, arrow, queue), to the rule. The rule will then return *null* if the symbols are inappropriate, or return a new symbol to the diagram. Attributes of each new symbol are “richer” than previous (collecting pieces of code). For this grammar, after several rule modifications, the generated program is represented within an attribute of a synthesized symbol (*simulation*).

## 7 XML FILE INTERCHANGE FORMAT

The objective was to give birth to a proven visual language (an example of a complete program written in this language can be found in Figure 8). This program is not dependent on the translator/compiler used. We defined an XML format (eXtended Markup Language) making possible to store it with all the attributes and to transfer it to any application in a text file (since to keep it in bitmap format (as a photo) would not obviously be appropriated).

In Figure 13 we can find the DTD corresponding to the defined XML format. In Figure 14 we can see a portion of one XML file corresponding to the barman problem. Each file is a book: a collection of sheets. Each sheet contains one ACD diagram. (Books can have one or more sheets).

```

<?xml version='1.0' encoding='UTF-8' ?>
<!--
<?xml version="1.0"?>
<!DOCTYPE Book SYSTEM "ACD.dtd">
<Book>
...
</Book>
-->
<!ELEMENT Visible (PCDATA)>
<!ELEMENT Value (PCDATA)>
<!ELEMENT Property (Visible|Type|Value)*>
<!ATTLISTProperty
  Name CDATA IMPLIED
>
<!ELEMENT Foreground (PCDATA)>
<!ELEMENT Background (PCDATA)>
<!ELEMENT Type (PCDATA)>
<!ELEMENT Y (PCDATA)>
<!ELEMENT X (PCDATA)>
<!ELEMENT Shape
  (Property|Foreground|Background|Type|Height|W
  idth|Y|X)*>
<!ATTLIST Shape
  Type CDATA IMPLIED
  Uid CDATA IMPLIED
>
<!ELEMENT Height (PCDATA)>
<!ELEMENT Width (PCDATA)>

<!ELEMENT WorkSheet (Shape|Height|Width)*>
<!ATTLIST WorkSheet
  Order CDATA IMPLIED
  Name CDATA IMPLIED
>
<!ELEMENT Book (WorkSheet)*>
<!ATTLIST Book
  Name CDATA IMPLIED >

```

Figure 13: DTD file (XML specification)



```

o o o
//@@@@@@@@@@@@ Activities END @@@@@@@@@@@@@
public void end_of_WASH(){
    r=Bsf.remove(WASH_BARMAN,Clock); //(returns r.time,r.at1,r.at2)
    Bsf.insert(IDLE,Clock,Clock,r.at2);
    for ( int i=3; i>0; i--){
        r=Bsf.remove(WASH_GLASS,Clock); //(returns r.time,r.at1,r.at2)
        Bsf.insert(CLEAN,Clock,Clock,r.at2);
    }
    //Attempts to start subsequent activities that may be viabilized
    begin_of_POUR(); //priority=1
    begin_of_WASH();
} //& end_of_WASH

o o o
//End of Program 'BARMAN Problem'

```

Figure 16: Extract of Java generated simulation program

The Figure 17, below, contains a screenshot of the bartender program execution, with configurable initial entities allocation, progressive running bar and final report.

```

----- ( 3 - Execution ) -----
INIT: 12GLASS>DIRTY 1DOOR>DOOR_FREE 10customer>OUTSIDE
1BARMAN>IDLE
...10%...20%...30%...40%...50%...60%...70%...80%...90%...100% [in 40ms]

Relatorio em t = 1000 (Warm-up period 0-120)
Fila

```

	In	Out	Now	Av-stay	Av-len
1 FEvent	267	264	3	9.189	2.614
2 CLEAN	101	94	7	75.511	7.669
3 FULL	94	94	0	0.000	0.000
4 DIRTY	96	93	3	22.129	2.365
5 DOOR_FREE	45	45	0	0.000	0.000
6 READY	94	94	0	0.000	0.000
7 WAIT	94	94	0	2.138	0.228
8 OUTSIDE	52	45	7	147.667	7.334
9 IDLE	125	125	0	1.328	0.189
10 ARRIVE_DOOR	46	45	1	19.844	1.000
11 ARRIVE_customer	46	45	1	19.844	1.000
12 POUR_customer	95	94	1	5.947	0.635
13 DRINK_customer	94	93	1	7.516	0.802
14 POUR_GLASS	95	94	1	5.947	0.635
15 POUR_BARMAN	95	94	1	5.947	0.635
16 DRINK_GLASS	94	93	1	7.516	0.802
17 WASH_BARMAN	31	31	0	5.000	0.176
18 WASH_GLASS	93	93	0	5.000	0.528

Figure 17: Screenshot of program execution and final statistical report

This task (automatic translation) is quite complex since it implies different abstraction levels, from an high level (activity world view) to a much lower level (event world view) using a generic programming language.

## 9 ARENA MODEL CREATOR

The hard part of this task was to create an activity-based executive over a process-oriented environment. The executive developed shows to be more efficient than a three-phase approach since we used a message passing mechanism that only tries to start an activity when entities have arrived to one of its predecessor queues. Given that we generate a model into a high-level simulation environment it becomes possible to overcome limitations of the activity-based approach since the model may be completed in ARENA.

Figure 18 and Figure 19 illustrate the model as automatically created in Arena for the Bartender Problem and a Screenshot of animation phase. Figure 20 have details of two activities with predefined statistics. Figure 21 have a screenshot with part of the logic template (code) of the activity block.

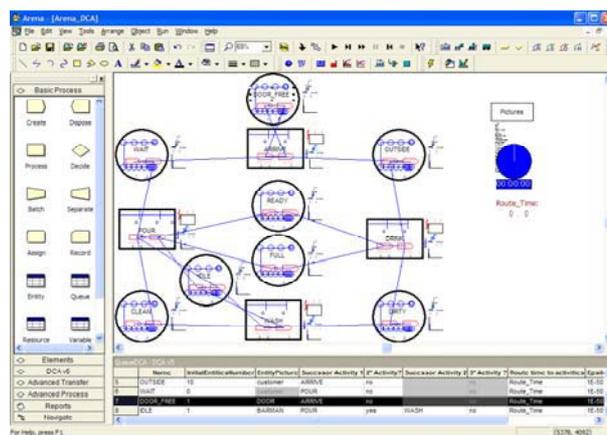


Figure 18: Screenshot of one automatically built model in Arena (Bartender)

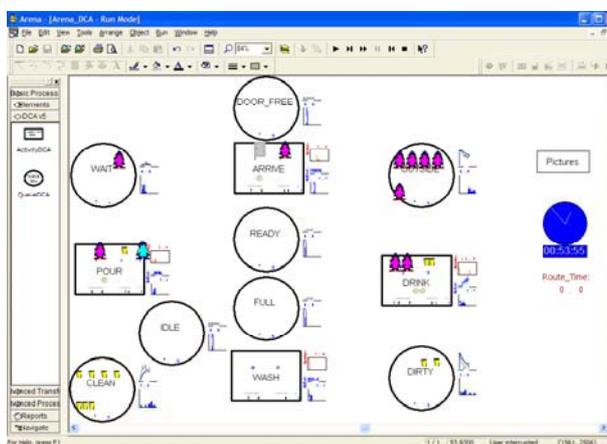


Figure 19: Screenshot of animations in Arena (Bartender)

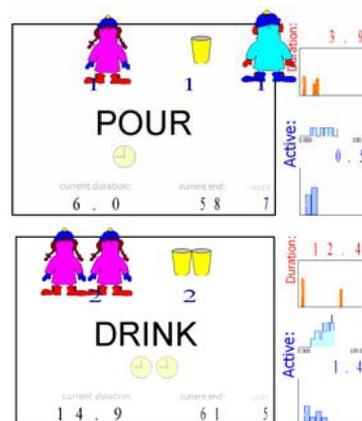


Figure 20: Screenshot of Arena animation details, with statistics (Bartender)

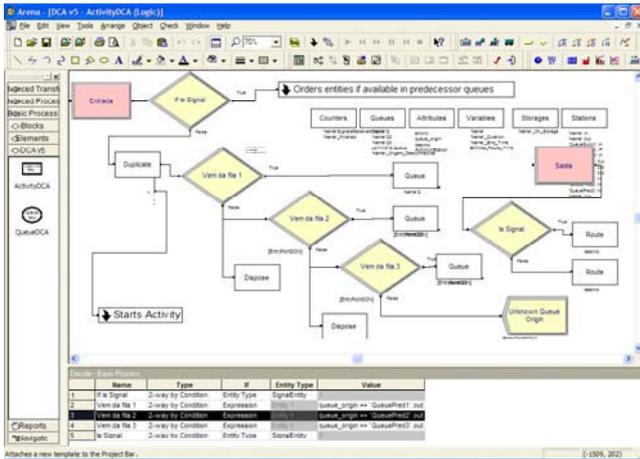


Figure 21: Activity definition - Part of Arena Logic Template.

## REFERENCES AND BIBLIOGRAPHY

Banks, Jerry. 1998. *Handbook of Simulation*. Industrial and Systems Eng. Georgia, Wiley.

Bekey, George A., ed. 2003. *Modeling and Simulation Theory and Practice (A Memorial Volume for Professor Walter J. Karplus)*. Kluwer Academic Publishers,

Bell, Brigham. 1992. "Using Programming Walkthroughs to Design a Visual Language". Ph.D. dissertation, Department. Of Computer Science, University of Colorado, Boulder, Technical Report CU-CS-581-92.

Bennett, Brian S. 1995. *Simulation Fundamentals*, Prentice Hall.

Campione, M.; K. Walrath; A. Huml. 1999. *The Java Tutorial Continued, The Rest of the JDK*, Addison Wesley.

Clementson, A.T. 1982. *Extend Control and Simulation Language, Users Manual*. (ECSL). CLE.COM Ltd, University of Birmingham.

Costagliola, Gennaro; Andrea de Lucia; Sergio Orefice and Genoveffa Tortora. March 1995. "Automatic Generation of Visual Programming Environments". *Computer IEEE, vol.8 n°3*, pp. 56-66.

Dias, Luís S. 1997. "Linguagens Visuais de Programação – Paradigmas e Ambientes", MSc Thesis (in Portuguese), Computer Science Department, University of Minho, Braga, Portugal.

Dias, Luis S. and A. Guimarães Rodrigues. 2002. "Language independent modelling of discrete event simulations – AIMS", *AIS'2002 – AI, Simulation and planning in High Autonomy Systems*, pp.195-200. SCS Publication.

Dias, Luis S. and A. Guimarães Rodrigues. 2002. "Towards simplicity in modelling for simulation", *Operational Research Society - Simulation Study Group. First biennial Simulation Study Group Workshop*, Birmingham, UK.

Filho, Wladimir De Lara Araújo and Celso M. Hirata. 2004. "Translating activity cycle diagrams to Java simulation programs". *IEEE 37th Simulation Symposium*, 18-22 April 2004. Page(s):157 - 164

## 10 CONCLUSION

The work presented in this paper could constitute a major step towards the generalisation of the use of simulation. In fact, we suggest the use of a simple interface (Activities Cycle Diagrams) to model a real situation. Then we present a tool capable of generating a simulation program. Based on event scheduling simulation modelling philosophy, our tool automatically generates a program to use Basic Simulation Facility routines. Based on process flow simulation modelling philosophy, our tool automatically generates an ARENA program. Furthermore the mentioned automatic generation of simulation programs does not require expertise in simulation.

Fishwick, Paul A. 1994. *Simulation Model Design And Execution; Building Digital Worlds*. Prentice Hall.

Freimer, M., T. Roeder, C. Harmonosky, L. Schruben, C. Standridge, I. Ståhl, 2004. You Are Going To Teach Simulation – Now What? Tips and Strategies. In *Proceedings of the 2004 Winter Simulation Conference*, ed. R .G. Ingalls, M. D. Rossetti, J. S. Smith, and B. A. Peters. 2057-2065. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Garrido, José M. 2001. *Object-Oriented Discrete-Event Simulation with Java - A Practical Introduction*, Kluwer Academic/Plenum Publishers.

Harrell, Charles R. et.al. 2003. *Simulation using PROMODEL*, McGraw-Hill, 2nd edition.

Henriques, Pedro Rangel. 1992. "Atributos e Modularidade na Especificação de Linguagens Formais", PhD thesis, University of Minho, Braga, Portugal.

Karayanakis, N. M. 1995. *Advanced System Modeling and Simulation W/ Block Diagram*. Springer.

Kelton, W. David; Randall P. Sadowski, and David T. Sturrock, 1998-2004. *Simulation With Arena* (third edition). McGraw-Hill.

Kheir, Naim A., (Ed.). 1996. *Systems Modeling and Computer Simulation*. 2nd Ed. Dekker.

Law, Averill M. and W. David. Kelton. 1991. *Simulation Modeling & Analysis*. McGraw-Hill.

Martins, F. Mário. 1998. "Programação Orientada aos Objectos usando Java"(in Portuguese), Computer Science Department, University of Minho, Braga, Portugal.

Mooney, Christopher Z. 1997. *Monte Carlo Simulation*. Sage Publications.

Morgan, B.J.T. 1984-1995. *Elements of Simulation*. Chapman&Hall.

Odum, Howard T. and Elisabeth C. Odum. 2000. *Modeling for all Scales: An Introduction to System Simulation*, Academic Press.

Overstreet, C. M. and Richard E. Nance. 2004. Characterizations And Relationships Of World Views. In *Proceedings*

of the 2004 Winter Simulation Conference, ed. R .G. Ingalls, M. D. Rossetti, J. S. Smith, and B. A. Peters. 279-287. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

- Pidd, Michael. 1984-1990. *Computer Simulation in Management Science*, Wiley.
- Pidd, Michael. 2004. Simulation Worldviews - So What?. In *Proceedings of the 2004 Winter Simulation Conference*, ed. R .G. Ingalls, M. D. Rossetti, J. S. Smith, and B. A. Peters. 288-292. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.
- Repenning, Alex and T. Sumner. March 1995. "Agentsheets: A medium for Creating Domain-Oriented Visual Languages", *Computer IEEE*, vol.8 n°3, pp.17-25.
- Rocha, Jorge Gustavo. 1995. "Especificação de Linguagens Visuais de Programação", MSc Thesis (in Portuguese), Computer Science Department, University of Minho, Braga, Portugal.
- Rodrigues, A. Guimarães. 1987-98. *Simulação*, Engineering School, University of Minho, Braga, Portugal.
- Rodrigues, A. Guimarães. 1995. "Modelação em simulação discreta, (lição de síntese)", University of Minho, Braga, Portugal.
- Ross, Sheldon. 2002. *Simulation*. Elsevier Academic Press.
- Rubinstein, Reuven Y., and Benjamin Melamed. 1998. *Modern Simulation and Modeling*, Wiley.
- Sargent, Robert G. 2004. Some recent advances in the process world view. In *Proceedings of the 2004 Winter Simulation Conference*, ed. R .G. Ingalls, M. D. Rossetti, J. S. Smith, and B. A. Peters. 293-299. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.
- Sarjoughian, Hessam S.; Francois E. Cellier, and Bernard P. Zeigler, (Editors). 2001. *Discrete Event Modeling and Simulation Technologies: A Tapestry of Systems and AI-Based Theories and Methodologies: A Tribute to the 60th Birthday of Bernard P. Zeigler*. Springer Verlag.
- Swain, James J. 1991-2003. "Simulation Software Survey". *OR/MS Today magazine* from Institute for Operations Research and the Management Sciences (INFORMS). Lionheart Publishing.
- Tocher, K.D. 1963. *The Art of Simulation*. UNIBOOKS – English Universities Press.
- Tumay, H. Harrington Kerim. 2000. *Simulation Modeling Methods To Reduce Risks and Increase Performance*, McGraw-Hill.
- Varanda M. J., Henriques P. e Rocha J. 1997. "Concepção, especificação e implementação de Processadores de Linguagens Visuais". *II Simpósio Brasileiro de Linguagens de Programação*.
- Zeigler, Bernard P.; Herbert Praehofer, and Tag Gon Kim. 2000. *Theory of Modeling and Simulation: Integrating Discrete Event and Continuous Complex Dynamic Systems*, 2<sup>nd</sup> ed. Academic Press.

## AUTHOR BIOGRAPHIES



**LUÍS MIGUEL DA SILVA DIAS** is a Lecturer at the Department of Production and Systems Engineering at the University of Minho. He has earned a degree in Informatics and Systems Engineering, and a Master in Informatics from the University of Minho. His interests are in modelling & simulation, operational research and visual languages.

<lsd@dps.uminho.pt>  
<<http://www.dps.uminho.pt>>



**ANTÓNIO JOSÉ M GUIMARÃES RODRIGUES** is a full Professor in the Department of Production and Systems Engineering at the University of Minho. He is, since 2002, the Rector of the University. He has earned a Master's Degree and Ph.D. from the University of Birmingham in Production Engineering. Prof. Guimarães Rodrigues interests are in simulation, operational research and mathematical programming. He is Associate Editor of *Investigação Operacional*, the Portuguese Journal of OR.

<agr@reitoria.uminho.pt>



**GUILHERME AUGUSTO BORGES PEREIRA** is a Professor in the Department of Production and Systems Engineering at the University of Minho. He is the director of the industrial engineering and management *licenciatura* (first degree). He has earned a Master's Degree in Operational Research. and Ph.D. in Manufacturing and Mechanical Engineering from the University of Birmingham. His interests are in simulation and operational research.

<gui@dps.uminho.pt>

University of Minho was founded in 1973, is located in Braga and Guimarães, in the north of Portugal. Has more than 15000 students, and around 500 Ph.D. Teachers.  
<[www.uminho.pt](http://www.uminho.pt)>

# DEVS MODELING OF SELF ORGANIZED COMPANIES' NETWORK

Lynda MEKAOUCHÉ, Fouzia OUNNAR, Patrick PUJO, Norbert GIAMBIASI  
Laboratoire des Sciences de l'Information et des Systèmes (LSIS) Université Paul Cézanne Avenue Escadrille  
NormandieNiémen13397 Marseille Cedex 20 – France  
{lynda.mekaouche, fouzia.ounnar, patrick.pujo, norbert.giambiasi}@lisis.org  
Tel. (+33) 4 91 05 60 10 – Fax. (+33) 4 91 05 60 33

## KEYWORDS:

Self evaluation, customers-suppliers relationship, DEVS formalism.

## ABSTRACT

The appearance of complex products on the market involved a change and an increase of complexity at the Customer/Supplier Relationship (CSR) level. Indeed, the company had recourse to outsourcing in order to ensure their survivals. Accordingly we present a new approach of the CSR, where the whole of the entities (customer/suppliers) are self organized in order to better answer to a given call for proposal launched by the customer, and in order to exploit the capacities of a supplier with better way. First, we describe the operation of the proposed approach. Then, we detail the DEVS modelling of self organized customers / suppliers.

## INTRODUCTION

The efforts carried out in order to improve the production management were concentrated on the improvement of internal management within each company with an aim of better answering the customer needs. Indeed, the studies were multiplied in order to install tools allowing companies to achieve their goals in term of performance improvement and profiles maximization. Researches were focused on the times and cost reduction and the increase in the products diversity like their quality. Industries globalization, customer requirements evolution and appearance of complex products, allow companies to realize that the internal improvement is important but not sufficient. This deduction pushed researchers to go so further to prove that the integration of company into a network is essential (Brito and Roseira 2003)(Castelain et al 2003)(Cousins and Spekman 2003)(Faems and Van 2003)(Silvadasan et al 2001). Thus, the companies had recourse to outsourcing and externalization (Ounnar and Pujo 2001) in order to realize the complex products. Through the externalization phenomenon, companies tend to gather for the realization of a joint project. Indeed, company fits in a customers/suppliers network, forming thus a supply chain network in order to optimize it by satisfying the customer. With the appearance of this new form of organization, the research widened, the targeted objectives does not relate to only one company but must meet the whole corporate

network. Accordingly, several topics were accosted, certain were focused on the co-operation between firms, noting that more recently, the agreements cooperation inter companies appeared as major form of valorization, competence, innovation capacity and reputation (Despontin et al 2002)(Telle 2003). Owing to the integration of the company into the network which generated the complexity of the Customer/Supplier Relationships (CSR), other studies were interested in the influence of these relations on costs inter organizational management (Brun and Staudacher 2000)(Harri 2002), durability of these relations (Alcouffe and Corrége 2004), dynamics of these relations (Lauras 2004). At last, other studies were interested on modeling of corporate network and on setting up of methodologies allowing modeling of this type of network (Bisigniano and Palermo 2003)(Bruzzone 2002)(Burlat 2004) (Chen and al 2001) (Dong and Nagurney 2002)(Villa 1998)

From these researches, we can conclude that the CSR control<sup>1</sup> is based on co-operation of various entities in order to achieve a common goal (Alvarez and Diaz 2004). This co-operation imposes integration of negotiation and communication means between companies. In other words, the development of the durable Customer/Supplier Relationships (CSR) depends on existing degree of confidence. Suggested solution in order to avoid any imbalance at the network is the implementation of the winning - winning principle. We attend these last years, to the appearance of industrial partnership relation. On this basis, we propose a new approach of CSR control by considering that the whole of the entities (customers/suppliers) partners communicating on the same medium of communication, negotiate to answer with the better way to the customers needs. In other words, to respond to Calls For Proposal (CFP) launched on the network by the customers, and to exploit suppliers' capacities with the better way. We propose thus to provide each supplier with a decision-making centre: Autonomous Control Entity (ACE), which allows him to self evaluate his performance in order to be able to take part to negotiation within a self organized network. This centre allows a supplier to become an intelligent production unit able to operate in self organization with other companies with an aim of seeking the best response to a

---

<sup>1</sup> Actions developed together in order to achieve common goals and react at a good moment to any dysfunction.

CFP launched on the network (Ounnar et al 2004). This entity is made up of four modules: Communication module, Interaction module, Optimization module and Planning module

In this paper, we will particularly presented Optimisation module. For that, we will present a formal description of this module and we will detail the calculation methods used for the self evaluation of an ACE with respect to a given CFP.

## GENERAL OPERATION OF THE ACE

The suggested approach allows the increase of the autonomy of the network entities. For that, the entities must have the capacity to negotiate and communicate among them in order to achieve their common goal which is to ensure collectively the distribution of the orders coming from the various customers with respecting the interests of each one. A partner can be customer, supplier or both. One of the customers launched a CFP on the network. This latter will be provided with a certain number of information such as: name of the transmitting entity (customer), description of the requested product, the quantity expected by the customer, lead time of end of negotiation, delivery lead time, etc ...All the ACEs which are connected to the network will received this CFP. Once the CFP received by a given ACE, via its communication module, the latter transmits the received information to the interaction module. The interaction module will check the feasibility of the CFP in technical term and then transmit the CFP to the optimization module. The optimization module starts the application of the selected multicriteria method (Ounnar 1999): Analytic Hierarchy Process (AHP) in order to obtain a classification of all the CFPs received, according to the entity capacity to treat them. The application of this method requires a set of qualitative or quantitative criteria (Ounnar and Pujo 2005). Among the quantitative criteria, appears operating time of the CFP. This data depends on the planning state and on the availability of equipments. We propose to obtain this data by the execution of an analytic method at the level of the planning module. This latter calculates the operating time of the CFP by studying the various possible states of insertion of this one in the entity planning. This result will thus be transmitted to the optimization module in order to finish the application of the multicriteria method. The interaction module compares then this performance with regard to the best actual performance and then sends it on the network if it is the best one. The various messages circulating on the network can be summarised as follow: CFP, RCFP, LCFP (Local Call for Proposal which is diffused by the entity), RLCFP (Response to Local Call for Proposal, response proposed by a given network partner), ERCFP (Entity Response to a Call for Proposal which is a proposed response to a CFP launched by a given partner). The objective of the suggested approach is to obtain a balance between charge/capacity at the level of

one supplier and to achieve a loads smoothing between the various suppliers with a further objective to propose an equitable system between network suppliers. For that, and in order to test the validity of the suggested approach through the test of the number of CFPs treated, negotiated, refused, at the level of each entity, we modeled the ACE by using DEVS formalism (Discrete EVent systems Specification) developed for modeling and simulation of discrete events dynamic systems. (Zeigler 1984). This validation allows the evaluation of the CFPs distribution on the partnership network.

As it was mentioned above, this paper is focused on the study of Optimization Module of an ACE. First, we describe the general operation of this module. Thus we explain how it allows the ACE to self evaluate its performance with respect to the received CFP, in order to estimate its own capacity to respond to this one. Then, we present DEVS models corresponding to this module and how they allow to validate the suggested approach. These models will be used as formal specifications in the system realization.

## Operation of Optimization Module

The performance evaluation is based on a multicriteria method. The selected method corresponds to the Analytic Hierarchy Process (AHP) method (Ounnar 1999). AHP is a powerful and flexible tool of decision-making for complex problems involving multiple qualitative and quantitative criteria. The method helps decision-makers to structure the significant components of a problem in a hierarchical tree-like structure. The results are then synthesized by reducing complex decisions into a series of simple comparisons and arrangements. AHP is thus a decision-making process that directly interprets the data by forming judgments through a scale of measurement inside a hierarchical structure. AHP involves four distinct steps.

- 1 Step 1 (setup): Decision-making criteria are generated. Hierarchical relationships are established among the criteria and are then represented in the form of a matrix.
- 2 Step 2 (weighing): The matrices are filled with criteria comparisons. The comparisons allow the calculation of the criteria-weighing vector.
- 3 Step 3 (ranking): The various solutions are ranked according to their ability to satisfy the various criteria.
- 4 Step 4 (evaluation): The final solution ratings are then calculated using the rankings determined in Step 3 and the weighing vector calculated in Step 2.

This process organizes a hierarchical decision-making problem in a mathematically rigorous manner to ensure proper results. It separates the decision-making process

into stages to enable the team working on the problem to focus successively on each step needed to make a decision. On the basis of qualitative or quantitative criteria, AHP method ensures to classify CFPs, according to the capacity of the entity to treat them. Among the quantitative criteria, appears operating time of the CFP. This data is obtained by the planning module. For that, the knowledge of the states of CFPs is necessary. The different CFPs received by the entity are placed in the planning module. A CFP can be in one of the following states:

- 1 Negotiated CFP: characterizing the fact that we have no information about its assignment.
- 2 Engageable CFP: characterizing the fact that an entity is the most successful on an order (the offer is better than the best of the received offers).
- 3 Pre-engaged CFP: a CFP is pre-engaged if it is “engageable” and it is selected as being one of the most priority of the CFP list. The entity appropriates temporarily this CFP. This CFP will be the following one that will take the state “Engaged” if there is no overbid.
- 4 Engaged CFP: the entity appropriates definitively the “pre-engaged” CFP on its planning at its engaged date.
- 5 Refused CFP: specify the fact that no proposition was made for this call for proposal.

### Formal Description of Optimization Model

In this section we present the DEVS model corresponding to the optimization module described above. This model is a coupled DEVS model composed of four atomic DEVS models (see figure.1).

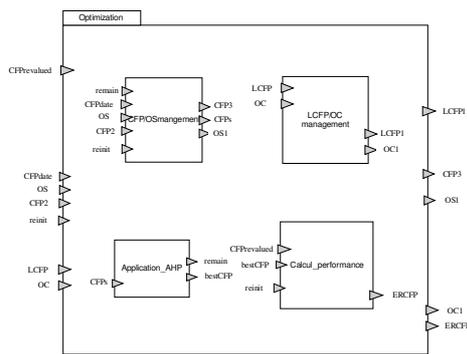


Figure 1. DEVS Model of Optimization Module

Atomic DEVS model specification is structured as follow:

$DEVS = (X_M, Y_M, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta)$ ; Where:

$X_M$ : The set of input ports through which external events are received.

$Y_M$ : The set of output ports through which external events are sent.

$S$ : The set of the sequential states.

$\delta_{ext}$ : the external transition function which specifies how the system changes state when an input (x) is received during the time “e” where  $e \in ta$ , then , the system passes from the current “ s ” state to a new one “s'” by applying the function .  $\delta_{ext}(s, e, x)$ .

$\delta_{int}$ : The internal transition function which specifies to which next state the system will transit after the time given by the time advance function has elapsed. The latter can generate an external output just before it takes place.

$\lambda$ : The output function, the latter generates an output event.

$Ta(s)$ : For a given state, s,  $ta(s)$  represents the time interval during which the model will remain in the state “s” if no external event occurs.

### Formal description of the “CFP/OS management” submodule

This module receives CFP as well as the orders of suppression of CFPs coming from the interaction module. The CFP/OS management submodule sends the CFP to the planning module which tests the possibility of insertion of this CFP and returns the CFP provided with its operating time (result of insertion). In addition, the Application\_AHP submodule informs the CFP/OS management submodule of its availability to receive CFP by sending to it the result of AHP in the form of a classified CFPs list. Then the CFP/OS management submodule sends to the Application\_AHP submodule the new list of CFPs to be classified.

Input ports:  $X_{CFPOS} = \{CFP2, CFPdate, OS, rest, reinit\}$ , where:

1  $CFP2 = \{CFP_i\}$ : indicates the arrival of a CFP<sub>i</sub> from the interaction module. The latter is represented by:

$Num_i \in \mathbb{N}$ : number of the CFP<sub>i</sub>.

Entity<sub>i</sub>: name of the entity defined on the set of string.

$CFP.Type_i \in \{feasible, unfeasible\}$ : it describes the feasibility of CFP<sub>i</sub> in technical term.

$CFP.state_i \in \{negotiated, engageable, pre-engaged, engaged\}$ : it describes the state of the CFP<sub>i</sub>.

$CFP.cond_i \in \{OK, not\ OK\}$ : describes the availability of the execution conditions.

$Q_i \in \mathbb{R}$ : CFP quantity expected by the customer.

$drecept_i \in \mathbb{R}$ : reception date of CFP<sub>i</sub>.

$OT_i$ : operating time of CFP<sub>i</sub> defined on the set of reals.

$DE_i \in \mathbb{R}$ : date of beginning of the execution of the CFP<sub>i</sub>.

$D_i$ : execution span defined on the set of reals.

$DL_i$ : delivery lead time defined on the set of reals.

$TEN_i \in \mathbb{R}$ : lead time of end of negotiation.

$CR_i$ : conformity rates defined by the customer who launches a CFP<sub>i</sub>. It is defined on the set of reals.

$RR_i$ : return rates, it is defined by the customer who launches the CFP. It takes its values on the set of reals.

2  $CFPdate = \{CFP_i\}$ : It indicates the arrival of a CFP<sub>i</sub> coming from the planning module on which a calculation of OT was carried out.

3  $OS = \{CFP_i\}$ : it indicates the arrival of a suppression order coming from the interaction module concerning a given CFP<sub>i</sub>. It is represented by:  $\{num_i, entity_i, CFP.type_i, CFP.state_i, CFP.cond_i, Q_i, drecept_i, OT_i, DE_i, D_i, DL_i, TEN_i, CR_i, RR_i\}$ .

4  $rest = \{listeCFPs\}$ : allows the reception of the result of AHP method applied at the Application\_AHP submodule. This result is represented by a list comprising all the CFP<sub>s</sub> received classified according to the capacity of the entity to treat them except the CFP classified first.

5  $reinit = \{on\}$ : allows the reinitialization of all the system.

State variables:  $S = \{phase, T busy, LT, L5\}$  where:

1 Phase =  $\{wait, wait\_calcul, CFP\_recept, fusion\_L5\_LT, insertion, launch\_AHP, date\_recept, suppression, CFP\_recept\}$ .

2  $\in R^+$ : defined the life time of the current state.

3 T: defined the instance of the class object of CFP frame.

4 busy: Boolean variable indicating if AHP method is carried out at the Application\_AHP submodule. It is initialized with false.

5 LT: List where are stocked the classified CFP<sub>s</sub> by the Application\_AHP submodule.

6 L5: list where are stocked CFP<sub>s</sub> launched by the network partners intended to be classified by AHP method.

Output ports:  $Y_{CFPOS} = \{CFP3, CFPs, OS1\}$ , where:

1  $CFP3 = \{CFP_i\}$ : send the CFP<sub>i</sub> to the planning module with an aim of calculating its operating time.

2  $CFPs = \{listeCFPs\}$ : Send the list of the CFP<sub>s</sub> to the Application\_AHP submodule, in order to classify them.

3  $OS1 = \{CFP_i\}$ : send the OS to the planning module.

The corresponding DEVS model is defined in Figure 2:

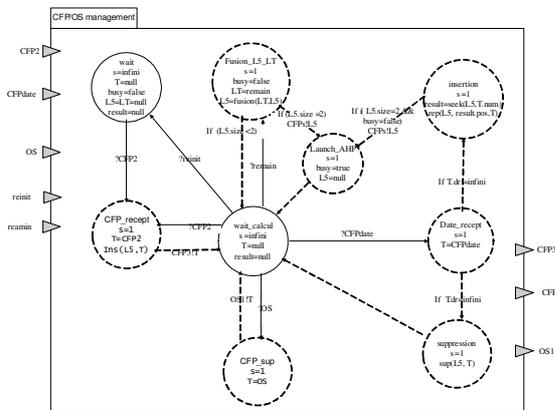


Figure 2 DEVS Model of CFP/OS management submodule

### Formal description of the “LCFP/OC management” submodule

This module ensures the management of CFPs coming from the planning module. It has to transmit them to the interaction module.

This submodule also ensures the management of Order of Change (OC) of a given CFP<sub>i</sub> state. Indeed, with the reception of an OC the submodule transmits it to the planning module.

Input ports:  $X_{LCFPOC} = \{LCFP, OC\}$ , where:

1  $LCFP = \{CFP_i\}$ : it indicates the arrival of a Local CFP coming from the planning module. This LCFP is composed of  $\{num_i, entity_i, CFP.type_i, CFP.state_i, CFP.cond_i, Q_i, drecept_i, OT_i, DE_i, D_i, DL_i, TEN_i, CR_i, RR_i\}$ .

2  $OC = \{CFP_i\}$ : It represents an order of change of a given CFP<sub>i</sub> state. This order is launched by the interaction module. This event is represented by:  $\{num_i, entity_i, CFP.type_i, CFP.state_i, CFP.cond_i, Q_i, drecept_i, OT_i, DE_i, D_i, DL_i, TEN_i, CR_i, RR_i\}$ .

State variables:  $S = \{phase, T\}$  where:

1 Phase =  $\{wait, LCFP\_recept, OC\_recept\}$ .

2  $R^+$ : it is the life time of the current state.

3 T: defined the instance of the class object of CFP frame.

Output ports:  $Y_{LCFPOC} = \{LCFP1, OC1\}$ , where:

1  $LCFP1 = \{CFP_i\}$ : send the LCFP to the interaction module.

2  $OC1 = \{CFP_i\}$ : send the OC on the state of a given CFP<sub>i</sub> to the planning module.

The corresponding DEVS model is defined in Figure 3:

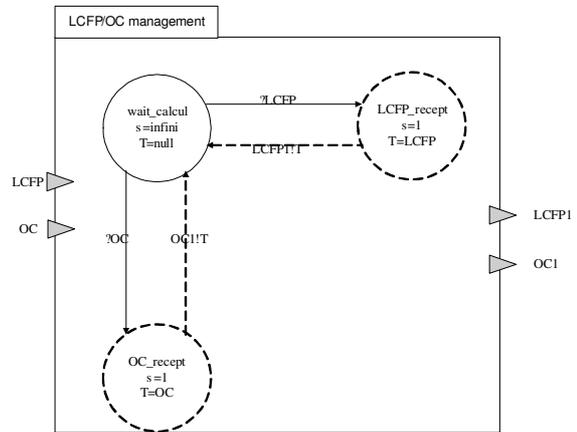


Figure 3 DEVS model of “LCFP/OC management” submodule

### Formal description of the “Application\_AHP” submodule

This module allows the application of the AHP on receiving CFPs list coming from the CFP/OS management submodule, with an aim of obtaining a classification of these CFPs.

Input ports:  $X_{Ap.AHP} = \{CFPs\}$ , where:

1  $CFPs = \{listeCFPs\}$ : allows the reception of a CFPs list coming from the management CFP/OS submodule, intended to be classified.

State variables:  $S = \{phase, , T, L9\}$  where:

1 Phase = {wait, App\_AHP, suppression}.

2  $\in R^+$ : defined the life time of the current state

3 T: defined the instance of the class object of CFP frame.

4 L8: list where the CFPs which are classified by AHP method are stocked.

Output ports:  $Y_{Ap,AHP} = \{bestCFP, rest\}$ , where:

1  $bestCFP = \{CFP_i\}$ : send the CFP classified first to the calcul\_performance submodule.

2  $rest = \{listeCFPs\}$ : send the list of the CFPs classified except the CFP classified first to the CFP/OS management submodule.

The corresponding DEVS model is defined in Figure 4:

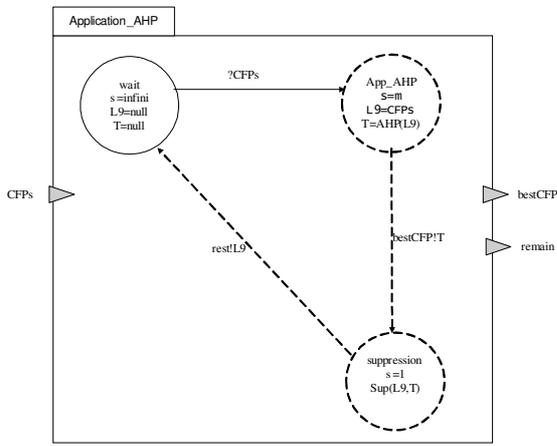


Figure 4 DEVS model of "Application\_AHP" submodule

### Formal description of the "calcul\_performance" submodule

This module allows the calculation of the entity performance with respect to the  $CFP_i$  classified first by AHP method. In the calculation of this performance it calls upon a mathematical formula based on a set of indicators applied by calcul\_perf function.

Input ports:  $X_{CP} = \{CFPrevalued, bestCFP\}$ , where:

1  $CFPrevalued = \{CFP_i\}$ : it indicates the arrival of revalued  $CFP_i$  coming from the planning module.

This last is represented by:  $\{num_i, entity_i, CFP.type_i, CFP.state_i, CFP.cond_i, Q_i, drecept_i, OT_i, DE_i, D_i, DL_i, TEN_i, CR_i, RR_i\}$ .

2  $bestCFP = \{CFP_i\}$ : allows the reception of the  $CFP_i$  on which the entity is better. This CFP is sent by the Application\_AHP submodule. It is represented by:  $\{num_i, entity_i, CFP.type_i, CFP.state_i, CFP.cond_i, Q_i, drecept_i, OT_i, DE_i, D_i, DL_i, TEN_i, CR_i, RR_i\}$ .

3  $reinit: \{on\}$ : allows the reinitialization of all the system.

State variables:  $S = \{phase, , T, TR, L6, L7, result\}$ , where:

1 Phase = {wait, seek\_RCFP, seek\_CFP, Insertion\_end, Insertion\_end1, Insertion\_end2, cruch}.

2  $R^+$ : defined the life time of the current state.

3 T: defined the instance of the class object of CFP frame.

4 TR: defined the instance of the class object of RCFP frame.

5 L7: list where the responses for a different  $CFP_i$ s proposed by the entity ( $ERCFP_i$ ) are stocked.

6 L7: list where the reevaluate  $CFP_i$ s are stocked.

For these variables are applied following functions:

1 seek (L,T.num): seek the element T in the list L. This function returns a pointer which points on the element T, if the element exists, or on the null value.

2 ins (L,last, T): inserts the element T at the last of the list L.

3 remp (L,pos,TR): replaces the element being in the position "pos" in the list L by element TR.

4 calcul\_perf (T): function allowing the calculation of the performance of a given CFP stocked in T by the application of a formula based on a set of indicators. This function returns a vector mad up of the following fields:  $\{num_i, entity_i, pi, drecept_i, TEN_i\}$ .

Output ports:  $Y_{CP} = \{ERCFP\}$ , where:

1  $ERCFP = \{ERCFP_i\}$ : allows the sending of the entity response concerning  $CFP_i$  on which it is better. This answer will be provided with the following information :  $\{num_i, entity_i, P_i, drecept_i, TEN_i\}$ .with:

$P_i$ : performance of the entity with respect to this call for proposal. It takes its values on the set of reals.

$drecept_i$ : date at which the response has been transmitted by the entity.

$TEN_i$ : lead time of end of negotiation defined on the set of reals.

The corresponding DEVS model is defined in Figure.5:



Figure 5 DEVS model of "calcul\_performance" submodule

## CONCLUSION

The goal of our study is to improve the decision making structures at the CSR level. After a presentation of the literature review on the work carried out on the improvement of the CSR, we have proposed a new approach of CSR control by considering that the whole of the entities (customers/suppliers) partners, communicating on the same medium of communication, negotiate to answer with the better way to the customers needs. We have thus provided each supplier with a decision-making centre: Autonomous Control Entity (ACE), which allows him to self evaluate his performance in order to be able to take part to negotiation. The objective of the suggested approach is to obtain a balance between charge/capacity at the level of each supplier and to achieve a loads smoothing between the various suppliers with a further objective to propose an equitable system between network suppliers. For that, and in order to test the validity of the suggested approach we have modeled the optimization model. Indeed, we have presented DEVS models corresponding to this module. These models will be used as formal specifications in the system realization. Further research will focused on the simulation of these models in order to validate the proposed approach.

## REFERENCES

- Alcouffe, C., Corrége, N. 2004 "Structures de gouvernance dans l'entreprise étendue: l'exemple AIRBUS". *Notes 393 de LIRHE*.
- Alvarez, E., Diaz, F. 2004 "An application of real-time scheduling system for turbulent manufacturing environment". In *Robotics and Computer-Integrated Manufacturing 20*, pp. 485-494.
- Bisignano, A., Palermo, O. 2003 "Fight environmental hostility! How negotiation can govern firm-institutions relationships". *19<sup>th</sup> annual Industrial Marketing and Purchasing Conference*, University of Lugano, Lugano, Switzerland.
- Brito, C. M., Roseira, C. 2003 "A Model for the Understanding of Supply Chain Networks". *19<sup>th</sup> annual Industrial Marketing and Purchasing Conference*, University of Lugano, Lugano, Switzerland.
- Brun, A., and Portioli-Staudacher, 2000 "A., Negotiation-driven supply chain co-ordination for small and medium enterprise. ECAI Workshop Notes – Agent Technologies and Their Application Scenarios" in *Logistics - 14th European Conference on Artificial Intelligence*, Berlino.
- Bruzzone, G. A., Mosca, R., Revetria, R. 2002 "Web integrated logistic designer and intelligent control for supply chain management". *Proceedings of the Summer Computer Simulation Conference*.
- Burlat, P. 2004 "La modélisation des réseaux d'entreprises : quelques travaux actuels et perspectives". *Actes de la journée de synthèse GDR MACS*, Aix en Provence.
- Castelain, E., Mathelin, S., Yim, P. 2003 "Analysis and modelling of the information system to manage the supply chain of a mail order selling firm with synchronous delivery". In *international association for management of technology (IAMOT)*.
- Chen, H., Amodeo, L., Chu, F. 2001 "modelling and performance evaluation of supply chain with Batch deterministic and stochastic petri nets". In *13<sup>th</sup> European Simulation Symposium*, pp. 415-419.
- Cousins, Paul D., Spekman, R. 2003 "Strategic supply and the management of inter- and intra- organizational relations". *Journal of Purchasing and Supply Management.No1*.
- Despontin, E., Briand, C., Esquiral, P. 2002 "Premier pas pour une aide à la coopération inter-entreprises aux niveaux opérationnels". *Actes de la journée du GRP*, Tarbes.
- Dong, J. Zhang, D. and Nagurney, A. 2002 "Supply chain networks with Multicriteria Decision-Makers". *Transportation and Traffic Theory in the 21<sup>st</sup> Century*, M.A.P. Taylor, editor, Pergamon, Amsterdam, The Netherlands, pp. 179-196.
- Faems, D., and Van, Looy B. 2003 "The role of inter organisational collaboration within innovation strategies: towards a portfolio approach". *10th International Product Development Management Conference, EIASM, Brussels (Belgium)*.
- Harri, I., kulma, 2002 "Accounting in customer-supplier relationships:Developing cost management in customer-supplier relationships: three case studies". *Proceedings of the 3rd Conference on New Directions in Management Accounting: Innovations in Practice and Research. Vol 2*. 699-716. Brussels, Belgium.
- Lauras, M. 2004 "Méthodes de diagnostic et d'évaluation de performance pour la gestion de chaînes logistiques : application à la coopération maison-mère-filiale internationales dans un groupe pharmaceutique et cosmétique", *thèse, I.N.P. de Toulouse*.
- Ounnar, F, Pujo, P. 2005 "Supplier evaluation process within a self-organized logistical network". In: *International Journal of Logistics Management*.
- Ounnar, F; Pujo, P, Mekaouche, L; Giambiasi, N. 2004 "Decentralized Self Organized Control of a Patnership Network in an Intelligent Supply Chain. " In: *IMS International Forum 2004: Global Challenges in Manufacturing*, Italy.
- Ounnar, F., Pujo, P. 2001 "Décentralisation des Mécanismes de Pilotage de la Relation Donneurs d'Ordres/Fournisseurs". *4e Congrès International de Génie Industriel, Vol 2*, pp. 1175-1185.
- Ounnar, F. 1999 "Prise en Compte des Aspects Décision dans la Modélisation par Réseaux de Petri des Systèmes Flexibles de Production". *Thèse I.N.P.G. -Laboratoire d'Automatique de Grenoble*.
- Sivadasan, S., Efstathiou, J., Huaccho, I., and Calinescu, A. 2001 "Complexity associated with supplier-customer integration". *Proceedings of the 17<sup>th</sup> National Conference on Manufacturing Research (NCMR 2001): Advances in Manufacturing Technology XV*, pp. 31-36, University of Cardiff, UK.
- Telle, O. 2003 "Gestion de chaînes logistiques dans le domaine aéronautique : aide à la coopération au sein d'une relation Donneur D'Ordres/Fournisseur". *Thèse de l'Ecole Nationale Supérieure de l'Aéronautique et de l'Espace*.
- Villa, A. 1998 "Organizing a 'network of enterprises': an object-oriented design methodology". *Computer Integrated Manufacturing Systems Vol. 11, N° 4*, pp. 331-336.
- Zeigler, B. 1984 "Multifaceted Modelling and Discrete Event Simulation", *Academic Press, London*.

# INDUSTRIAL MODELLING AND SIMULATION SKILLS EVALUATION PROCEDURES FOR RESEARCHERS

*Agostino G. Bruzzone, Enrico Bocca*  
**MISS - DIPTEM - CIELI - University of Genova**  
Via Opera Pia 15 – 16145 Genova, Italy  
st.itim.unige.it

*Marina Massei, Enrico Briano*  
**MISS - Liophant Simulation**  
Via Molinero 1 – 17100 Savona, Italy  
www.liophant.org

## ABSTRACT

Nowadays there are several techniques and systems that allows rational evaluation of candidates' skills and make simpler the researchers' work; so in this paper the main goal is to present an innovative way of evaluation of engineering performances and capabilities for candidates with background in industrial modelling and simulation.

## INTRODUCTION

It is interesting to underline the potential of the use of quantitative computer models and simulators in order to guarantee a significant support for the evaluation and recruitment of well-prepared young resources in engineering field.

Nowadays the standard human resource evaluation/recruitment procedures are based on analysing and solving little case studies, working alone or in team, evaluating carefully the skills of the candidates while solving problems and their team working capabilities; however with the new approach proposed by the authors based on the use of simulation models it is possible also to evaluate interactions, results, problem solving and reasoning, even if a problem on evaluating different solutions in term of effectiveness may occur.

In effect one of the main current problems for young post-graduated resources is that their capabilities in problem solving are strongly reduced by the qualitative approach applied in the recruitment policies. Computer may be used as a useful tool in order to simplify complex calculating operations and to face with the different scenarios that may occur.

During the recruitment process the candidates usually analyse the test case and provide a reasonable solution,

identifying criticalities, bottlenecks and economic analysis, especially in terms of costs or revenues, but almost all these parameters are only qualitative, and so it is very difficult for the researcher to have a strong detailed feedback on the proposed solution; for these reasons the authors have proposed the introduction of computer models devoted to provide measurable output: so it is possible to evaluate real-time the impacts of the solutions proposed by the candidate in term of overall performances.

Quantitative computer models are the new frontier on the recruitment process: it is possible to introduce new possibilities like the use of the web technologies devoted to spread the selection base to a large community all around the world and to provide the candidates a self-evaluating procedure, but not only: using the computer it is possible, during the final selection stage, to evaluate also the interaction without obscuration of the originality of the single solutions. In the following some applications of this methodology will be presented as well as the results reached with this experimental campaign.

## THREADS RELATED TO USE OF SIMULATION IN EDUCATION

The first field analyzed in this paper are the evaluation procedures in the education and training sector: these two areas are strictly linked and for a long time these sectors have benefited of the use of new techniques; for this reason it is interesting to evaluate the impact of the use of simulators and computer models, because it is very interesting to evaluate the solutions provided by students in the simulator, guaranteeing the possibility for the students to interact each other and to face with a "Virtual World".

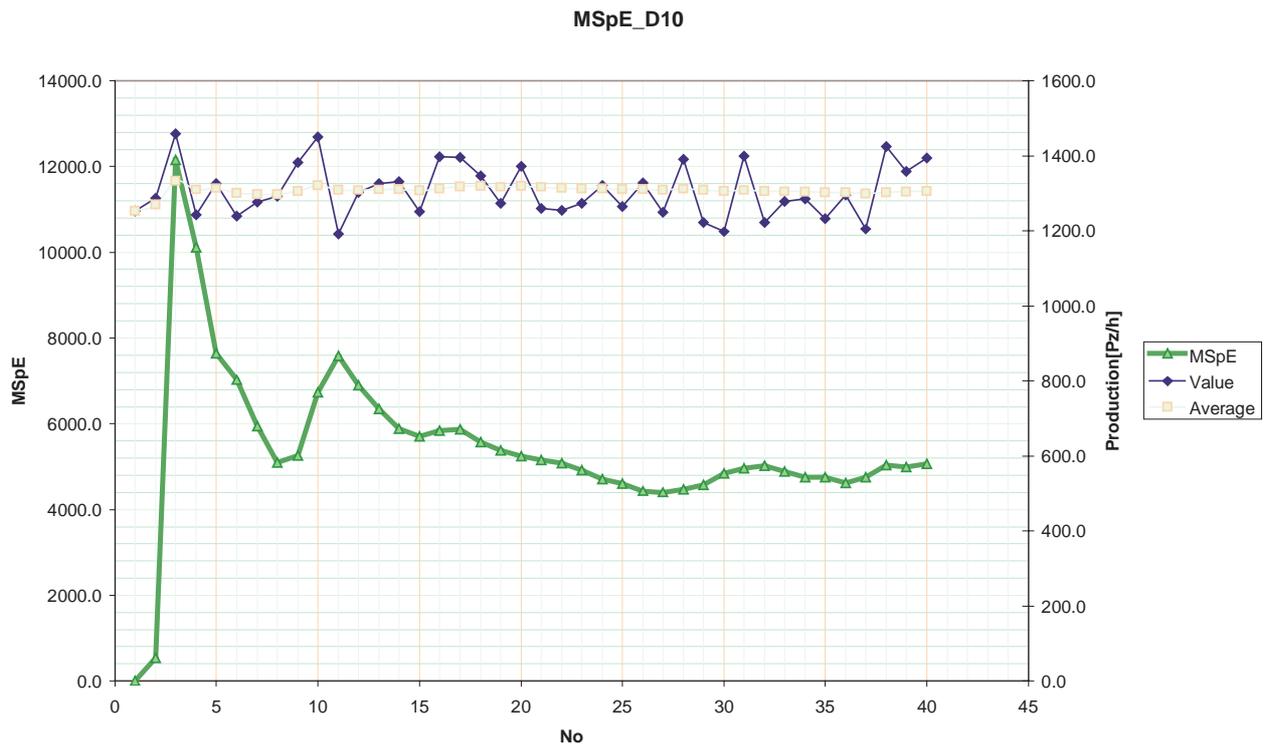


Figure 1: Example of Validation Report requested to Candidates based on DOE on a Simulator Scenario

With these tools it is possible to use models devoted to support real learning processes based on “Direct Experience”, but the changing is not pretty simple: the candidates have some preliminary requirements to satisfy, like good fundamentals (i.e.statistics), deep knowledge on DOE (Design Of Experiments) and similar methodologies, the awareness of the model constraints and limits, and so on.

The main concern about using this method could be related to the fact that the students, interacting directly with the models, could waste their time not discovering anything useful or losing their selves in a misunderstanding of goals due to the complexity of the systems.

In effect all the models proposed have inside approximations and this represents a further potential error source because operating in improper conditions the approximations could lead students to induct wrong relations from the simulation; therefore usually it is very difficult to discover those potential errors and to correct them, even because complex logical operations inside simulators are covered with very good looking and attractive animation or graphic that makes them look very realistic.

Even considering these aspects and risks the author experience shows that the main limit for using those

techniques in education is the lack of tool availability in this specific field; so there is a large potential for development of tools in this area.

### SCIENTIFIC SKILL EVALUATION BY USING M&S

Simulation applied in the field of scientific capability evaluation and research selection is to be considered a logic step due to sector evolution: very often in effect selection procedures are nowadays based on the results of the approach of candidates to simulated scenarios. In this case the term “simulated” is not linked to computer simulation, but in a certain sense it is more understandable how computer M&S techniques could very usefully help to define those scenarios, in a realistic and challenging way not possible with other approaches.

Based on these possibilities it is possible to see a parallel with the evolution of the simulation for training especially in the military field, where the first steps were made in the middle of XX century, evolving to distributed operations in the 90s and continuously developing in present times towards automatic evaluation systems.

## Testing Unitary Production

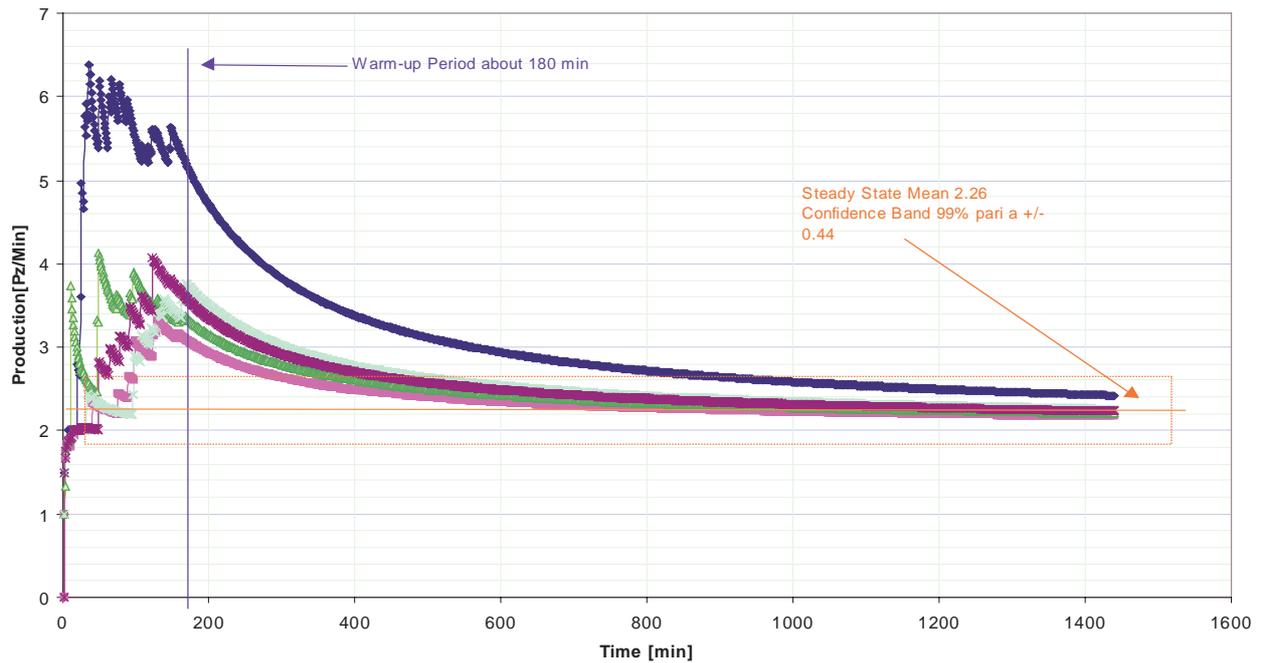


Figure 2: Evaluation Criteria applied to the reports provided by different candidates

But any case it is an innovative approach to define new models for operating in the personnel selection, such as:

- Fast screening for evaluating large numbers of candidates that operate on small but challenging system interconnected through the web interacting with simulation systems and communicating the results
- Complex problem remote analysis that can be submitted to the remote instructor by a group of trainees that can operate in a longer timeframe remotely connected.

Quick screening can enlarge the number of candidates allowing to concentrate the traditional screening, more time consuming and expensive, on a smaller and pre-selected number of candidates, allowing also to redesign the final selection procedure putting focus on additional issues. The second type of selection could be instead applied to quantitative comparison of experts with an effort limited for what concerns internal resources and procedure times. Using these approaches and other new ones in the process of researchers selection can lead to an improvement of evaluation approach leading to comparative and quantitative performance analysis.

In this paper it is proposed a set of preliminary applications in which to apply those concepts with the experiments made and their successful result.

### CASE STUDIES AND APPLICATIVE EXAMPLES

The first example of use of computer simulation for researchers screening, presented in this paper, has been carried out in the academic personnel evaluation, it is very difficult indeed to bound research in a simulation model, but there are some specific situations in which research need a very deep integration with these particular expertise. The application was made in the engineering field, especially in the industrial engineering sector, that requires the capability of testing in a scientific way problems affected by variables to be measured, data to be analyzed and compared, conceptual liaisons to be estimated. In this case the capacity of the candidate to analyze and evaluate realistic problems using computer instruments in order to solve them can be identified using simulations models; in particular even if this is a very limited application of these concepts, it is important to underline how quantitative results based on computer simulation are useful to measure such skills.

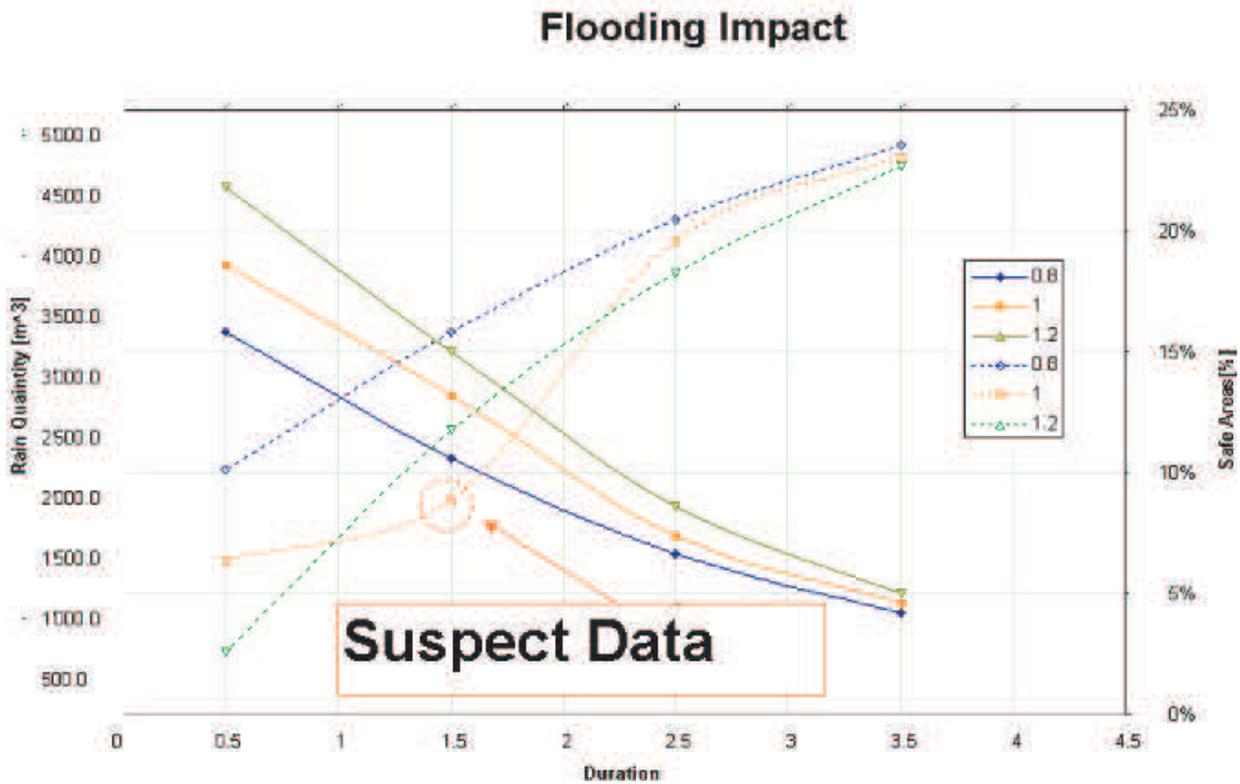


Figure 3: Setting up a Challenge in Data provided in a Scenario for a Researcher Selection

Obviously these results need to be integrated with other evaluation procedures in order to support effectively the overall selection process.

Let the authors introduce a set of computer simulation package devoted to be applied in the sector of evaluating post-graduates for positions in academic environments; among the others some of the packages developed for industrial engineering candidates was related to different kind of simulators:

environmental management, mechanical department, manufacturing facility, engineering tenders, etc.

In particular focusing on the case For instance in the case manufacturing, it was request to estimated and identify the real system warm up period and the time baseline in order to obtain results to compared with real system data.

In addition on this example, the authors were developed other three different simulators tools to be applied in this sector to evaluate candidate focusing on more technical release in experimental techniques such as that related to complex systems: missile launching procedure, micro particle interactions, weather impact on flooding phenomena over a geographic region.

For instance In that example, in order to select different technical/scientific thematic areas the candidate has to identify some correlations, in particular comparing the results obtained by using simulation with the real data.

In this case, devoted to selection on different technical/scientific thematic area, the candidates was requested to identify, by using the simulator, some correlation also comparing the results with real data.

One important part usually is to request candidates to determine correlations among different factors and/or the goal is to identify the wrong data using the validated model.

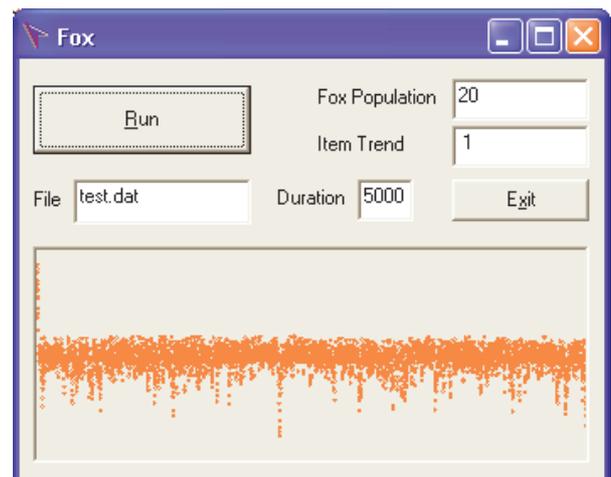


Figure 4: Example of a Simple VB Simulator devoted to Candidate Evaluation

The tools was extensively used in researcher evaluation processes involving industrial engineering personnel, allowing to obtain a motivated quantitative selection quickly over a set of 45 candidates; the candidates able to pass the computer simulation was subjected to interview for final evaluation.

## CONCLUSIONS

This proposed approach is indeed innovative and useful, providing very significant support in the selection of research technicians. On the basis of the results coming from these preliminary experiences it is possible to say that these techniques could provide solid and reliable performance measurement baselines and to focus on some aspects, allowing to focus on some others thanks to traditional techniques.

The extension of candidates' number applying distributed procedures for screening can add more opportunities for evaluation of researchers and also to improve their preparation. Meanwhile this approach allows a quantitative evaluation that could also be used by the candidates as feedback to improve their skills identifying precisely their errors.

## REFERENCES

- Amico V., Guha R., Bruzzone A. G. (2000) "Critical Issues in Simulation", Proceedings of SCSC2000, Vancouver, July
- Bara B.(1995),"Cognitive Science: A Developmental Approach To The Simulation Of The Mind", Lawrence Erlbaum Ass., ESussex
- Berkowitz L. (2000) "Causes and Consequences of Feelings", Cambridge Press, Cambridge
- Bruzzone A.G., M.E. (1998) "Upgrading Civil Protection Systems: The impact of New Techniques on Emergency Management Training", Proceedings ITEC98, Lausanne, April 28-30
- Bruzzone Agostino (1999) "Port Terminal Simulators as Main Supports for Design, Training & Management", Proc. of Port Logistics99, Alexandria, Egypt, Feb 14-16
- Bruzzone A.G. (1996), "Smart Integrated Interface for Evaluating and Supporting Human Performances", ESPRIT Information Technology: Basic Research Working Group 8467 Simulation for the Future: New Concepts, Tools and Application (SiE-WG), January 29, Bruxelles
- Bruzzone A.G. (1999) "Interactive Web-Based Tools for Education in Industrial Management", Proceeding of ITEC99, The Hague (NL), April
- Bruzzone A.G., Vio F., Capasso S. (2000) "MOSLES: Modelling & Simulation for Transport and Logistic Educational Support", Proceedings ITEC2000, The Hague, April
- Bruzzone A.G., Giribone P., Revetria R. (2000) "Simulation as Educational Support for Production and Logistics in Industrial Engineering", Proceedings of WinterSim, Orlando, December
- Bruzzone A.G., R.Mosca, A.Naamane, C. Frydman, N.Giambiasi, G.Neumann, D.Ziems, S.Capasso, R.Signorile, S.Ghosh, A.Nisanci, P.Elfrey, C.Briano, M.Brandolini (2001) "Logistics & Education in IEPAL International Cooperation", Proc. of HMS2001, Marseille, Oct.
- Giribone P., Bruzzone A.G. (1997) "Production Games: An Improvement in the Education and Training for Engineers working in Production and Logistics", Proceedings of European Simulation Symposium, ESS'97, Passau, Germany, October
- Giribone P. & A.G.Bruzzone (1995) "Training System through Simulation and AI Techniques of Process Plant Management", Proc. of FAIM'95, Stuttgart, June 28-30
- Harrington H. J., Tumay K., (2000), "Simulation Modeling Methods: To Reduce Risks and Increase Performance", McGraw-Hill, New York
- Klaus G, Troitzsch K.,(1999), "Simulation for the Social Scientist", Open Univ.Press, Buckingham
- Lord R. G., Klimoski R. J., Kanfer R., (2002), "Emotions in the Workplace", Jossey-Bass, SF
- Mosca R., Giribone P. & A.G.Bruzzone (1995) "Simulator OOModeling for Educational purposes in the Industrial Plant Sector", Proceedings of WMC95, Las Vegas, Nevada (USA), January
- Mosca R., Bruzzone A.G. & Costa S. (1996) "Simulation as a Support for Training Personnel in Security Procedures", Proc. of Military Government & Aerospace Simulation, New Orleans LA, April 8-11
- Mosca R., P.Giribone & A.G.Bruzzone (1994) "Simulation & Automatic Parking in a Training System for Container Terminal ", Proceedings of ITEC94, The Hague, April 26-28
- Reason J., (1990), "Human Error", Cambridge University Press, Cambridge
- Sarason I.C., Spielberger C.D.(1989), "Stress and Anxiety", Hemisphere Publishing Co., Washington
- Viazzo S., Simeoni S., (2004) "Models for Improving Recruitment Procedures", Proceedings of MAS2004, Bergeggi, Italy, October
- Watts D. J. (1999) "Small worlds" Princeton University Press, Princeton

# **Agent-Based Simulation**



# A ROUTING ALGORITHM INSPIRED FROM A DISTRIBUTED AUTONOMOUS MULTI-AGENT SYSTEM –THE ANT COLONY

Ruchir Jha

Department of Information Technology  
Nirma University of Science and Technology  
Chharodi, Ahmedabad – 382 481, Gujarat, India  
E-mail: ruchir\_jha@yahoo.com

## KEYWORDS

Routing Algorithms, convergence rates, Ant Colonies, Meta Heuristics, base IRP.

## ABSTRACT

Contemporary routing protocols are found to be less resilient to pathological conditions involving load variations and changes in local link metrics. [2] As an example, the Routing Information Protocol (RIP) presents poor convergence rates. At the same time for the Open Shortest Path First (OSPF), which was designed to provide connectivity among network nodes and to reroute traffic in case of sporadic node or link failures, frequent link metric changes can lead to wildly oscillating routing tables. Such facts thus, necessitate an approach to routing, which shall overcome the shortcomings of the routing algorithms currently in use. The paper discusses meta-heuristics based on the behavior of real ants in real time ant colonies, for routing to address these issues.

## 1. INTRODUCTION

Behavioral [3,4] characteristics of social animals like ants, swarms etc. have long served as meta-heuristics for solutions to stochastic combinatorial optimization problems. This is because such approaches deal with problems like premature convergence, population explosion in the solution space, etc better than conventional distributed-GA approaches. This is because Swarms and Ant-Colonies function as a decentralized and autonomous multi-agent system, which consists of myriad simple and cooperative individuals. As individuals, swarm agents (or ant colony members) behave with simple rules and based on locally perceivable information. As a collection, they run in a common environment and collaborate to achieve remarkable global goals. Their intelligence lies in the interactions among individuals and between the individuals and the environment. Ubiquitous computing as this might seem, routing in large-scale computer networks thus becomes a proposed domain for application of swarm and ant-colony heuristics.

However the ant colony routing algorithm (Antnet-RSV) stated in this paper shall interact with an IRP (Internal Routing Protocol) setup, to build on its path base. The IRP deployed at Layer-3 provides for all the basic network information requirements of the Antnet-RSV layer, which the former will provide to the latter, using its custom flooding algorithms, link-setup and connectivity testing mechanisms. Please note that the Antnet-RSV layer requires a base-IRP layer to establish a where's who know-how for the given subnet under consideration.

It can be recommended to rely upon a distance-vector algorithm, for the same, which will also minimize the routing node's "neighbor-base" building time. More, the fact that this will keep the node's form of information more local than global, may just add on as a plus, during the path-exploration phase of the ant, as we shall see in later sections. This can be considered metaphorical with the ant knowing about the whereabouts of the food source, though finding the shorter route to the same is a different issue altogether, which shall be considered hereby. In the last section, I conclude by giving an algorithm based on Darwin's Theory of Natural selection, applied to, finding a routing solution from the given solution-space.

## 2. THE ANTNET-CL ALGORITHM

The algorithm reported here, based in a connectionless framework, follows the work of Di-Caro and Dorigo (Di-Caro, Dorigo-1998) and is informally summarized as follows:-

Each node of the network retains a record of packet destinations as seen on data packets passing through that node. This is used to launch periodically, however asynchronously, launch "forward" ants with destinations stochastically sampled from the collected set of destinations.

Once launched the forward ant uses the routing table information to make probabilistic decisions regarding the next hop to take at each node. While moving forward, on every node the ant takes a timestamp and the node-identifier information. The same is later used

to update the routing tables along the path followed. If a forward ant is found to be traveling in a loop, that is it re-visits a given node, it is killed. On reaching the destination node, the total trip time is estimated, and a backward ant is created. The backward ant returns using the same path as followed by the forward ant, though using a priority queue, and not the queue of the data-packets. At each node on its path, the backward ant makes updates to the node's routing table to reflect the relative performance of the path. When the backward ant reaches the source, it "dies", and this can be implemented as some variant of IP's TTL.

## 2.1. Analysis

However the above-stated schema experiences some shortcomings. The above scheme first of all assumes that the routing tables maintained at the nodes are accurate, which in reality is never the case. If a node instantaneously goes down, or in the face of some sporadic link failure, the time taken for the network to converge might induce substantial overhead. Moreover, for a large network, with many routing nodes, the ants going out on links might need to maintain information quite global in nature (viz. the number of nodes in the network). The amount of information carried by one particular ant agent in the network would be huge for a large network. Both of these fundamentally violate the base on which the routing metaphor with behavior of social insects, stands. (Sec-Introduction). This necessitates use of an IRP employing a DV algorithm (distance vector algorithm), which then emphasizes on the performance expectations of the ant!. As it would be surely uneconomical to use this algorithm, which might then do nothing except eat up a considerable slice of the node's processing time.

## 3. THE ANTNET-RSV ALGORITHM

The work described in this section derives its inspiration from [Gross et al 1992, Beckers et al 1989], though the fundamental approach used is discussed in [White 96]. The algorithm works in the framework of resource-reservation and hence the 'RSV'. This approach uses three types of ant agents, namely "explorers", "allocators", "deallocators". Explorer agents exhibit the foraging behavior of ants and preferentially follow trails of pheromones laid down by previous explorers. Allocator agents traverse the path determined by explorer agents and allocate the bandwidth on the links used in the path. Similarly, when the path is no longer required deallocator agents traverse the path and deallocate the bandwidth used on the links. When explorer agents reach their destination they backtrack along the route chosen and drop pheromone in order to mark the path. Upon arrival back at the source node a decision is made whether or not to

send an allocator agent. The decision is made based upon  $m$  previous allocator agents' paths. If  $p\%$  (where  $p$  is the pheromone intensity on the selected path) of the agents follow the same path, the path is said to have emerged and an allocator agent is created and enters the network in order to allocate bandwidth. Allocator agents traverse the path indicated by the highest concentrations of the pheromones dropped by their associated explorer agents.

- 1) *Create an explorer at a frequency 'f', with the destination selected, as a function of probability p, where*

$$P = \frac{\text{Traffic}_{i \rightarrow k}}{\sum_{s \rightarrow d} \text{Traffic}}$$

- 2) *Initialize stack S for ant agent 'A'*

**S [top] → 0;**  
**Top[node\_identifier]=current\_node**  
**Top [time\_stamp] = 0;**

- 3) *Choose next-hop on the basis of routing table entry in the currently visited node with a probability Q.*

$$Q = \frac{P_{I \rightarrow k} + \Psi [L_N]}{1 + \Psi | \text{Neighbors}_C - 1 |}$$

- 4) *Create backward ant B, such that the values of stack S are now popped, and the ant on its way back using high-priority queues updates its pheromone tables.*

**While (S [Top] < 0)**  
**Top--;**  
**Update (P\_table (m, n));**

- 5) *Create allocator ant at a time interval equal to some integral multiple of total time spent on exploration and scan the pheromone table. "Sniff" for highest pheromone value, initialize stack for allocator and put it on the link.*

[Fig1- Modified AntNET-RSV]

It is possible that network bandwidth has already been allocated by the time the allocator agent is sent and in this case the allocator agent backtracks to the source node rolling back resource allocation and decreases pheromone levels such that a later. A decision to re-send an allocator agent is made at a later time after a

back-off period has been observed. During the back-off period explorer ants continue to search for routes. Fig-1 provides the trace for the same.

In Step-3 of the above algorithm, the expression for 'Q' is explained hereby.

The Ant-Colony system is a society of cooperating individuals, where agents form their perceptions by making current observations at the same time, backtracking for previous perceptions formed, when a predecessor agent was faced with the same choice. This is effectively implemented with the help of this expression.

$P_{I \rightarrow k}$  : When an explorer reaches a given node, the probability with which it selects its next hop is given by this variable. Its value is given by:

**Pheromone (source [I]→destination [k])**

---

$\Sigma$  Pheromone (source→destination)

The pheromone values are nothing but the route preference values set by the backward ants, which had traversed the same route before the current explorer. As there can be many routes to the same destination, the above expression is evaluated.

$\Psi$ : This is the heuristic correction factor, and it has been experimentally determined as that for the value of  $\Psi$  ranging between 0.2 and 0.5, the algorithm converges optimally.

$L_N = 1 - \text{Queue (source[I] ->destination[k])}$

---

$\Sigma_{(n=1 \text{ to } \text{Neighbor}\{\text{current}\})} \text{Queue}^N$

As described above, the algorithm needs to maintain a perfect blend between the currently perceived values and those learnt from backtracking. The above value provides for the same. It's a measure of the queue-length on each outgoing interface of the given node, at the given instant.

**Neighbors  $c$ .**

It gives the number of neighbors of the current node, which the explorer is visiting at the given instant.

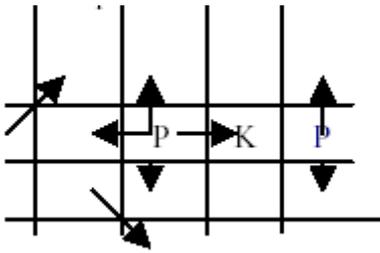
The given algorithm provides for filling the loopholes of the previous approach as discussed in 1.2. Furthermore, such an approach can be used for varied types of Internet applications seeking higher reliability of communications. However, there is one more issue to which the algorithm's resilience is not as it should be. After the exploration phase is completed, and when its time to send out the allocators, it's a possibility that the

agent's choice of path for resource allocation might wildly oscillate between two given choices, having the same pheromone intensity. One solution to this problem can be stated as assigning an 'age' parameter to every route added in the pheromone table. However for large networks with thousands of exploring agents returning to the same node at a given instant, this solution may not be quite convergent.

To combat such superfluous routing situations, an algorithm is stated in the next section, which draws its inspiration from the **Theory of Natural Selection**, and belongs to the Cellular Genetic Framework.

#### 4. ALGORITHM-SURVIVAL\_OF\_THE FITTEST

Genetic algorithms are efficient algorithms for searching complex fitness landscapes inspired by biological evolution. GA-s has been successfully applied to various complex optimization problems. However while using GA-s a problem presents itself, which is premature convergence, which is as a result of rapid loss of population diversity in the GA search space. And it's because of this that the search oscillates between sub-optimal solutions[Li, X. and Sutherland]. However CGA's are said to be effectively preserving population diversity. In a cellular GA, individuals are mapped onto a 2-dimensional lattice, each cell corresponding to each individual. The operators of selection and crossover are restricted within the local neighborhood of each individual. The "isolation-by-distance" method in a way allows a slow diffusion of good genes across the lattice, thereby producing individuals capable of adaptation to the then problem environment. The model I thus present is based on a selection method inspired by predator-prey interaction dynamics. Such dynamics contributes to maintain selection pressure and therefore population diversity. In this model, the prey, which represent, potential solutions, are free to move around a 2-dimensional lattice and breed with other prey. The selection pressure is maintained by predators, which roam around the lattice and consume the weakest prey in their vicinity. This kind of selection procedure efficiently maintains healthier prey, richer on the fitness factor over successive generations. The experimental results of this procedure have yielded more productivity than the conventional GA. The robustness of this mechanism lies in the fact that it relies on a dynamic spatial structure. This is quite in contrast to the conventional GA-static spatial structure, which one may observe. Furthermore this strategy maintains selection pressure by killing off prey rather than the direct replacement theory of replacing the least fit individual with the fitter one. This in a way helps implement an essential degree of population diversity along with check on a possible population explosion.



[Fig-2 2-dimensional lattice for the Prey-predator interaction]

This model consists of a 2-dimensional lattice where the prey and predator population resides. Both can occupy one cell at a time. Initially a large number of prey are distributed across the lattice. Then the model follows the following steps:

**Step: 1** Each prey is given the chance to move into one of the neighboring cells (where each cell has a probability of occupation set to 1/8 for a typical Moore Neighborhood) But we define a variable **RandomMoveProbability** = 0.5, for each prey individual to move, so that half of them would move one step on the lattice and the other half would stay where they were. This value is given to stop the scenario from slipping into a “population explosion” mode, where every prey breeds, as its got the chance to move. If the cell, a prey is trying to occupy is already occupied, the attempt is aborted, and the latter tries again, and the process may continue until a random upper bound (in our case 10) is reached. One might note, that varying the value of the upper bound, one might achieve a visible trade off between size and quality of population under consideration.

**Step: 2** Each one selects from its neighbors the fittest one and breeds with the same of course excluding its own-self. If the prey has no neighbors, its not allowed to breed, otherwise the prey breeds, using the traditional operators of mutation and crossover [as shall be discussed in the next section]. The offspring generated is placed on a randomly selected unoccupied cell, and the placement policy implicitly suggests that the offspring can face a radically different breeding environment as compared to its parents. This fulfills the objective of maintaining diversity among the existing population.

**Step: 3** We now enter the hunt. The predator kills its weakest neighbor prey. If a predator has no prey, it moves exactly the same way as the prey. However the predator can move more than once prey time step.

**Step: 4** Go back to Step-1 and re-enter the reproduction phase if sufficient number of evaluations is not reached, else continue. The importance of this step lies in the fact that we may need to balance between the predator and prey populations. For this we adopt the following formula where extent is the number of moves a predator can make before the prey. Thus one can envision the algorithm as an iterative switch between the prey and predators in locomotion.

$$\text{Extent} = (\text{Actual\_no\_of\_prey} - \text{Preferred\_no\_of\_prey}) / \text{Number\_of\_predators}$$

A predator can kill one prey per unit increment until the value of extent is reached. For example there are 450 prey, and the preferred number is 120 and the number of predators is 80 then the extent to which predators can stay in the hunt phase is ‘4’. However after the hunt phase, the algorithm again slips into the prey reproduction phase. A potent benefit of using the above equation is, that the algorithm always has a check on the total population size, as well as, even if the Actual\_no\_of\_preay value reaches an eventual minimum, the child prey generation is always being injected into the fitness landscape, producing better off springs.

#### Implementation:

The first question one might ask for the above stated procedure is what is prey and predator in the court of routing optimization.

Let

$W_{\max}$  = maximum weight associated with a link

$W_{\text{flow}}$  = weight associated with a given flow.

$\text{Pheromone}_{\text{path}}$  = Pheromone value in the routing table for the path under consideration.

$U_{\max}$  = maximum link utilization

$\mu_{\text{Rerouted}}$  = no. Of rerouted flows associated with the given change made in routing pattern

Thus, consider the following objective function:-

$$\Phi = (W_{\max} * U_{\max} + W_{\text{flow}} * \mu_{\text{Rerouted}}) / \text{Pheromone}_{\text{path}}$$

A deficiency of OSPF based traffic-engineering lies in the transient behavior while changing routing pattern from one metric to another. After metric modifications are detected, ISPF routers may plausibly distribute the new link state information after having performed recomputation through the shortest path algorithm. During this transition phase inconsistencies might arise, affecting active connections, which need to be rerouted which in turn maybe through increased packet loss or

packet reordering. Therefore its advisable to consider the amount of rerouted traffic or the number of affected flows, while considering the problem of making routing algorithms adapt and converge. Thus the same is considered in the expression of  $\phi$ .

#### Prey:

The objective function  $\phi$  can be vectorized to represent a prey individual like:

$(W_{max}, U_{max}, W_{flow}, \mu_{Rerouted}, Pheromone_{path})$

Such prey values can be distributed across the 2 dimensional lattices as a random function like:

**Distribution (I, J) = Random (I\*j) [Prey (I, J)]**

#### Predator:

The predator should logically consist of two modules. Finding its weakest prey, and second to kill it. This can be implemented as a set of 2 functions as below:

**Hunt (I, J) = (I / U<sub>rerouted</sub>)<sup>p</sup>**

As one may possibly want to minimize the maximum number of flows rerouted we choose the inverse of this parameter as our fitness function. This way routing solutions with smaller maximum link utilizations receive higher fitness values, and thus have a higher chance to be reproduced when a new generation is being set up.

In order to influence the reproduction phase we apply power scaling to the above function. With  $p < 1$ , we can achieve that fitness values of bad solutions are increased relatively to the best ones, thus avoiding that they die out too fast and that the optimization procedure converges too early. For  $p > 1$ , the gap between good and bad solutions is increased, forcing the process to converge optimally.

#### 4.1. Breeding- Crossover and Mutation

In this section, we adopt a real coded GA for the predator-prey model; each prey individual represents a chromosome, which in turn is a vector of genes, which are floating point numbers. The CGA based on the predator-prey model works similar to its binary counterpart, except that the crossover and mutation operations are slightly different. The real coded crossover involves two functions. The first behaves similar to the standard crossover operator. The difference is that instead of swapping binary values the values in the slots of floating point array or in other words the gene segments present on chromosomes is swapped. For example if we have two parents  $p1 = (x1, x2, x3, \dots, xn)$  and  $p2 = (y1, y2, \dots, yn)$ , and the crossover

point is between  $X1$  and  $X I + 1$ , then one child corresponds to  $C1 = (x1, x2, \dots, y I + 1, \dots, yn)$  and  $C2 = (y1, y2, \dots, x I + 1, \dots, xn)$ . We apply this operator to 50% of the prey population. The second operator is called as blend operator (BLX- $\alpha$ ) first introduced by Eshelman and Schaffer. BLX- $\alpha$  generates a child  $c1 = (C1, C2, \dots, CN)$ , such that  $Ci$  is a randomly chosen floating point number from the interval

**[Min I -  $\Delta * \alpha$ , Max I +  $\Delta * \alpha$ ]**. Its been their observation that highly optimal convergence rates are achievable for  $\alpha = 0.5$ .

#### 5. CONCLUSION

The algorithm discussed in this paper can be effectively used to resolve the situation encountered by Antnet-RSV, when the pheromone values, found by the allocator agents for the same destination, are same, for two distinct paths.

#### 6. FURTHER ENHANCEMENTS

Implementing an implied metaphor of the ant-routing mechanism would be, implementing the evaporation of pheromone trails on the links, in order to minimize the  $\mu_{rerouted}$  value. This is because, in the real ant colony system, pheromone trails being liquid in nature, evaporate after a given time interval, thus forcing the ants to make the same routing decisions again.

#### 7. REFERENCES

- Schoonderwoerd. 1996. Hewlett Packard Labs, Bristol-UK, Ant System Implementation- HP-Report.
- Goss S.; Beckers R.; Deneubourg J.L.; Aron S.; Pasteels J.M. 1990. "How Trail Laying and Trail Following Can Solve Foraging Problems for AntColonies, in Hughes R.N. (ed.) NATO ASI Series, Vol. G 20, Behavioural Mechanisms of Food Selection, Springer Verlag, Berlin.
- Grassé P.P., 1959. *La reconstruction du nid et les coordinations inter-individuelles chez Bellicositermes natalensis et Cubitermes sp. La theorie de la stigmergie: Essai d'interpretation des termites constructeurs. In Insect Societies*, Vol. 6, pp. 41-83.
- Hölldobler B. and Wilson E.O., 1994 *Journey to the Ants*. Bellknap Press/Harvard University Press.
- Shapiro J. A. 1988. *Bacteria as multi cellular organisms*, Scientific American, 1988 pp. 82-89.
- White A.R.P., 1996. *Routing with Ants*, Nortel internal report.
- Eshleman Schaffer. 1990. Real coded Genetic Algorithms for real-time problem schema
- Wright-A. 1991. Genetic Algorithms for real parameter optimization.
- Li, X. and Sutherland, S. (2004). A Real-Coded Cellular Genetic Algorithm Inspired by Predator-Prey Interactions", In Kay Chen Tan, Meng Hiot Lim, Xin Yao,

and Lipo Wang, editors, Recent Advances in Simulated Evolution and Learning, Advances in Natural Computation, pages (to appear). World Scientific 2004

**RUCHIR JHA** was born in Ahmedabad, Gujarat, India, in 1983. He went to Nirma University, where he

is currently an undergraduate student enrolled in the Bachelor of Engineering in Information Technology course. His interests lie in the fields of Artificial Intelligence and routing simulations. He is currently working as a trainee in Hewlett Packard-India.

# STRUCTURAL VALIDATION OF SYSTEM DYNAMICS AND AGENT-BASED SIMULATION MODELS

Hassan Quadrat-Ullah  
York University  
School of Administrative Studies  
4700 Keele St. Toronto, ON, M3J 1P3, Canada  
Email: [hassanq@yorku.ca](mailto:hassanq@yorku.ca)

## KEYWORDS

Simulation; System Dynamics; Structural Validity

## ABSTRACT

Simulation models are becoming increasingly popular in the analysis of important policy issues including global warming, population dynamics, energy systems, and urban planning. The usefulness of these models is predicated on their ability to link observable patterns of behavior of a system to micro-level structures. This paper argues that structural validity of a simulation model -right behavior for the right reasons- is a stringent measure to build confidence in a simulation model regardless of how well the model passes behavior validity tests. That leads to an outline of formal structural validity procedures available but less explored in system dynamics modeling 'repertoire'. An illustration of a set of six tests for structural validity of both system dynamics and agent-based simulation models follows. Finally, some conclusions on the increased appeal for simulation models for policy analysis and design are presented.

## 1. INTRODUCTION

For the remaining pages follow the general guidelines below: Models have been developed and applied to both operational problems as well as policy issues. However, the need of and the evaluation criteria of model validation differs for each case. For instance, in the case of operational problems, the results of a model can be accepted or rejected by exposing the results to a face validity test (Hermann 1967; Emshoff and Sission 1970). In a face validity test, experts assess how the model and its results are close to the real system. Model solutions can be tested in real world environments: e.g., another service window can be opened in the bank; efficiency of the oil refinery can be enhanced under the recommended actions; or inventory control system can be used improve customer satisfaction (Gass 1983).

In contrast, the majority of policy models such as system dynamics (SD) type model and agent-based models are built for the analysis of policy, exploration of possible future scenarios, and management purposes

(Gass 1983; Sterman 1984; Oliva 2003; Scholl 2001). From policy research perspective, modeling resolutions to important issues including global warming, population dynamics, energy systems, and urban planning simply defy a face validity test. Instead, for policy models, the key issue in validation is deciding (i) if the model is acceptable for its intended use, i.e., does the model mimic the real world well enough for its stated purpose (Forrester 1971; Goodall 1972; Forrester and Senge 1980) and (ii) how much confidence to place in model-based inferences about the real system (Barlas 1989, 1994; Curry et al. 1989). In order to assess the theoretical content of a policy model, it is imperative to look at the modeling process itself. Therefore, before we could attempt to illustrate the validation for SD models, it is crucial to examine SD modeling process first.

The appeal of SD models in the analysis of policy and managerial issues is due to their ability to link observable patterns of behavior of a system to micro-level structure and decision making processes. In order words, SD models are causal models (Barlas 1989). The crux of SD modeling process is to identify how structure and decision policies help generate the observable patterns of behavior of a system and then identified structures and decision policies be implemented. Therefore, the identification of the appropriate structure is the first step in establishing validity of a SD model. Once the structural validity of a SD model is sufficiently established, behavior validity - how well the model-generated behavior mimics the observed behavior of the real system - is assessed to achieve the overall validity of the model or to build confidence in the model (Gass 1983; Sterman 1984). In fact the validation process becomes iterative: structural validity-behavior validity-structural validity.

Since structural validity involves stakeholders of the model: modelers, clients, and policy researchers, I argue that structural validity is a stringent measure to build confidence in a SD model regardless of how well the model passes behavior validity tests. The second objective of this paper is to illustrate by the way of examples how some of the tests that already exist in the SD validation "repertoire" can help increase confidence in policy models. It is hoped that policy modelers, as a result of our illustrations, will appreciate the usefulness

of already existing but less explored tests in validation of policy models.

For the discussion of this paper, model refers to a SD type simulation model. However, there exist strong similarities between SD and agent-based modeling approaches: (i) both are unique in modeling nonlinear, complex systems such as urban planning systems, (ii) both assume that micro-structures of a system are responsible for its behavior, and (iii) both aim at discovering leverage points in complex systems, modelers of agent-based models seek them in rules and agents, while SD modelers do so in the feedback structure of a system (Scholl 2001). Therefore, arguments made and the validity procedures illustrated in this article should equally benefit agent-based modeling community. This paper is organized as follows: In § 2, an argument that structural validity is a stringent measure to build confidence in SD type models is established. Structural validity procedures are described in § 3. § 4 provide an illustration of structural validity tests. Conclusions are presented in § 5.

## 2. STRUCTURAL VALIDITY AS A STRINGENT MEASURE FOR A MODEL VALIDATION

In general, validation of SD models draws on two fundamental assumption of SD modeling process: (1) SD models are built to fulfill a purpose, and (2) structure of the model drives its behavior (Forrester 1961). SD modeling process begins with ‘conceptualization’ of the policy issue and produces a ‘quantitative computer simulation model’ for policy assessment and design. The purpose of the model informs the construction of both qualitative and quantitative model.

Since its inception, SD has linked the validation of a model with its “purpose”. As Forrester emphatically states that the validity of model should be judged by its suitability for a particular purpose and validity, as an abstract concept divorced from purpose, has no useful use (Forrester 1961). This view of model validation is widely shared by other modelers and policy scientists (Barlas and Carpenter 1990; Holling 1978; Overton 1977). Forrester and Senge (1980) stress that a model is built for a purpose and its validity is determined by the extent to which it satisfies that purpose.

Although SD modeling process is iterative in nature, essence of a SD type model lies in how well the problem has been conceptualized and causal relationships are identified or the qualitative model is constructed. It is the qualitative modeling stage that takes the temporal precedence over the quantitative modeling stage of any SD modeling endeavor: you have to have a conceptual model ready before any effort to realize a computer simulation model could ensue. At the qualitative modeling stage, focus is on (i) having

appropriate representation of the problem, and (ii) identifying the causal relationships between the elements of the conceptual model. If problem is either misrepresented or the causal relationships in the model are faulty, model generated data or model’s recommendations would simply be misleading. Or in Barlas’s words, you will get “right behavior for the wrong reasons” Therefore, structural validity: “right behavior for the right reasons” becomes the core of the SD modeling validation process (Barlas 1989).

Moreover, model validation depends on the cultural context and background of the model builders and model users. It depends on whether one is an “observer” (e.g., an academic researcher) or an “operator” (e.g., a decision maker who must act without waiting for data of further analysis (Greenberger et al. 1976). Nevertheless, involvement of stakeholders in the modeling process results in the increased credibility of the model (Kleindorfer et al. 1998). Again it is the conceptual model building stage of SD modeling process where the involvement of stakeholders is prominent: e.g., model assumptions and model boundary: what to model and what not to model is decided based on clients’ needs and model builders’ approach to modeling. Thus, the conceptual modeling stage allows realize the expertise of the relevant stakeholders and hence increase the likelihood of the acceptance of the model-based recommendations (Coyle and Exelby 2000). Consequently, structural validity that assesses the validity of the conceptual model becomes a stringent measure to build confidence in a SD model. It must be emphasized here that in no way I am discounting the usefulness of behavioral validity of a SD model. Instead, I want to highlight the significance of structural validity, often less explored in SD model validation endeavors.

## 3. STRUCTURAL VALIDITY PROCEDURES

Identification of the appropriate structure, responsible for the ‘right’ behavior, is a multidimensional process: problem representation, logical structures, and mathematical and causal relationships. Forrester and Senge (1980) discussed some of the tests used for structural validation of a SD model:

*Boundary adequacy:* Whether the important concepts and structures for addressing the policy issue are endogenous to the model?

*Structure verification:* Whether the model structure is consistent with relevant descriptive knowledge of the system being modeled?

*Parameter verification:* Whether the parameters in the model are consistent with relevant descriptive and

numerical knowledge of the system?

*Dimensional consistency:* Whether each equation in the model dimensionally corresponds to the real system?

*Extreme conditions:* Whether the model exhibits a logical behavior when selected parameters are assigned extreme values?

Barlas (1989) has demonstrated that behavior sensitivity test, originally suggested by Forrester and Senge (1980) as a behavior validity test, can detect major structural flaws of the model despite the fact that model can generate highly accurate behavior patterns. He termed it as a *structurally-oriented behavior test*: Whether the real system would exhibit a similar high sensitivity to those parameters to which model behavior displays high sensitivity.

#### 4. AN ILLUSTRATION OF STRUCTURAL VALIDITY TESTS

All the tests listed in §3 have been applied to evaluate the structural validity of a system dynamics model MDES RAP: a model for understanding the dynamics of electricity supply, resources and pollution (Qudrat-Ullah and Davidsen 2001). These tests by no means are exhaustive but constitute the core of battery of tests for the structural validity of SD type simulation models. The purpose of the model is to assess the impact of investment incentives on electricity-generating technology mix and emissions level, over the long term (the simulations runs from 1980 to 2030). MDES RAP is a dynamic general disequilibrium representation of Pakistan's electricity supply sector, excluding nuclear generation. An illustration of the applicability of structural validity tests to MDES RAP, one-by-one, follows. Although MDES RAP is not an urban planning model *per se*, structural validity tests being demonstrated here are applicable to any simulation model build to support policy decision making in complex dynamic systems with uncertain data including urban planning systems.

##### **Boundary Adequacy**

Consistent with the purpose of MDES RAP, all the major aggregates: electricity demand, investment, capital, resource, production, environment, and costs and pricing are generated endogenously. Only one variable, GDP is exogenous variable. The historical GDP of Pakistan is represented annually from 1980 to 2000 and linear extrapolation is used for the remaining years.

##### **Structure Verification**

The structural verification is of fundamental importance in the overall validation process. For the structural

verification of MDES RAP, a two-pronged approach was applied. First, during the construction of the model, we utilized (i) the specific case-Pakistan's data (or available knowledge about the real system), and (ii) the sub-models/ structures of the existing models of the domain, as given in Table 1. The causal relationships developed in the model, which were based on the available knowledge about the real system, provided a sort of 'empirical' structural validation. The adopted sub-models of the existing models of the domain served as a 'theoretical' structural validation (Forrester and Senge, 1980).

Table 1: Adopted Structures in MDES RAP

Structures/ Concepts	Remarks
Investment incentive dynamics (Dyner and Bun, 1997)	Causal structure was adopted
Substitution mechanism between electricity and oil (Davidsen, 1989)	Structural formulation was adopted
Production capital structure (Moxnes, 1990)	Structural formulation was adopted
Gross margin (Serman, 1980)	Structural formulation was adopted

##### **Parameter Verification**

The values assigned to the parameters of MDES RAP are sourced from the existing knowledge and numerical data form case-Pakistan's data. For illustration purpose, Table 2 lists some of the parameters, their values and the source.

Table 2: Some Parameters of MDES RAP and Their Assigned Values

Parameters in the Model	Assigned Values	Source
Time to Adjust Investments	2 (years)	(PEY, 1990; PEY, 1991; PEY, 1997)
Average Physical Life of Capital (oil)	30 (years)	
Average Physical Life of Capital (hydro)	40 (years)	
Target Limit for CO <sub>2</sub> Emission	20.20 M tons	
Construction Delay for Power Plant (oil)	4 (years)	
Construction Delay for Power Plant (hydro)	6 (years)	
Fuel Efficiency	0.4 (%)	
Safety Margin for Resource Inventory	0.5 (year)	
Operating Cost (oil)	0.57 (\$/MWh)	
Operating Cost (hydro)	0.22 (\$/MWh)	

## Dimensional Consistency

Dimensional consistency test requires that each mathematical equation in the model be tested if the measurement units of all the variables and constants involved are dimensionally consistent: in (apples) = out (apples). For instance, the following equation represents one of the equations of MDES RAP. This equation describes that share of each competing electricity generating technologies (EnergyTechShare) in the new capital investments being made is dependent on two factors: (i) the coefficient for the distribution of  $\alpha$  and (ii) the cost of electricity generating technology (CostOfElectTech).

$$\text{EnergyTechShare} = \text{EXP}(-\alpha) * \text{CostOfElectTech}$$

Is this equation dimensionally consistent? To answer, we need to know (i) Is the value of  $\alpha$  based on the real system? and (ii) What is the dimension of the dimension of  $\alpha$ ?

The value of  $\alpha$  is estimated based on the variation in the fuel costs of electricity generation technology, in Pakistan. We considered all 17 locations of thermal power plants, where the fuel is consumed to generate the electricity. The fuel costs at each of these sites were obtained to estimate the value of  $\alpha = 0.249$  (MWh/\$). No if we do the dimensional analysis of the equation above, we can have:

$$\begin{aligned} [\text{dimensionless}] &= [(\text{MWh}/\$) * (\$/\text{MWh})] = \\ &[\text{dimensionless}] \end{aligned}$$

Thus, not only the value of  $\alpha$  is based on the existing knowledge of the real system but also the equation is dimensionally consistent.

Both the extreme conditions test and the structurally-oriented behavior test are explained in detail in Qudrat-Ullah (2004).

In summary, the structure of MDES RAP was exposed to all these tests for overall structural validity. Based on these evaluations, we have strong confidence in MDES RAP's ability to generate "right behavior for right reasons".

## Structural Validation of Agent-based Simulation Models

In agent-based modeling, agents are seen as the generators of emergent behavior in a given space (Holland 1999). In Holland's view, the interactions between the agents are nonlinear and the overall behavior of the system cannot be obtained by summing the behaviors of the isolated agents. On the other hand, in SD "feedback" structures are seen as intrinsic in real systems and the generators of the aggregate system behavior (Richardson 1992). Thus, both the modeling

approaches aim at discovering leverage points in complex aggregate systems, modelers of agent-based models seek them in rules and agents, while SD modelers do so in the feedback structure of a system (Scholl 2001). In Scholl's words, "At the very least, it will be insightful to compare the aggregate behavior and emergent influence on the environment of agent-based models with the predictions of aggregate-level feedback models regarding the same subject area". Therefore, it is prudent to apply structural validation tests illustrated in the previous section on agent-based models. In fact, only after successful structural validation of models, any meaningful comparison could ensue.

## 5. CONCLUSION

Although structural validity tests constitute but one of two general types of tests required to build confidence in a SD type simulation model, these tests nevertheless are the core of SD modeling validation process and have temporal precedence over the other type of tests: behavior validity tests. Illustrations provided through the applications of six tests in this paper can help the modelers (and users) in policy domain including urban planning to lend an effective and tangible support to the process of building confidence in a simulation model.

Informed by the 'purpose' and structurally tested simulation models, be it SD type or agent-based type, should result in the increased appeal for simulation models for policy analysis and design. The policy issues exist. The simulation models are being built. Validation need and challenges are being met. Policy analysis simulation modeling community owes no apology to those who would only believe in face validity testing alone.

## REFERENCES

- Barlas, Y. 1989. "Multiple tests for validations of system dynamics type of simulation models." *European Journal of Operational Research* 42 (1), 59-87.
- Barlas, Y. 1994. "Model validation in system dynamics". In *Proceedings of the 1994 International System Dynamics Conference*, E. Wolstenholme and C. Monaghan (Eds.). Sterling, Scotland, 1-10.
- Barlas, Y. and Carpenter, S. 1990. "Philosophical roots of model validation: Two paradigms." *System Dynamics Review* 6(2), 148-166.
- Coyle, G. and Exelby, D. 2000. "The validation of commercial system dynamics models." *System Dynamics Review* 16(1), 27-41.
- Curry, G. L., Deuermeyer, B. L. and Feldman, R. M. 1989. "Discrete Simulation". Holden-Day, Oakland, CA.
- Davidson, P. 1989. "A petroleum life cycle model for the United States with endogenous technology, exploration, recovery, and demand." Working Paper # 1910-89. MIT, Boston.

- Dyner, I. and Bun, D. 1997. "A system simulation platform to support energy policy in Columbia". In *System Modeling for Energy Policy*, D. Bunn, and I. Dyner (Eds.). John Wiley, Chichester, 259-271.
- Emshoff, J. R. and Sission, R. L. 1970. "Design and use of computer simulation models". Macmillan, New York.
- Forrester, J. W. 1971. "The model versus modeling process." *System Dynamics Review* 1(2), 133-134.
- Forrester, J. W. and Senge, P. M. 1980. "Tests for building confidence in system dynamics models." *TIME Studies in the Management Science* 14, 209-228.
- Gass, S. I. 1983. "Decision-aiding models: Validation, assessment, and related issues for policy analysis." *Operations Research* 31 (4), 603-631.
- Goodall, D. W. 1972. "Building and testing ecosystem models". in *Mathematical Models in Ecology*. J. N. J. Jeffers (Ed.). Blackwell, Oxford, 173-194.
- Greenberger, M., Crenson, M. A., and Crissey, B. L. 1976. "Models in the Policy Process". Russell Sage Foundation, New York.
- Hermann, C. 1967. "Validation problems in games and simulations." *Behavioral Science* 12, 216-230.
- Holland, J. H. 1991. "Emergence from Chaos to Order". Addison-Wesley, Reading, Mass.
- Holling, C. S. 1978. "Adaptive Environmental Assessment and Management". John Wiley & Sons, New York, NY.
- Kleindorfer, G. B., O'Neill, L., and Ganeshan, R. 1998. "Validation in simulations: Various positions in the philosophy of science." *Management Science* 44 (8), 1087-1099.
- Moxnes, E. 1990. "Interfuel substitution in OECD-European electricity production." *System Dynamics Review* 6(1), 44-65.
- Oliva, R. 2003. "Model calibration as a testing strategy for system dynamics models." *European Journal of Operational Research* 151, 552-568.
- Overton, S. 1977. "A strategy of model construction". In *Ecosystem Modeling in Theory and Practice: An Introduction with Case Histories*, C. Hall and J. Day (Eds.). John Wiley & Sons, New York, Reprinted 1990, University Press of Colorado, 49-73.
- PEY, 1990. "Pakistan Energy Yearbook 1989". Ministry of Petroleum and Natural Resources, Govt. of Pakistan, Islamabad.
- PEY, 1991. "Pakistan Energy Yearbook 1989". Ministry of Petroleum and Natural Resources, Govt. of Pakistan, Islamabad.
- PEY, 1997. "Pakistan Energy Yearbook 1989". Ministry of Petroleum and Natural Resources, Govt. of Pakistan, Islamabad.
- Qudrat-Ullah, H. and Davidsen, P. 2001. "Understanding the dynamics of electricity supply, resources, and pollution: Pakistan's case." *Energy* 26 (6), 595-606.
- Richardson, G. P. 1992. "Feedback Thought in Social Science and System Theory". University of Pennsylvania Press, Philadelphia.
- Scholl, J. 2001. "Agent-based and system dynamics modeling: A call for cross study and joint research". In *the proceedings of 34<sup>th</sup> Annual Hawaii International Conference on System Sciences (HICSS-34)*, Vol. 3, Maui, Hawaii.
- Sterman, J. D. 1980. "The use of aggregate production functions in disequilibrium models of energy-economy iterations". Working Paper # D-3234. MIT, Cambridge.

- Sterman, J. D. 1984. "Appropriate summary statistics for evaluating the historical fit of system dynamics models." *Dynamica* 10 (Winter), 51-66.

#### AUTHOR BIOGRAPHY



**HASSAN QUDRAT-ULLAH** was born in Gujrat, Pakistan and went to the Bahauddin Zakarya University, Multan, where he studied Applied Mathematics and obtained his M. Sc. degree in 1988.

He worked for several years for the Chashnupp Power Project before moving in 1996 to the University of Bergen, Norway where he completed his M. Phil. degree in System Dynamics. Hassan then joined in 1999 the NUS Business School, National University of Singapore for his PhD degree in Decision Sciences and graduated in 2002. He did post-doctoral fellowship at Carnegie Mellon University, USA in 2002-2003 before joining York University, Canada as Assistant Professor in Management Science in 2003. His research interests include system dynamics modeling and simulation; dynamic decision making; and computer-simulated interactive learning environments. His e-mail address is: hassanq@yorku. ca.

# INTELLIGENT AGENT CONTROL USING INDUCTIVE, DEDUCTIVE AND CASE BASED REASONING

Agris Nikitenko  
Riga Technical University, Department of Systems Theory  
Meza street 1 k.4, Riga LV-1048, Latvia  
E-mail: [agris.nikitenko@rembox.lv](mailto:agris.nikitenko@rembox.lv)

## KEYWORDS

Intelligent agents, autonomous intelligent systems, autonomous robots, agent based modelling.

## ABSTRACT

The paper deals with the problem of intelligent system's design for complex environments. A possibility to integrate several technologies into one basic structure that could form a kernel of an intelligent system or intelligent agent has been discussed. An alternative structure is proposed in order to form the basis of an intelligent system that would be able to operate in complex environments.

The proposed structure is very flexible because of features that allow adaptation via learning and adjustment of the used knowledge. Therefore, the proposed structure may be used in environments with stochastic features such as hardly predictable events or elements (intelligent entities). This paper discusses a possibility to use the proposed structure in order to model intelligent entities or entire intelligent systems for hardly predictable environments or environments with stochastic features in agent based modeling domains [Ref. 18,19]. The basic elements of the proposed structure have found their implementation in a software system and an experimental robotic system. The software system as well as the robotic system have been used for experimentation in order to validate the proposed structure - its functionality, flexibility and reliability. Both of them are shortly presented in the paper. The basic features of each system are presented as well. The most important results of experiments are outlined and discussed at the end of the paper. Some possible directions of further research are also sketched at the end of the paper.

## INTRODUCTION

The Artificial intelligence is one of the youngest branches of modern science. During a short period of time (lasting only several decades) there have been developed a lot of different technologies and approaches to solve various types of problems existing in the field of artificial intelligence. The complexity of those tasks that can be performed by intelligent systems is growing from year to year. Therefore, the range of application of artificial intelligence (AI) technologies has been significantly

widened. One of the challenging tasks that is always hard to accomplish is simulation of intelligent entities such as animals, controlling units or even humans. The task is even more challenging if the simulated environment is complex and has some stochastic features or entities with stochastic behavior. Another important direction related with the application of AI technologies is the simulation of a complex environment in order to collect all the necessary information about it and use the information to adjust some system for operation in a real environment. For instance, the simulation of a mars surface before launching the mars exploration robots to the red planet helps collecting important information in order to adjust the robotic system.

In both examples – simulation robot or intelligent entity should be able to learn and to adapt in order to deal with incompleteness of the available information and stochastic features of the environment.

Before trying to build a structure of an autonomous intelligent system, it is necessary to define the environment in which the system will operate. The basis of such a definition can be found in the assumption that every object can be described as a system [Ref. 1] Obviously, a complex environment can be described as a complex system. There are several basic features that define a complex system [Ref. 2, Ref. 3]:

- uniqueness – usually complex systems are unique or a number of similar systems is unweighted.
- hardly predictable – complex systems are very hard to predict. It means that it is hard to calculate the next state of a complex system if the previous states are known. The hard predictability may be related to the mentioned stochastic elements or features of the environment.
- an ability to maintain some progress resisting against some outer influence (including the influence of the intelligent system).

Of course, any complex system has a general feature of a system such as a set of elements, a set of relations or links [Ref. 3] that may alternate during the simulation process.

Obviously, if the modeled system operates autonomously in a complex environment, it has to form some model of the environment. It is not always possible to build a complete model of the environment for different reasons. That may be caused by a huge space of possible states of the environment (or even

infinite), expands or other reasons. It means that an intelligent system will use only an incomplete model of the environment during its existence.

The structure presented in the paper exploits an adaptation and an uncertain reasoning technique as general methods to deal with the incompleteness of the system's model of its environment. The proposed structure is built to address issues related with agent based modeling by offering a way to build intelligent and more or less autonomous software or hardware agents.

## **BASIC FEATURES OF AN INTELLIGENT SYSTEM**

In this section the basic features of the proposed structure are outlined and explained according to the previous research activities.

Summarizing, the basic features of the proposed structure are as following [Ref. 4]:

- An ability to generate a new knowledge from the already existing in the system's knowledge base. This ability can be achieved by means of deductive reasoning. In order to increase the efficiency a case based reasoning may be combined with deductive reasoning [Ref. 6]. This feature, obviously, includes also an ability to reason logically. The proposed structure does not state the kind of deductive reasoning that should be used. The only rule is that the selected deductive reasoning method has to address demands of a particular task. As it is described above, complex environments may be very dynamic and even with stochastic features. Therefore, some uncertain reasoning techniques may be the most suitable for complex environments. The experimental systems described below also have uncertain reasoning techniques implemented as a deductive reasoning module.
- An ability to learn. As it is assumed above, in complex environments the intelligent system eventually will not have a complete model of the environment. Therefore, the environment will be hardly predictable. Also complex environments are dynamic – in other words the system will face with new situations very often. Obviously, some mechanisms of adaptation should be utilized. From the point of view of intelligence an adaptation includes the following main capabilities: capability of acquiring new knowledge and adjustment of the existing knowledge. In other words, the inductive reasoning module refers to capability of acquiring new knowledge or learning. This feature may be implemented by means of inductive reasoning. During an operation the intelligent system collects a set

of facts through sensing the environment that forms an input for learning.

- An ability to reason associatively. This feature is necessary due to a huge set of possible different situations that the intelligent system may face with. For example, there may be two different situations that can be described by  $n$  parameters ( $n$  is a big enough number) where only  $k$  parameters are different ( $k$  is a small enough number). Obviously, these situations may be assumed as similar. Therefore, an associative reasoning is used – to reason about objects or situations that are observed for the first time by the intelligent system similarly to reasoning about the known situations and using knowledge about the known situations and object. The associative reasoning is realized through using associative links among similar objects and situations. Each object or situation may be accessed or identified by a set of features, thus this mechanism operates similar to the associative memory [Ref. 5, Ref. 8]. An issue about which objects and situations should be linked is conditioned by a particular task or goal of the system's designer.
- An ability to sense an environment. This feature is essential for any intelligent system that is built to be more or less autonomous. The feature also includes an ability to recognize objects / situations that the system has faced with as well as an ability to obtain data about new objects. All sensed data are structured in frames (see below). During the frame formation process the sensed environment's state is combined with system's inner state, thereby allowing the system to reason about the system itself as well as the relation between system's inner state and sensed environment's state. Also the sensed system's and environment's states are used to realize the feedback in order to adjust system's knowledge. Thus, the system's flexibility is increased.
- An ability to act. This feature is essential for any intelligent system that is designed to do something. If the autonomous system is unable to act, it will not be able to achieve its goals. Obviously, the system has to act in order to achieve its goals as well as to obtain the feedback information for readjusting its knowledge or to learn new knowledge. The way of acting and the purpose of acting vary depending on the goals of the system's designer or user.

The listed above features form the basis for an intelligent system that operates in a sophisticated environment. According to the features of complex systems that are listed above, any of them may be

implemented, as it is needed for a particular task. In other words, the implementation methods and approaches are dependant on the purposes of the system itself.

Nevertheless, the main question is how to bind all of the features in one whole – one intelligent system.

Obviously, there is a necessity for some kind of integration. There are many good examples of different kinds of integration. For example, so called soft computing which combines fuzzy logic with artificial neuron nets [Ref. 6] or case based reasoning combined with deductive reasoning [Ref. 7].

In order to adjust an intelligent system for some particular tasks different structures may be used [Ref. 14]. This paper presents one of the alternative structures that may be used in order to form a kernel of an autonomous intelligent system.

The proposed structure is based on intercommunicating architecture. In other words, the integrated modules are independent, self-contained, intelligent processing modules that exchange information and perform separate functions to generate solutions [Ref. 14].

## STRUCTURE OF THE INTELLIGENT SYSTEM

According to the list of very basic features there can be outlined the basic modules that correspond to the related reasoning techniques:

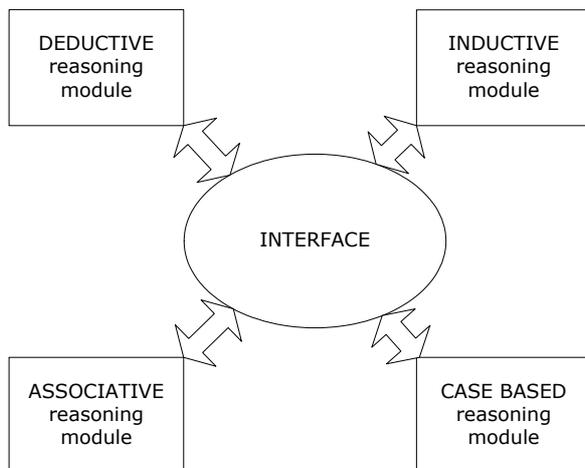


Figure 1. Basic modules

As it is outlined in the figure 1, there are four basic modules that form system's kernel. Each of the modules has the following basic functions:

- **Deductive reasoning module**  
This module performs deductive reasoning using *if..then* style rules. In order to implement adaptation functionality, this module may exploit some of uncertain reasoning techniques. In the proposed structure the main purpose of this module is to predict future states of the environment as well as the inner state of the system. During the reasoning process *if..then* rules are used in

combination with input data obtained from system's sensors. The proposed structure itself does not state what kind of deductive reasoning method should be used. It depends on particular goals of the system's designer. In both practical implementations a forward chaining certainty factor based reasoning had been used [Ref. 17]. If a task requires fuzzy reasoning or any other reasoning technique may be used as well.

- **Inductive reasoning module**  
This module performs an inductive reasoning or in other words – inductive learning. It learns new rules and adds them to the rule base. Also the incoming data from system's sensors are used.

Again the proposed structure does not state what kind of inductive learning technique is used. The only limitation is the requirement to produce rules that could be used by the deductive reasoning module. For example, if the fuzzy reasoning is used, then the result has to include fuzzy rules.

- **Case based reasoning module**  
Case based reasoning operates with “best practice” information that helps to reduce planning time as well as provides this information for modeller in explicit manner. As said above in complex environments there may be a lot of unique situations. To extract (or to learn) any rule an intelligent system needs at least two equal (or similar – the most part of feature (attributes) are equal) situations. It means that in complex environments a lot of situations experienced by the intelligent system may remain unused. Obviously, these unique situations (or cases) may be extremely valuable not only for the intelligent system but also for the modeller who uses the system. The case based reasoning module is involved to process and use these unique situations..

- **Associative reasoning module**  
This module links objects according to similarities among object features as well as situations, thus allowing to reason associatively. This module allows to reason about new situations or new objects using knowledge about similar objects or situations. It is an essential ability in complex and dynamic environments in order to increase a flexibility of the intelligent system.

Of course, the intelligent system needs additional modules that would supply it with the necessary information about the environment and mechanisms to perform some actions. Therefore the basic structure shown in figure 1 is complemented with additional modules. The enhanced structure is depicted in the following figure 2:

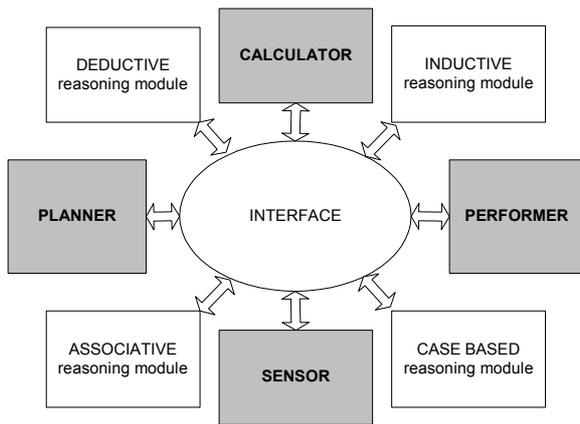


Figure 2. Enhanced structure

The additional modules are drawn in grey. Each of the additional modules has the following basic functions:

- Planner module. This module is one of the central elements of the system. Its main function is to plan future actions that lead to the achievement of goals of the system. In author's opinion an ability to predict future events or situation in the most obvious manner demonstrates the intelligence of the system. During the planning process three of the basic reasoning techniques are involved – deductive, case based and associative reasoning. The result of the planner is the sequence of actions that are expected to be accomplished by the system, thereby achieving its goals.
- Sensor module. The module's purpose is to collect information from system's sensors about the environment and the system's inner state. The sensed information is portioned in separate frames (see below) and forwarded to the interface (discussed later). Once the information is forwarded, it is available for other modules and may be used for readjustment of knowledge, for learning new knowledge or other purposes.
- Performer module. This module performs a sequence of actions that are listed in the plan. Also this module uses information about the current state of the system and environment in order to determine whether the instant actions can be accomplished. If not appropriate, feedback information is sent to the sensor module.
- Calculator module. This module collects and produces any quantitative data needed for reasoning. For example, in both implementations (see below) this module is used to calculate certainties of rules including rules newly generated by the inductive reasoning module. Thus, this module is directly involved in the knowledge readjustment process. The functionality of the module may be enhanced according to the

necessities of the particular tasks or goals of the system's designer.

As it is depicted in figures 1 and 2, all of the modules need some interface to communicate with each other. Therefore all of the modules use a central element – *Interface* in order to communicate to each other. They are not communicating to each other directly thereby a number of communication links is reduced as well as all of information circulating in the system is available for any module, if there is such a necessity.

A simplified structure of the interface is depicted in the following figure 3:

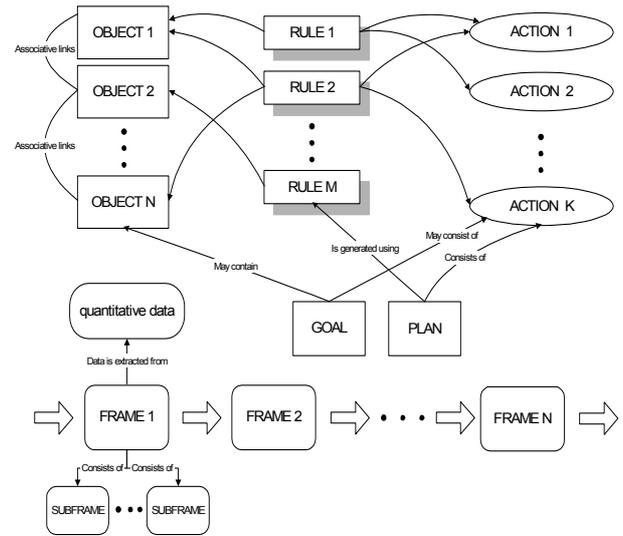


Figure 3. Structure of the interface.

The structure consists of several basic elements. The fundamental element of the whole structure is *an object*. **Object**. Objects are key elements in the interface structure. They correspond to some kind of entities in the environment (or the intelligent system). Every object is described with a set of features (attributes). Each feature has some value. As it is depicted the Figure 3 objects are linked to each other by associative links. These links form the basis for associative reasoning. When the intelligent system runs into a new situation some subset of objects is activated. These objects map out those entities that the intelligent system currently senses. If there is no rule that can be activated, then the intelligent system may try to activate the associated objects. Thus, the system can try to reason about the objects by using the associated rules. The result may be less feasible, but using association among objects the system can run out of the dead end situations. A mechanism of associative memory is very useful when the system works with noisy data. This mechanism allows to correct faults of the sensing mechanism [Ref. 8]. For example, if the input vector of the sense which corresponds to some entity has some uncertain or incorrect elements (attributes of object) then the system would not be able to activate any of the objects. In this case associative memory mechanism will activate the closest object

[Ref. 8] thus, the sensing error will be less significant for the reasoning process.

**Rules.** Rules are any kind of notation that represent causalities. In the practical experimentations a well known *if..then* notation was used. As it is depicted in the Figure 3 rules are linked to objects and actions. When the system activates objects by using associative links the linked rules also are activated, thus the system can scan a set of “associated” rules as well. This ability improves system’s ability to adapt. As it is depicted in Figure 3 rules are linked to actions. Rules (for example, those of type *if..then*) may include references not only to facts but also to actions. Thereby, rules through deductive reasoning are used in the planning process.

**Actions.** Actions are some kind of symbolic representation that can be translated by the intelligent system and cause the system to do something. For example “*turn to the right*” causes the system to turn to the right by  $90^0$ . Each action consists of three parts: precondition, body and postcondition. Precondition is every factor that should be true before the action is executed. For example, before opening the door it has to be unlocked. Body is a sequence of basic (or lower level) actions that are executed directly – for example, a binary code that forwarded to the motor controller causes the motors to turn (in the case of a robotic system). Post conditions are factors that will be true after the execution of action. For example, after opening the door, the door will be opened. It is important to stress that both implementations of the structure do not have any postcondition information at the beginning. All of the postconditions are learned during the system’s runtime.

**Frames.** Frames are some kind of data structures that contain the sense array from the environment and from the system. It means that frames contain snapshots of the environment’s and the system’s states.

As it is depicted in Figure 3 frames are chained one after another, thus forming a historical sequence of the environment’s and the system’s states. Frames can be structured in hierarchies. Hierarchies help to see values of features that cannot be seen in a single snapshot. For example, motion trajectories of some object, etc. Frames form an input data for learning (induction module) algorithms as well.

**Goal.** A goal is some kind of task that has to be accomplished by the system. It can be defined in three ways: as a sequence of actions that should be done, as some particular state that should be achieved or as a combination of actions and states. The third option is implemented in robotic systems described below.

**Plan.** A plan is a sequence of actions that is currently executed by the system (performer module). It may be formed using both basic and complex actions. After the plan is accomplished, it is evaluated depending on whether the goal is achieved or not thereby forming feedback information for the calculator module.

**Quantitative data.** This element is used to maintain any kind of quantitative data that is produced by the

calculator module and is used during the reasoning process. For example, it may contain certainties about facts or rules, possibilities, etc. Quantitative data are collected during the reasoning process as well as during the analysis of the input data – feedback data. All of those components together form an interface for the basic modules: Inductive, Deductive, Case based and Associative reasoning. Fundamental elements of the structure are implemented in the experimental software and robotic systems that are shortly described below.

## EXPERIMENTAL SOFTWARE SYSTEM

As it is mentioned above, fundamental elements of the proposed structure have found their implementation in experimental software systems. The implemented elements are: Case based reasoning, Inductive reasoning and Deductive reasoning. Deductive reasoning is implemented as a statement logic module based on the rules designed in *if...then* manner. The induction module is implemented using a very well known algorithm ID3 [Ref. 9]. It has its more effective successor C4.5 [Ref. 10]. The case based reasoning module is implemented using pairs {situation, action}. Each of pair has its value that determines how effective it is in a particular case. During the planning this value determines which actions are selected if more than one action may be selected. The maximum length of the plan is limited in order to avoid infinite planning due to lack of the necessary knowledge for successive planning. The environment is implemented as world of rabbits and wolf (domain of prey and hunter). There are also defined additional objects – “obstacles”. The number of rabbits and obstacles is not specified, thus allowing the definition of very complex configurations of the environment. The intelligent system is implemented as wolf. Rabbits may be moving or standing at one place. The wolf can catch rabbits. The wolf is moving according to its plan. The researcher (modeller) can freely change the number and place of obstacles and rabbits during the system’s runtime, thus acting as a stochastic element in the system’s environment. The goal also may be defined and changed at any time by the researcher (modeller). The intelligent system demonstrates the flexibility of the proposed structure. The results of experiments and experience accumulated during the implementation shows that new types of objects can be introduced without changing the proposed structure. It means that even being incomplete this structure demonstrates good ability to adapt and to operate.

## EXPERIMENTAL ROBOTIC SYSTEM

The implemented robotic system is the next step of validation of the proposed structure.

The robotic system is a semi autonomous intelligent system that encapsulates all of the mentioned above

elements of the proposed structure and the interface among basic modules described above.

The system's input consists of the following sensors:

- Eight IR (infrared) range measuring sensors;
- Electronic compass;
- Four bump sensors (two front and two rear micro switches);
- Four driving wheel movement measuring resistors (two for each driving wheel in order to achieve reliable enough measurements).

Two Basix-X [Ref. 15] microprocessors are used in order to communicate with PC and perform input data preprocessing and formatting. Prepared and formatted data as frames (see above) are sent to PC via RS-232 connection. All other modules of the intelligent system are implemented as a PC-based software that has user-friendly interface allowing simple following the system's operation, collection of research sensitive data, changing system's goals, etc....

The PC-based software system implements and demonstrates all of the structure's elements mentioned above. The system is built for research purposes only, in other words, it is built for experiments in order to examine and validate the proposed structure. Therefore, the system's user interface is built to be as flexible as possible allowing its user to manipulate with the robot's state, goals and results at runtime.

The most important features of the system are:

- Ability to work with multiple goals with mixed structure that may include – actions, states or both;
- Ability to adapt via using inductive learning algorithm C 4.5[Ref. 10];
- Case-Based reasoning is used to store information about best-practice cases and to use this information during the planning process;
- Ability to reason via using Certainty theory ideas, thus allowing addition of new rules that may be conflicting with existing rules in the rule base;
- Ability to reason using associative links among objects (situations);
- The system's knowledge and system's state relevant data are stored and processed in an explicit and easy way to follow manner, thus demonstrating advantages of the used knowledge based techniques.

It is important to stress that at the very beginning of the system's operation it has no information about the consequences of each action – it needs to learn them. But if it is necessary the system's rule base may be filled with rules, cases and other research relevant information, thus allowing to model some particular state of the system.

All of the necessary experiments are not finished yet - the system is under research process, but even the first experiments demonstrate a very good ability of adaptation and learning new sequences of actions in order to achieve goals. All of the conceived

experiments may be split into two major groups – experiments with goals that require matching of one action to one goal and goals that require more than one action in order to achieve the goal. Till now only the first group of experiments has been accomplished.

## **POSSIBLE ADVANCES AND FUTURE RESEARCH**

In order to queue actions one after another, thereby building a sequence of actions that lead to achievement of the goal a planning module is used. The planning module is built as a classic single goal planner. If there is more than one goal, the planner builds plans one by one for each goal. Usually for autonomous systems there is a necessity to work with more than one goal at the same time, for instance, to follow the charge of batteries and to avoid obstacles. If the system is a team member, the team's goals should be taken into account as well. In a common situation the avoidance of obstacles may have a higher priority than following the charge of batteries. If the battery charge is low the global priorities may change. In other words the system should be able to handle so-called global dynamics [Ref. 16] of the plans and their priorities. The mentioned ability is essential in such domains as robot soccer game or other similar very dynamic and complex environments. The proposed structure cannot handle globally dynamic planning yet. This is one of the directions for future research activities.

Obviously there may be tasks that cannot be accomplished using a single intelligent system. For example, simulation of some complex environments such as battlefields, transport systems etc. Therefore, more than one system should be used, thus forming a multiagent environment. There are different ways to design a multiagent system [Ref. 11, Ref. 12]. In different domains different solutions may be applied. Referring to the said above, another direction of farther research and experiments may be outlined – adjustments of the proposed structure in order to allow the intelligent system operate in a multiagent environment. One of the most sophisticated problems in such a multiagent environment is communication because every communication parameter may be variable. It is easy to imagine that two intelligent systems may try to communicate using different knowledge representation schemas, different knowledge, different communication protocols, different type of “conversation” (for example: questioning, answering, argumentation, etc.) or even different physical communication channels (radio frequency, verbal communication, etc.) [Ref. 13].

## **CONCLUSIONS**

Practical experiments show that the proposed structure may be very flexible even in very changing environments with variable goals. In both cases an

adaptation and ability to learn is essential and both of them are persistent in using the proposed structure. In agent based modeling the central element, obviously, in an agent and its behavior [Ref. 18, 9]. The proposed structure demonstrates capability to operate autonomously that makes it useful for agent based simulation in order to model intelligent entities or intelligent systems. Therefore it is reasonable to carry out further research and experiments in order to advance this structure. In spite of the first results that are quite promising there are still some open questions that should be answered in the further research activities. The most important questions are: how to enhance the system in order to control it using multiple goals or multiple (competitive) plans, what are the necessary improvements for effective operation in intercommunicating multiagent environments.

## REFERENCES

- [Ref.1]  
V.N. Volokova "System theory and system analysis methods for control and communication", Radio and Communication, 1983. (В.Н. Волкова «Теория систем и методы системного анализа в управлении и связи» Радио и Связь 1983.)
- [Ref.2]  
V.V.Druzinina, D.S. Kontorov "Systemtechnique", Radio and Communication, 1985. (В.В. Дружинин, Д.С. Конторов «Системотехника» Радио и Связь 1985.)
- [Ref.3]  
A.D.Cvirkun "Structure of complex system", Soviet Radio, 1975 (А.Д.Цвиркун «Структура сложных систем» Советское радио 1975.).
- [Ref.4]  
A.Nikitenko, J.Grundspenkis "The kernel of hybrid intelligent system based on inductive, deductive and case based reasoning", KDS2001 conference proceedings, St. Petersburg, 2001.
- [Ref.5]  
Bradley J. Rhodes "Margin Notes Building a Contextually Aware Associative Memory", *The Proceedings of the International Conference on Intelligent User Interfaces*, New Orleans, LA, 2000.
- [Ref.6]  
Akbarzadeh M.R., Tunstel E., Kumbla K. & Jamshidi M. "Soft Computing Paradigms for Hybrid Fuzzy Controllers: Experiments and applications", IEEE International Conference on Fuzzy Systems, 1998.
- [Ref.7]  
P.S.Rosenblum "Improving Accuracy by Combining Rule-based and Case based Reasoning", *Artificial Intelligence* 87, 1996.
- [Ref.8]  
A.M.Wichert "Associative Computation", der Fakultät der Informatik der Univesität Ulm Wrotslav, 2000.
- [Ref.9]  
J.R. Quinlan "Comparing Connectionist and Symbolic Learning Methods", Basser Department of Computer Science, University of Sydney, 1990.
- [Ref.10]  
J.R. Quinlan "Improved Use of Continuous Attributes in C4.5", *Journal of Artificial Intelligence Research*, 1996.
- [Ref.11]  
P.Scerri, D.Pynadath, M.Tambe "Adjustable Autonomy in Real-world Multi-Agent Environments", Information Institute and Computer Science Department, University of Southern California, 2000.
- [Ref.12]  
P.Stone, M.Veloso "Multiagent systems: A Survey from a Machine Learning Perspective", *IEEE Transactions on Knowledge and Data Engineering*, 1996.
- [Ref.13]  
J.Dale, E.Mamdeni "Open Standards for Interoperating Agent-Based Systems", 2000. (last changes)
- [Ref.14]  
S. Goonatilake, S. Khebbal "Intelligent Hybrid systems", John Wiley & Sons, 2002.
- [Ref.15]  
© NetMedia Inc. [www.basicx.com](http://www.basicx.com)
- [Ref.16]  
H. Burkhard, J. Bach, R. Berger, B. Brunswieck, M. Gollin "Mental models for Robot Control", "Advances in Plan-based control of Robotic agents", International seminar proceedings, Dagstuhl Castle, 2001.
- [Ref.17]  
J.J.Mahoney, R.J.Mooney "Comparing methods for refining certainty-factor rule-bases", *Proceedings of 11<sup>th</sup> international workshop on Machine Learning*, Rutgers, 1994.
- [Ref.18]  
A. Drogoul, D. Vanbergue, T. Meurisse "Multi-Agent Based Simulation: Where are the Agents?", *Proceedings of MABS'02, LNCS, Springer-Verlag. Bologna, Italy, 2002.*
- [Ref.19]  
P. Davidsson, "Multi Agent Based Simulation: Beyond social simulation", LNCS, Springer Verlag, 2000.

## AUTHOR'S BIOGRAPHY

**AGRIS NIKITENKO** was born in Riga and attended Riga Technical University, where he studied information technologies and obtained his Master degree in Computer Sciences in 2001. He has been working in IT sector for several years as a system analyst and project manager. Now he is employed as assistant by Riga Technical University, Department of Systems Theory. Also he is a doctoral student in Riga Technical University and is working on his thesis related with hybrid intelligent systems. The most important achievement in his thesis is implementation of a semi-autonomous robotic system that is shortly described in this paper. His e-mail address is [agris.nikitenko@rembox.lv](mailto:agris.nikitenko@rembox.lv)

# ECOLANG – A COMMUNICATION LANGUAGE FOR SIMULATIONS OF COMPLEX ECOLOGICAL SYSTEMS

António Pereira  
Pedro Duarte

CEMAS – UFP, University Fernando Pessoa  
Praça 9 de Abril, 349, 4249-004 Porto, Portugal  
E-mails: {apereira; pduarte}@ufp.pt

Luís Paulo Reis  
NIAD&R - LIACC

Faculty of Engineering - University of Porto  
Rua Dr. Roberto Frias, s/n 4200-465 Porto, Portugal  
E-mail: lpreis@fe.up.pt

## KEYWORDS

Ecological Modelling, Intelligent Agents, Simulation Models, Communication Language.

## ABSTRACT

This document introduces ECOLANG, a communication language used for simulations of complex ecological systems. The language was developed with the main purpose of interchanging information between a simulation application of aquatic ecosystems (EcoDynamo) and several external agents. These agents have some goals about the simulation results or the simulated system. Examples of the former may be a calibration agent, with the goal of optimising the fit between observed and simulated results, whereas an example of the latter may be an aquaculture/farming agent, looking for production optimisation. This document focus on messages exchanged between EcoDynamo and the two mentioned agents providing examples of the communication protocol. This work is part of a larger project, where ECOLANG will be used as a tool for “goal-oriented intelligent simulations”, towards sustainable management of coastal ecosystems.

## INTRODUCTION

Coastal ecosystems are but a small part of the area covered by the seas. However, their importance is very large, considering that over 60% of the world population lives within 60 km from the sea. These ecosystems are used for multiple purposes such as fishing, tourism, aquaculture, harbour activities and as the final destination of many pollutants. This diversity of uses implies complex and, in most cases, conflicting management decisions. The huge number of possible combinations generated by the different management decisions and options, the opposite interests of stakeholders and some institutional authorities and the slowness of the decision process make it very difficult to implement efficient automatic management policies.

In this context, the use of intelligent agents (Weiss 1999; Wooldridge 2002) seems to be very promising. Each institutional authority and stakeholder may be modelled as an agent, interact with simulation tools - able to predict the outcome of different decisions - get

results and configure new conditions for further simulation experiments.

The use of ecological modelling is a widespread practice in the management of coastal ecosystems. In particular, for coastal lagoons, located between land and open sea, where aquaculture plays an important role in local economy, the estimation of ecosystem carrying capacity can supply important indicators for ecosystem sustainability and returnable profit (Duarte et al. 2003).

Ecological models are simplified views of nature used to solve scientific or management problems. They include physical, chemical and biological processes to describe the main features of the ecosystem under analysis. One of the most important compromises is to find the optimal time and spaces scales for the model (Jørgensen and Bendoricchio 2001).

The simplest geometric representation is the zero-dimensional (0D) model, which simulates the system as a point and all changes are only time dependent. One-dimensional (1D) representation models assume that the system is characterized by a prevailing one-directional flow (horizontal or vertical) and the properties of the system vary along that direction and time. When the system is large enough to present sensible variation of the properties, vertical and horizontal division is required and two or three-dimensional (2D or 3D) representations are more common. Models of large lakes, coastal lagoons or river estuaries are examples of these representations.

The construction of mathematical models to make predictions about the evolution of an ecosystem, generally does not take into account the influence of the management decisions taken by the authorities. One possible way of doing that is to allow hypothetical authorities' decisions interacting with model simulation experiments.

This paper introduces ECOLANG, a communication language used to allow the interaction between ecological simulation experiments and several agents, representing either users of the system under simulation or applications designed to perform specific modelling tasks. The system conceptualisation, including a case study and the system architecture for the first application of ECOLANG is described in the next section.

Section 3 describes ECOLANG requirements, its messages format and types, and presents some examples. Finally, some conclusions are synthesised and directions of future research presented in the Conclusions section.

## SYSTEM CONCEPTUALISATION

### Case Study

The case study chosen for this work is Ria Formosa (south of Portugal); see Figure 1. This is a lagunary system with an area of c.a. 100 km<sup>2</sup>, mostly included in a Natural Park. It is used for tourism, fishing, and aquaculture and as an harbour. Local population fluctuates dramatically between winter and summer due to tourism. Therefore, it is an example of an ecosystem with a lot of conflicting uses.

Ria Formosa is being modelled as a 2D vertically integrated, coupled hydrodynamic-biogeochemical model, based on a finite difference bathymetric staggered grid (Vreugdenhill 1989) with 282 lines by 470 columns (132540 grid cells) and a spatial resolution of 100m (Figure 1).

The model is forced by tidal height at the sea boundary, light intensity, air temperature, wind speed, cloud cover and boundary conditions for some of the state variables. The time step used is 3 seconds and the variables simulated are: water temperature, current speed and direction, dissolved substances, suspended matter and phytoplankton concentrations and bivalve species biomass.

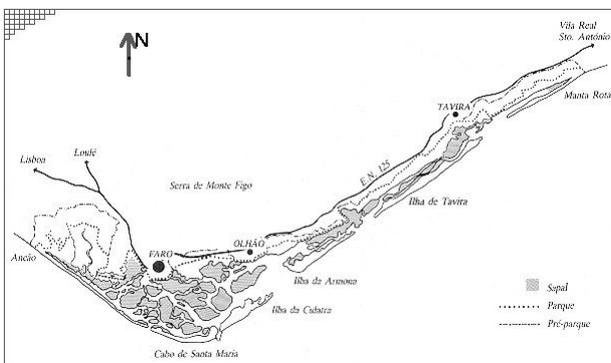


Figure 1 - Ria Formosa (South of Portugal). Part of the model grid is shown at the upper left corner (resolution of 100m)

### System Architecture

The architecture proposed in this work will be exemplified using two agents - a Calibration agent and a Shellfish Farmer agent (Figure 2). These two agents are the first developed entities interacting with the simulation application EcoDynamo (Pereira & Duarte in prep).

#### EcoDynamo Application

EcoDynamo (**E**cological **D**ynamics **M**odel) is an application built to enable physical and biogeochemical

simulation processes of aquatic ecosystems. It's an object oriented program application, built in C++ language, with a shell that manages simulation experiments runs, the graphical user interface, the communications between classes and the output devices where the simulation results are saved.

Different classes simulate different variables and processes, with proper parameter and process equations. Classes can be selected or deselected from shell dialogs determining its inclusion or exclusion in each simulation run of the model.

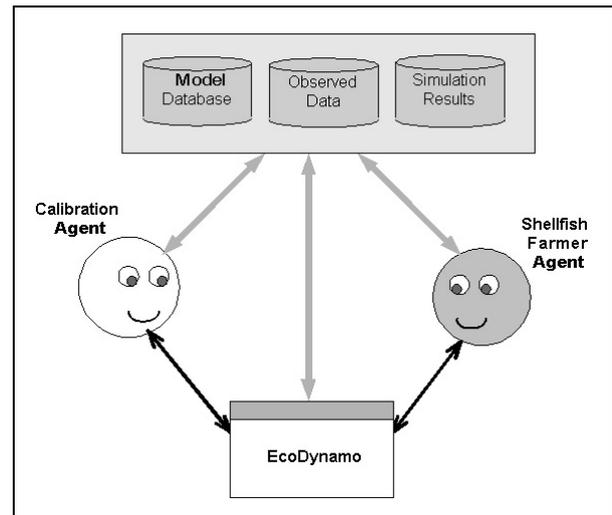


Figure 2 – Agent-based Simulation System Architecture

The simulated processes include:

- hydrodynamics of aquatic systems: current speeds and directions;
- thermodynamics: energy balances between water and atmosphere and water temperature;
- biogeochemical: nutrient and biological species dynamics;
- anthropogenic pressures, such as biomass harvesting.

The ecosystem characteristic properties are described in a model database: morphology (geometric representation of the model), dimensions (number of grid cells), classes, variables, parameter initial values and ranges.

In EcoDynamo there are different options available for the output of results – file output in text or hierarchical data format (\*.hdf) – and graphical outputs. Both \*.hdf and graphics output use MatLab<sup>®</sup> subroutines.

This application has an interface module (implementing the EcoDynamo Protocol based in ECOLANG) that enables communications with other programs for external control. For example, the simulation runs can be controlled by commands like start/stop/pause/restart/step simulation.

Simulation activity can be spied with the help of log files, activated previously before the simulation run.

### Calibration Agent

The Calibration Agent (CA) is an Intelligent Agent (Wooldridge 2002) that communicates through a LAN with the simulation application with full control over the simulation (classes, time, parameters, etc.).

Details concerning the calibration procedure followed by the CA are described in (Pereira et al. 2004). Its main algorithm is depicted in Figure 3.

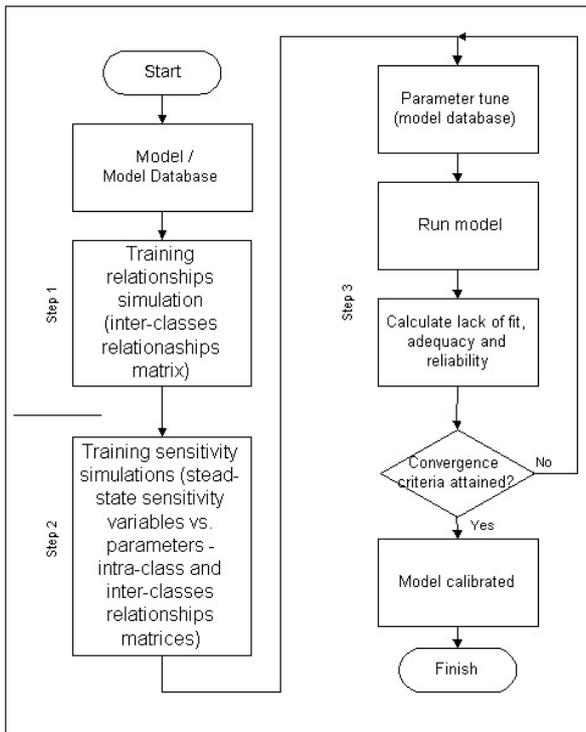


Figure 3 - Calibration Agent procedure diagram (from Pereira et al. 2004)

Its purpose is to tune model equation parameters in order to fit the model to observed data, towards model calibration and validation. The CA acquires knowledge about the behaviour of the system processes in three phases (Pereira et al. 2004):

- Building matrices that synthesize the interclass and inter-variable relationships;
- Analysing the intra and interclass steady-state sensitivity of different variables to different parameters and among variables;
- Iterative model execution, measuring model lack of fit, adequacy and reliability (Sholten and Van der Tol 1998; Scholten et al. 1998) until a convergence criteria is attained.

### Shellfish Farmer Agent

The Shellfish Farmer Agent works in a later phase, when the model is calibrated and validated. This agent simulates human reasoning and interaction with the

ecosystem using very simple rules. Its general objective is to maximise bivalve production without exceeding ecosystem carrying capacity (Duarte et al. 2003). The agent seed actions (introducing shellfish juveniles in the coastal region) will be restricted by environment rules that allow seeding in some areas but not in others.

### Ecological Modelling Language

ECOLANG is a high-level language capable of describing ecological systems in terms of regional characteristics, living agent's perceptions and actions. This language, along with a specific communication protocol, will enable the agents in the multi-agent system to understand each other in the ecological domain. It enables communications at different levels:

- Configuration – intended to select classes and change variables or parameter initial values to run the model.
- Execution – commands over the simulation execution (run, stop, pause,...).
- Statistics – intended to collect results from simulation experiments, either online or offline operation, compare results with previous simulation experiments or observed data and advise the configuration module the expected actions to take.
- Definitions – used to define regions of the model domain by names and aggregate several cells (or boxes) or regions into one region, according to common properties.
- Events – spontaneous messages that agents generate to inform some important events or results.

## ECOLANG MESSAGES

### Requirements

The format of the ECOLANG messages enables easy readability, simplicity and expandability. It also reflects the independence from any computational platform used for simulation. The main requirements for the message format are:

- Easy to extend to new concepts and definitions;
- Easily readable by the agents but also by humans;
- Robust enough to enable simple syntax validation;
- Easy to implement unambiguous error messages.

Message definitions used by ECOLANG follow the BNF formalism. Backus-Naur Form (BNF) is the best-known meta-language (a language used for describing languages) in the field of computer science. It was invented by John Backus and Peter Naur (Naur 1960) to describe the syntax of Algol 60 in an unambiguous manner.

ECOLANG notation is an extension to the original BNF formalism adding the following meta-symbols:

- { } used for repetitive items (one or more times);
- [ ] encloses types of values;
- Terminal symbols use bold face letters.

The basic message structure is:

```
<MESSAGE> ::= message (<ID> <SENDER> <RECEIVER>
  <MSG_CONTENT>)
<ID> ::= [integer]
<SENDER> ::= [string]
<RECEIVER> ::= [string]
<MSG_CONTENT> ::= <DEFINITION_MSG> |
  <ACTION_MSG> | <PERCEPTION_MSG>
```

<ID> - The identifier of the message sent by sender - each sender manages its own numbering system for the messages.

<SENDER> - The name of the agent or application that sends the message. It must not have spaces.

<RECEIVER> - The name of the agent or application target of the message. It must not have spaces.

<MSG\_CONTENT> - The message body properly.

Messages can be from three basic types: definitions, actions and perceptions. While definitions are generic messages (used by anyone of the communication partners), actions and perceptions are specific to each kind of involved agent.

ECOLANG syntax with some examples is demonstrated in the next paragraphs. The examples are for a Calibration Agent and a Shellfish Farmer Agent. The former includes Action and Perception messages that allow controlling model runs, to initialise model parameters and evaluate model results, respectively. The general approach is to compare these results with real data and change model parameters iteratively, until some convergence criteria are attained (Pereira et al. 2004). The latter example includes Action, Perception and Definition messages, to control model runs, analyse model results and define cultivation strategies, iteratively.

## Definitions

Messages used for definitions are limited, for the time being, to define regions. Each region will be referenced by a name, a specific type and a specific area. It can be defined, also, as a union of other regions:

```
<DEFINITION_MSG> ::= define (<REG_NAME>
  <REGION>)
<REG_NAME> ::= [string]
<REGION> ::= <REGION_TYPE> <REGION_AREA> |
  {<REG_NAME>}
```

The type of region defines land or water and, in this case, it will be characterized by its quality and the type and quality of its sediments:

```
<REGION_TYPE> ::= <LAND_REGION> |
  <WATER_REGION>
<LAND_REGION> ::= land
```

```
<WATER_REGION> ::= <WATER_CARACT>
  <SEDIMENT_CARACT>
<WATER_CARACT> ::= <SUB_INTERTIDAL>
  <WATER_QUALITY>
<SUB_INTERTIDAL> ::= subtidal | intertidal
<WATER_QUALITY> ::= <QUAL_SCALE>
<QUAL_SCALE> ::= excellent | good | poor
<SEDIMENT_CARACT> ::= (<SEDIMENT_TYPE>
  <SEDIMENT_QUALITY>)
<SEDIMENT_TYPE> ::= sandy | sand_muddy | muddy
<SEDIMENT_QUALITY> ::= <QUAL_SCALE>
```

The region area is the union of one or more simple regions, each one of these defined by a point or a basic polygon (rectangle, square, circle or arc):

```
<REGION_AREA> ::= {<SIMPLE_REGION>}
<SIMPLE_REGION> ::= <POINT> | (rect <POINT>
  <POINT>) | (square <POINT> <POINT> <POINT>
  <POINT>) | (circle <POINT> [real]) | (arc
  <POINT> [real] [real] [real] [real])
<POINT> ::= (point [real] [real])
```

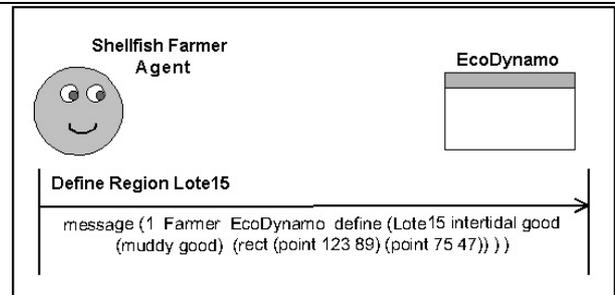


Figure 4 – Definition message

## Shellfish Farmer Agent

### Actions

The shellfish farmer agent cultivates bivalves and its actions are seed, inspect and harvest bivalves:

```
<ACTION_MSG> ::= <SEED_ACTION> |
  <INSPECT_ACTION> | <HARVEST_ACTION>
```

When the shellfish farmer agent wants to seed bivalves, it indicates the characteristics of the bivalves and where they will be seeded. To inspect, it indicates the region(s). To harvest, it indicates the region(s), the characteristic of the bivalves to harvest and when harvest should occur:

```
<SEED_ACTION> ::= seed (<REG_NAME> <TIME>
  <BIVALVE> <WEIGHT>)
<INSPECT_ACTION> ::= inspect (<REG_NAME>
  <TIME>)
<HARVEST_ACTION> ::= harvest (<REG_NAME> <TIME>
  <BIVALVE>)
<BIVALVE> ::= <BTYPE> <BCARACT>
<BTYPE> ::= mussel | oyster
<BCARACT> ::= <SHELL_LENGTH> | <DENSITY>
<SHELL_LENGTH> ::= (length [real])
<DENSITY> ::= (density [real])
```

The reference time can be immediately (now) or a value that is the number of seconds from January 1, 1970 00:00:

```
<TIME> ::= now | [integer]
```

### Perceptions

The perceptions of this agent are the answers to the actions that, previously, it has done:

```
<PERCEPTION_MSG> ::= <SEED_RESULT> |
<INSPECT_RESULT> | <HARVEST_RESULT>
```

The answer to the seed action can be positive or negative. The answer to the inspect action will be a message with bivalve characteristics. The answer to the harvest action can be negative or positive and, in this case, the kind and amount of bivalves harvested:

```
<SEED_RESULT> ::= seed_result (<ACTION_ID>
<ACTION_RESULT>)
<INSPECT_RESULT> ::= inspect_result
(<ACTION_ID> {<BIVALVE>})
<HARVEST_RESULT> ::= harvest_result
(<ACTION_ID> <ACTION_RESULT> <WEIGHT>)
<ACTION_RESULT> ::= ok | failed
<ACTION_ID> ::= <ID>
<WEIGHT> ::= [real]
```

The <ACTION\_ID> field identifies the <ID> of the action message.

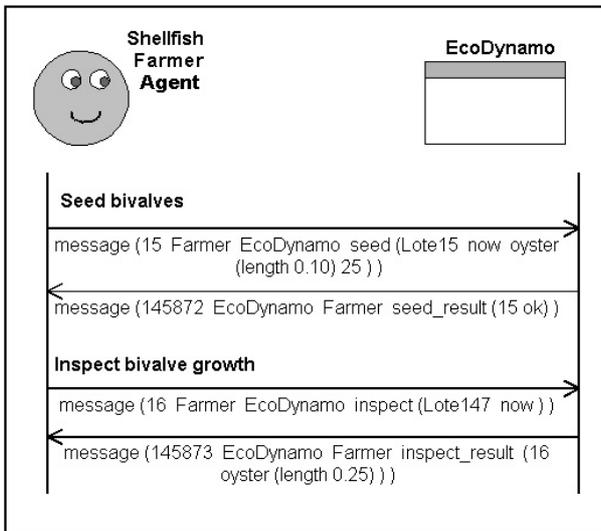


Figure 5 – Farmer Agent seeding and inspecting regions

### Calibration Agent

#### Actions

The calibration agent is designed for model calibration, in order to approximate simulation results to observed data. The actions allowed to this agent are divided in four different types:

```
<ACTION_MSG> ::= <MODEL_ACTION> | <EXEC_ACTION>
| <SPECS_ACTION> | <REG_ACTION>
```

- Actions to select the simulation model: open or close model, ask the model in simulation:

```
<MODEL_ACTION> ::= <OPEN_MODEL> | <CLOSE_MODEL>
| <GET_MODEL>
<OPEN_MODEL> ::= open_model <MODEL_NAME>
<MODEL_NAME> ::= [string]
<CLOSE_MODEL> ::= close_model
<GET_MODEL> ::= model_name
```

- Actions over the simulation execution: initialise, run, pause, step, stop:

```
<EXEC_ACTION> ::= initialise | run | stop |
step | pause
```

- Actions over the simulation model: select/enquiry classes, initial values for variables and parameters, time step and time interval and simulation sub-domain (part of the model grid):

```
<SPECS_ACTION> ::= <SP_CLASSES> | <SP_VARS> |
<SP_PARMS> | <SP_TIME> | <SUB_DOMAIN>
<SP_CLASSES> ::= <GET_CLASSES> |
<SELECT_CLASSES>
<GET_CLASSES> ::= get_available_classes |
get_selected_classes
<SELECT_CLASSES> ::= select_classes
{<CLASS_NAME>}
<CLASS_NAME> ::= ([string])
<SP_VARS> ::= <GET_CLASS_VARS> |
<GET_VAR_VALUE> | <SET_VAR_VALUE>
<GET_CLASS_VARS> ::= get_variables <CLASS_NAME>
<GET_VAR_VALUES> ::= get_variable_value
<CLASS_NAME> <VAR_NAME> <CELL>
<VAR_NAME> ::= ([string])
<CELL> ::= [integer]
<SET_VAR_VALUE> ::= set_variable_value
<CLASS_NAME> {(<VAR_NAME> <BOXES> [real])}
<BOXES> ::= (<SUB_DOMAIN>) | {(<CELL>)}
<SP_PARMS> ::= <GET_PARMS> | <SET_PARMS>
<GET_PARMS> ::= get_parameters <CLASS_NAME>
<SET_PARMS> ::= set_parameters <CLASS_NAME>
{(<PARAM_NAME> [real])}
<PARAM_NAME> ::= ([string])
<SP_TIME> ::= <GET_TIME> | <SET_TIME>
<GET_TIME> ::= get_time_spec
<SET_TIME> ::= set_time_spec <STEP>
<START_TIME> <FINISH_TIME>
<STEP> ::= [integer]
<START_TIME> ::= [integer]
<FINISH_TIME> ::= [integer]
<SUB_DOMAIN> ::= subdomain <DOMAIN>
<DOMAIN> ::= all | {(<REG_NAME>)}
```

- Actions over the output results: select registry variables, type, frequency, time interval and sub-domain of register and activate survey mode:

```
<REG_ACTION> ::= <REG_FILE> | <REG_VARS> |
<REG_LOG> | <REG_TIME> | <REG_TRACE>
<REG_FILE> ::= output_file <FILE_NAME>
<FILE_NAME> ::= ([string])
<REG_VARS> ::= <GET_VARS> | <SELECT_VARS>
<GET_VARS> ::= get_available_variables
<SELECT_VARS> ::= select_variables
<OUTPUT_TYPE> {(<VAR_NAME>)}
(<SUB_DOMAIN>)
<OUTPUT_TYPE> ::= file | graph | table
<REG_LOG> ::= log <LOG_TYPE> {(<LOG_STEP>)}
<LOG_TYPE> ::= xml | xls | txt
<LOG_STEP> ::= [integer]
```

```

<REG_TIME> ::= <GET_REG_TIME> | <SET_REG_TIME>
<GET_REG_TIME> ::= get_output_time
<SET_REG_TIME> ::= set_output_time <STEP>
    <START_TIME> <FINISH_TIME>
<REG_TRACE> ::= trace

```

*Perceptions*

The perceptions of the calibration agent could be answers to previous actions initiated by the agent itself or spontaneous messages sent by the simulation application when significant events occur. There are five types of perceptions:

```

<PERCEPTION_MSG> ::= <MODEL_RESULT> |
    <EXEC_RESULT> | <SPECS_RESULT> |
    <REG_RESULT> | <EVENT_MSG>

```

- Answers to actions over the model:

```

<MODEL_RESULT> ::= <OPEN_RESULT> |
    <CLOSE_RESULT> | <GET_RESULT>
<OPEN_RESULT> ::= open_result (<ACTION_ID>
    <ACTION_RESULT>)
<CLOSE_RESULT> ::= close_result (<ACTION_ID>
    <ACTION_RESULT>)
<GET_RESULT> ::= model (<ACTION_ID>
    <MODEL_NAME>)

```

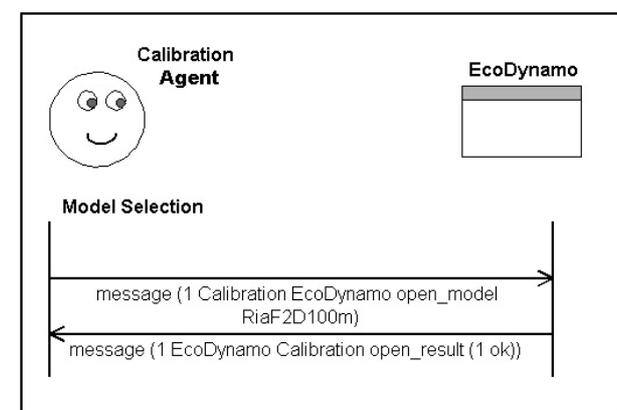


Figure 6 – Calibration Agent opens model “RiaF2D100m”

- Answers to actions over the simulation model:

```

<SPECS_RESULT> ::= <CLASSES_RESULT> |
    <VARS_RESULT> | <PARMS_RESULT> |
    <TIME_RESULT> | <SUB_DOMAIN_RESULT>
<CLASSES_RESULT> ::= <CLASSES_AVAILABLE> |
    <CLASSES_SELECTED>
<CLASSES_AVAILABLE> ::= classes_available
    (<ACTION_ID> {<CLASS_NAME>})
<CLASSES_SELECTED> ::= classes_selected
    (<ACTION_ID> {<CLASS_NAME>})
<VARS_RESULT> ::= <CLASS_VARS> | <VAR_VALUE> |
    <VAR_SET>
<CLASS_VARS> ::= variables (<ACTION_ID>
    {<VAR_NAME>})

```

```

<VAR_VALUE> ::= variable_value (<ACTION_ID>
    <VAR_NAME> <CELL> [real])
<VARS_SET> ::= variable_set_result (<ACTION_ID>
    <ACTION_RESULT>)
<PARMS_RESULT> ::= <CLASS_PARMS> | <PARMS_SET>
<CLASS_PARMS> ::= parameters_value (<ACTION_ID>
    {(<PARM_NAME> [real])})
<PARMS_SET> ::= parameters_set_result
    (<ACTION_ID> <ACTION_RESULT>)
<TIME_RESULT> ::= time_spec (<ACTION_ID> <STEP>
    <START_TIME> <FINISH_TIME>)
<SUB_DOMAIN_RESULT> ::= subdomain_result
    (<ACTION_ID> <ACTION_RESULT>)

```

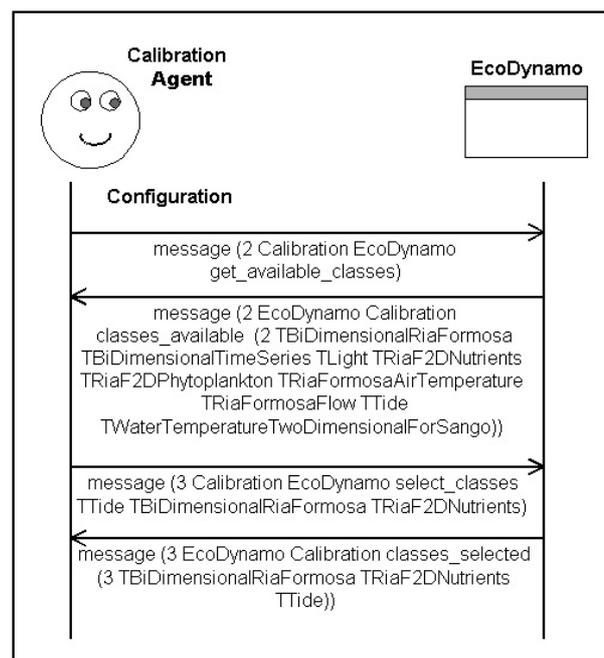


Figure 7 – Calibration Agent configuring simulation

- Answers to actions over the simulation execution:

```

<EXEC_RESULT> ::= exec_result (<ACTION_ID>
    <ACTION_RESULT>)

```

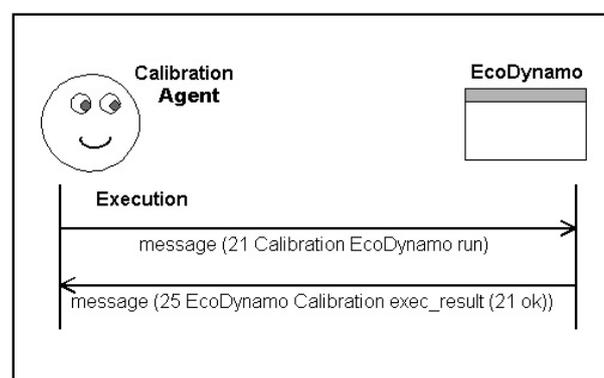


Figure 8 – Calibration Agent runs model

- Answers to actions over the output register:

```

<REG_RESULT> ::= <FILE_RESULT> |
  <REG_VARS_RESULT> | <LOG_RESULT> |
  <REG_TIME_RESULT> | <TRACE_RESULT>
<FILE_RESULT> ::= output_file_result
  (<ACTION_ID> <ACTION_RESULT>)
<REG_VARS_RESULT> ::= <GET_VARS_RESULT> |
  <SELECT_VARS_RESULT>
<GET_VARS_RESULT> ::= variables_available
  (<ACTION_ID> {<VAR_NAME>})
<SELECT_VARS_RESULT> ::= select_variables_result (<ACTION_ID>
  <ACTION_RESULT>)
<LOG_RESULT> ::= log_result (<ACTION_ID>
  <ACTION_RESULT>)
<REG_TIME_RESULT> ::= output_time (<ACTION_ID>
  <STEP> <START_TIME> <FINISH_TIME>)
<TRACE_RESULT> ::= trace_result <TRACE_STATUS>
<TRACE_STATUS> ::= on | off

```

- Spontaneous messages from simulation application:

```

<EVENT_MSG> ::= <REG_MSG> | <LOG_MSG>
<REG_MSG> ::= register (<REG_INDEX> <REG_TIME>
  <CELL> <VAR_NAME> [real])
<REG_INDEX> ::= [integer]
<REG_TIME> ::= [integer]
<LOG_MSG> ::= logger (<STEP_NR> <CLASS_NAME>
  <FUNC_TYPE> <DATA_CLASS> <VAR_NAME> <CELL>
  [real])
<STEP_NR> ::= [integer]
<FUNC_TYPE> ::= Inquiry | Update
<DATA_CLASS> ::= <CLASS_NAME>

```

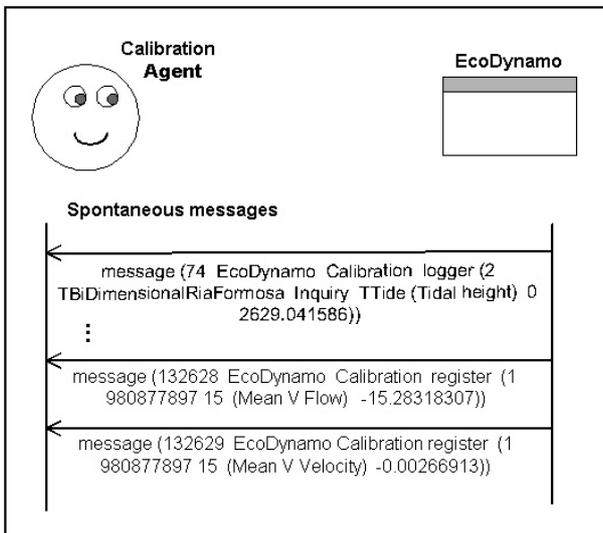


Figure 9 – Calibration Agent receiving spontaneous messages

## CONCLUSIONS AND FUTURE WORK

The definition of a communication language for simulation of complex ecological systems is the first step to an open road of possibilities to make the simulation experiments of ecological models more interactive and comprehensible to end users. With high-level messages,

it is possible to build intelligent agents that can inter-mediate the simulation experiments and the users.

With this language, it is possible to change some model characteristics and system configuration during simulation experiments. It may be very useful for the calibration of large and complex models – a generally hard and tedious process – if one may define an agent, capable of simulating the trial and error learning process that any modeller has to go through, when calibrating a model.

Generally, the management of coastal ecosystems may be done in many different ways and there is hardly one optimal solution, but most likely a “family” of “good” management options. Giving the large complexity of these systems and the numerous synergies between environmental conditions and management options, finding “good” choices cannot be reduced to simple optimisation algorithms, assuming linear or some defined form of non-linear relationship between a set of parameters, variables and goal seeking functions. Mathematical models may be very useful in finding “good” management solutions. However, finding these may require many trial and error simulation experiments and this is why using agents that may look automatically for the mentioned solutions may be advantageous. This will require the *a priori* definition of “good” solutions and constraints. For example, one may wish to increase aquaculture production but keeping water quality within certain limits for other uses. In any case, a high level language as ECOLANG is necessary to link simulation software with specialized agents.

## ACKNOWLEDGEMENTS

This work was supported by the DITTY project – “Development of an Information Technology Tool for the Management of European Coastal Lagoons under the influence of river-basin runoff”, contract number EVK3-2002-00084 EU (European Commission 2003), by the ABSES project – “Agent-Based Simulation of Ecological Systems” (FCT/POSI/EIA/57671/2004) and by the FCT research scholarship SFRH/BD/16337/2004.

## REFERENCES

Duarte, P.; R. Meneses; A.J.S. Hawkins; M. Zhu; J. Fang and J. Grant. 2003. “Mathematical modelling to assess the carrying capacity for multi-species culture within coastal waters”, *Ecological Modelling* 168 (2003): 109-143.

European Commission. 2003. “Development of an Information Technology Tool for the Management of European Southern Lagoons under the influence of river-basin runoff – The DITTY Project”. S.P.I.03.135, Directorate-General of Joint Research Centre, Institute for Environment and Sustainability.

- Jørgensen, S.E. and G. Bendoricchio. 2001. *Fundamentals of Ecological Modelling*, Elsevier Science Ltd, 3rd edition.
- Naur, P. (ed.). 1960. "Revised Report on the Algorithmic Language ALGOL 60.", *Communications of the ACM*, Vol. 3 No.5 (May), 299-314.
- Pereira, A. and P. Duarte. in prep. EcoDynamo – Ecological Dynamics Model Application, Technical Report, University Fernando Pessoa.
- Pereira, A.; P. Duarte and L. P. Reis. 2004. "Agent-Based Ecological Model Calibration – On the Edge of a New Approach". In: C. Ramos and Z. Vale (eds), *Proceedings of the International Conference on Knowledge Engineering and Decision Support*, pp. 107-113, ISEP, Porto, Portugal, July. ISBN: 972-8688-24-5.
- Scholten H. and M.W.M. Van der Tol. 1998. Quantitative validation of deterministic models: when is a model acceptable? In: *The proceedings of the Summer Computer Simulation Conference, SCS*, San Diego, CA, USA: 404-409, ISBN: 1-56555-149-4.
- Scholten H.; M.W.M. Van der Tol and A.C. Smaal. 1998. *Models or measurements? Quantitative validation of an ecophysiological model of mussel growth and reproduction*. Paper presented at the ICES Annual Science Conference, Cascais, Portugal.
- Vreugdenhil, C.B. 1989. *Computational hydraulics, An introduction*, Springer-Verlag.
- Weiss, G. (ed.). 1999. *Multi-Agent Systems: A Modern Approach to Distributed Artificial Intelligence*, MIT Press.
- Wooldridge, M. 2002. *An Introduction to Multi-Agent Systems*, John Wiley & Sons, Ltd.

## AUTHOR BIOGRAPHIES

**ANTÓNIO PEREIRA** was born in Porto, Portugal and obtained a degree in Electrotechnical Engineering and a MSc in Artificial Intelligence and Computation in the University of Porto. After graduation, he worked 15 years in EFACEC Electronic Systems (a portuguese industrial company) and 2 years in Siemens, Information and Communications group. Since 2003 he is dedicated to research in the field of Agent-Based Simulation systems and is a PhD student in Electrotechnical Engineering. His e-mail address is [apereira@ufp.pt](mailto:apereira@ufp.pt).

**PEDRO DUARTE** was born in Lisbon, Portugal, obtained his degree in Biology at the University of Lisbon and his PhD in Environmental Sciences at the New University of Lisbon, in 1995. He has been involved in several European projects doing research in the field of Ecological Modelling applied to environmental problems such as carrying capacity estimation for aquaculture. His e-mail address is [pduarte@ufp.pt](mailto:pduarte@ufp.pt).

**LUIS PAULO REIS** was born in Porto, Portugal and have a PhD in Electrotechnical Engineering (Coordination in Multi-Agent Systems) in the University of Porto. He has been researching in the area of (Multi-Agent) intelligent simulation for several years in different projects including FC Portugal simulated robotic soccer team – World and European champion in 2000. His e-mail address is [lpreis@fe.up.pt](mailto:lpreis@fe.up.pt) and his web page can be found at <http://www.fe.up.pt/~lpreis>.

# MULTI-AGENT SIMULATION OF DISPUTED MARKETING SITUATIONS

Yuri A. Ivashkin, Anton V. Shcherbakov, Elena A. Rogozhkina  
Moscow State University of Applied Biotechnology  
Russia, Moscow, Talalikhina str., 33  
E-mail: [ivashkin@msaab.ru](mailto:ivashkin@msaab.ru)

## KEYWORDS

Simulation, Intelligent agent, multi-agent system, agent-basic technology, marketing, price strategy.

## ABSTRACT

This article examines multi-agent simulation model of interaction and choice of optimum pricing strategy of firms - oligopolists during competitive struggle for profit and market share. Methods, algorithms and program realization in universal simulation system Simplex3 (Schmidt 1996, 2001) *agent-based simulation* of interaction of competing elements in disputed marketing situations are described. *The feature* of agent-oriented *models* is that they describe individual characteristics of all participants of the market (Gorni 1996, Weppner 1998) and their criterion functions on each step of interaction, instead of the average or generalized parameters of a condition.

The considered simulation model of the market with suppliers and buyers of production opens the **events** mechanism of the conflict and dynamics of interaction of competing elements - *intellectual agents* during his sanction and stabilization of a condition status of the marketing environment.

## INTRODUCTION

Marketing is dynamically developing industrial-organizational system functioning in conditions of information uncertainty, determined first of all, by rigid competition between firms-manufacturers and firms-resellers. However dynamics of the real market is in most cases unpredictable and its final condition status cannot be predicted analytically or by the logic analysis from an initial condition status as it grows out from multistage interaction of many active elements of the market environment and outage factors. A characteristic example is *the disputed market situation with contestant firms* - the manufacturers of the goods, struggling for maximum profit and a market share. In this case simulation of behaviour of each active element in conditions of *counteraction* is concerned with a choice of strategy of pricing with allowance for condition statuses and strategies of others oligopolists - competitors, their weak and strengths, the information on a financial position and features of marketing and productive activity.

*Price strategy* is determined by the purposes of the company - mastering and deductions of a determined

market share, maintenance of the planned volume of the profit, suppression of activity of competitors, etc. depending on character of demand, manufacturing costs and sale of goods, real value of the goods, strategy of competitors, etc.

*The initial price* is established, proceeding from product cost and the planned profit, solvency of the buyer and the price developed in the market; the price of indifference at which for the consumer its all the same, the goods of what firm to get. *The current and final prices* differ from initial depending on price strategy of firm-oligopolist and can be reduced in process of development and market saturation on a background of a competition or to raise at determined favorable situations.

The aim of *multi-agent modeling* is studying the influence of various types of price strategy on oligopol market per market share, profit and a sales volume depending on an established commodity price, market average price and qualities of the goods, strategy of competitors, heterogeneities of the market, not price factors and other marketing characteristics. For this purpose models and algorithms of *agent-based simulation* of interaction of competing elements - *the intellectual agents (IA)* varying the properties and behaviour depending on a condition status of other elements and the marketing environment are offered. Models of IA, describing individual characteristics of a condition status and behaviour of each participant of the market conflict on each step of interaction are consolidated in *multi-agent simulation model* of the marketing situation reproducing dynamic interaction of intellectual agents - contestant firms with an opportunity of identification of a condition status and forecasting of optimum price strategy.

Strategies of the resolution of conflict in all cases are reduced to two consecutive phases: *individual struggle* with counteractive party and in case of exhaustion of its opportunities - to search of the *cooperative joint solutions* conducting to compromise achievement of an overall aim. Thus the strategy realization in current environment may be achieved by a logic sequence of operations that are depends on individual agent parameters changing and estimation.

## SIMULATION MODELING OF OLIGOPOL DISPUTE

The block diagram of agent-oriented imitating model of oligopol market is submitted on fig. 1 and includes functional blocks of the firms-sellers struggling for

maximum profit and a market share, block *Market* and collective block *N* of buyers of *Buyer* production. Marketing process of stabilization of the market for two oligopolists with the subsequent occurrence of the third is considered. For each buyer his requirements to a product and firm are generated. For this purpose in basis component *Buyer* elements of a bidimensional array which rows correspond to criteria, and columns - to serial numbers of buyers are played. For every oligopolist in a random way, parameters of an offered product and accompanying services are played as well. Each firm is described in a separate basis component, accordingly *Firm1*, *Firm2*, *Firm3* with parameters of a condition status (the price, volume of tenders, discounts, advertising, quality, commodity range, the credit of trust, remoteness, delivery) in arrays *par [1..9]* and in mobile component *Query*.

In the transactive period the buyer, coming on the market (basis component *Market*), considers the tender

of each firm and chooses as much as possible corresponding to his requirements. Thus calculation of buyers and quantities of the purchased goods for each of firms is conducted. In the end of the transactive period totals are summed up, and the information on quantity of buyers and on remnants of the goods on warehouses is sent to each firm. Proceeding from this and taking into account the last experience (the sales volume and demand for the last periods), oligopolist chooses one of possible strategies for the following period of time with a variation of the price, tenders and not price factors. Variables of a condition status of basis components of structure of imitating model are (fig. 1):

*Cust F1-Cust F3* - number of clients, accordingly firms 1-3;  
*QueryStock1-QueryStock3*, *FirmStock* - stores of tenders on each firm and as a whole;  
*balances [1..3]* - array of remnants of the goods;

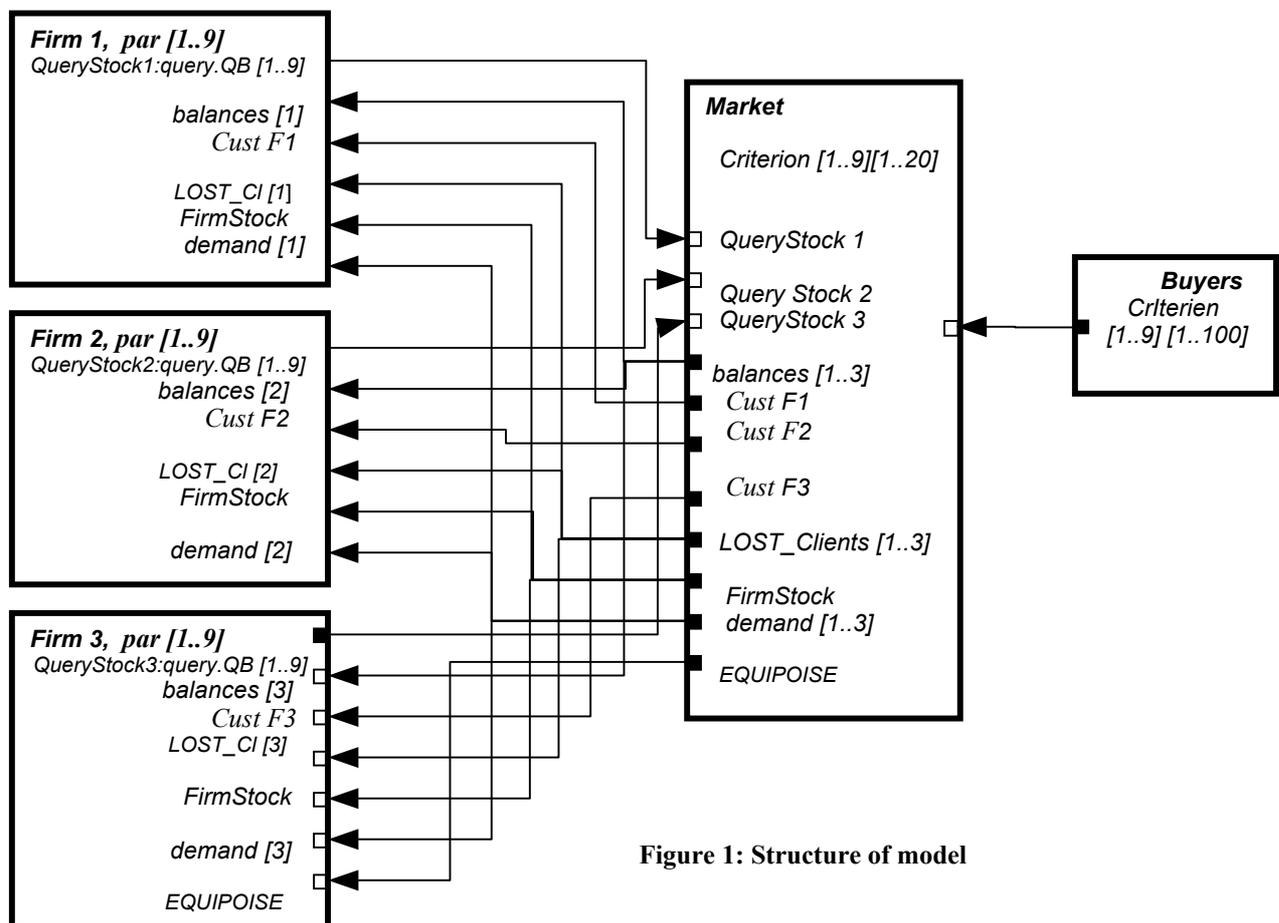


Figure 1: Structure of model

*LOST\_Clients [1..3]* - array, which elements shows whether there are at the competitor not catered clients;  
*EQUIPOISE* - the logic variable equal TRUE at an establishment in the market of balance of the prices;  
*Criteria [1..9][1..100]* - array of criteria of buyers;  
*demand [1..3]* - array, maintaining volume of missed sales on each firm.

### COMMON ALGORITHM OF MODELING STRATEGY

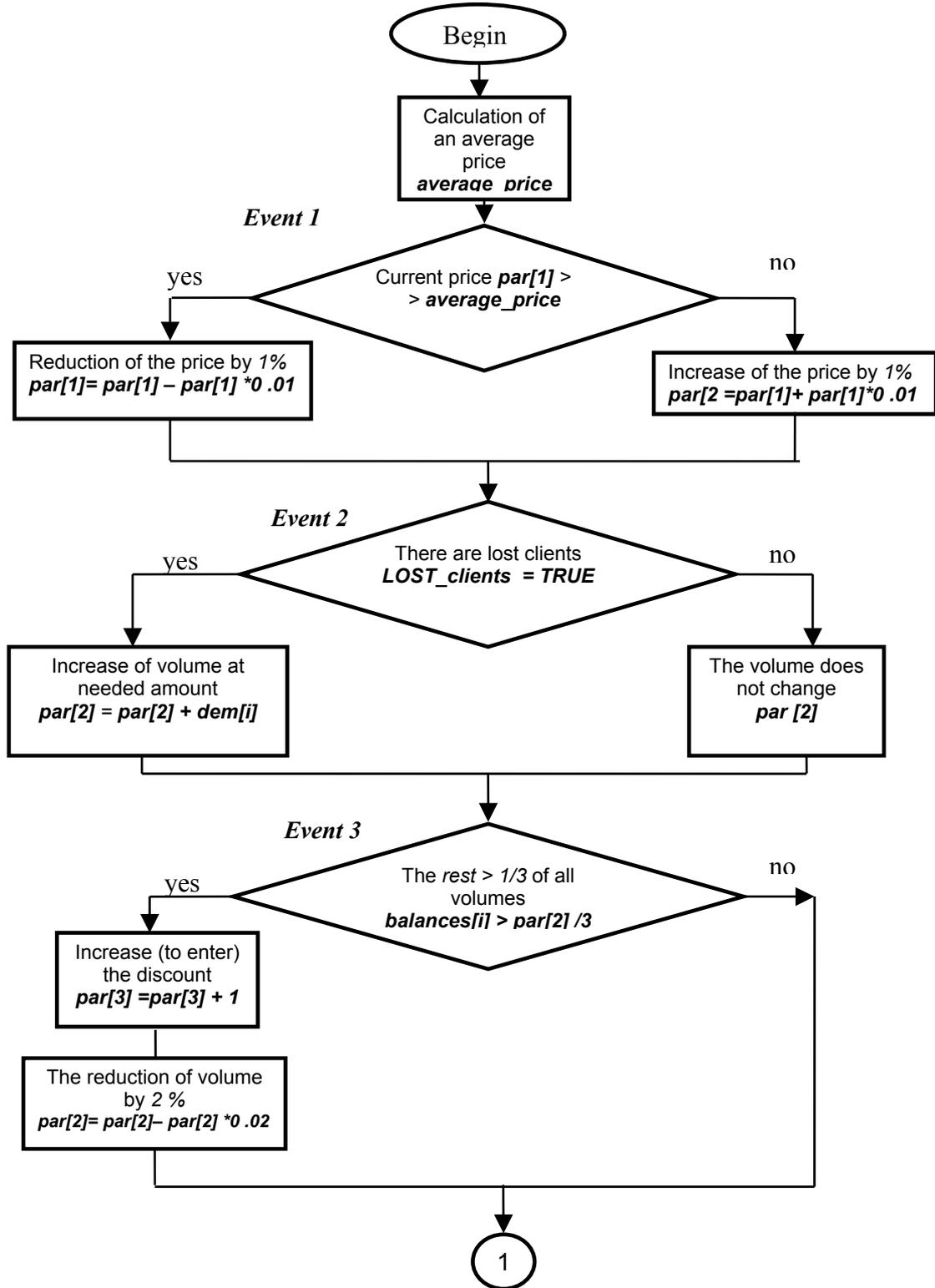
The algorithm of a choice of strategy is realized in language *Simplex-MDL* in universal system of imitating modeling *Simplex3* (Schmidt, 1996, 2001), developed in Passau university (Germany) and is reduced to reproduction of the following events

(Fig. 2).

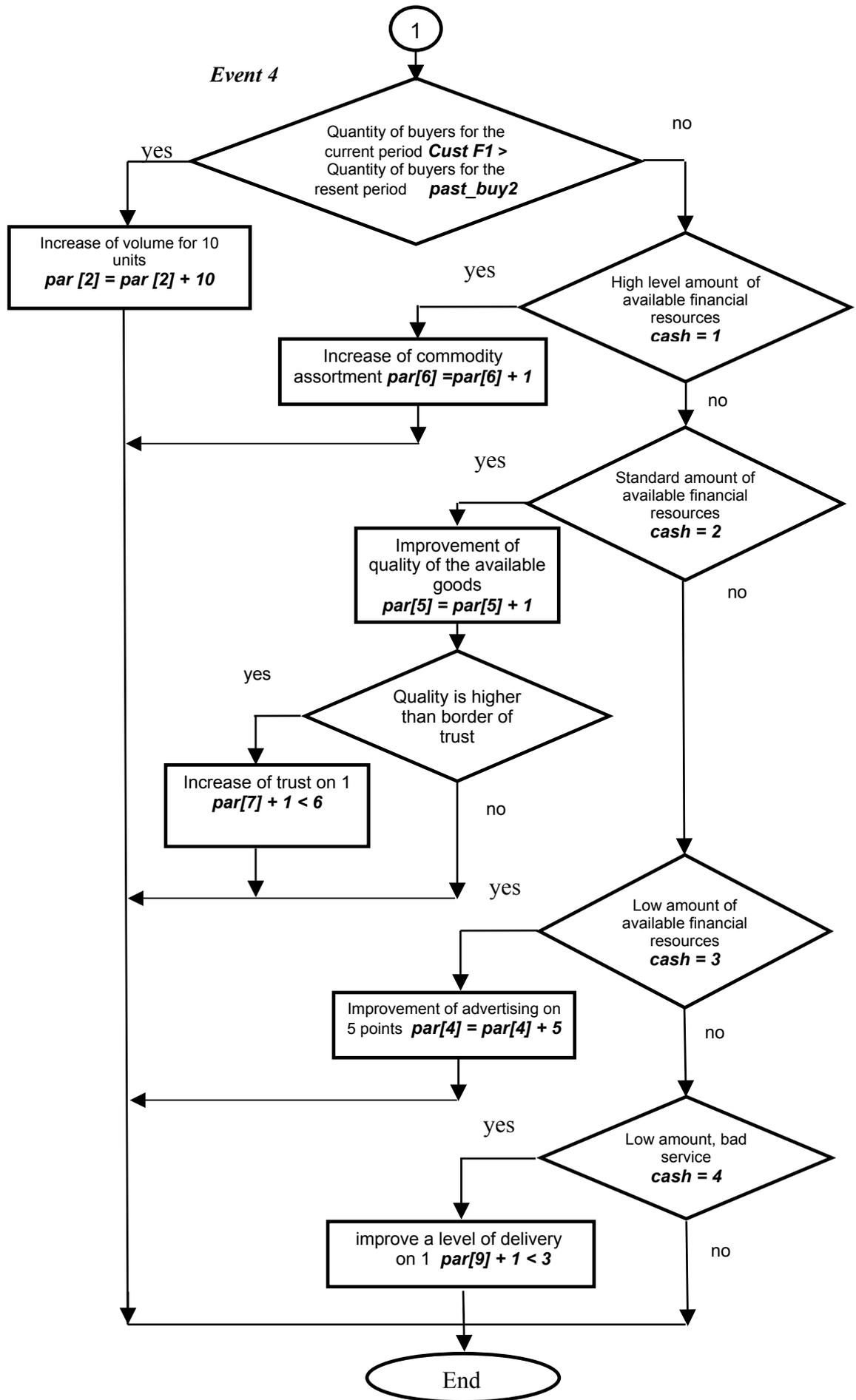
**Event 1.** Each seller traces behaviour of competitors, determining for itself an average price of the goods *average\_price* in a preceding period. Depending on, whether there will be a current price *par [1]* more or

less than average, its reduction or increase at 1 % follows.

In **event 2** each firm at presence of not catered clients increases the scope of supply *par [2]* by the missing size *dem [i]*.



**Figure 2: Algorithm of a choice of strategy in dependence on current prices of competitors and own demand (the beginning)**



**Figure 2: Algorithm of a choice of strategy in dependence on current prices of competitors and own demand (the end)**

**Event 3.** If the stock balance of firm *balances [i]* is more than 1/3 of all let out volumes *par [2]*, oligopolist enters discounts for sale and reduces the let out volume by determined percent (for example, on 2 %).

Further, proceeding from fluctuations of demand, oligopolist varies not price factors (**event 4**). If in the current period the quantity of buyers *Cust F* increases in comparison with the last period *past\_buy2* the firm increases volume of tenders by 10 units. If demand is reduced, the firm undertakes actions according to quantity of available free money resources (variable *cash*). In the accepted strategy at the sufficient sum of receipts, *cash = 1*, oligopolist expands the commodity range, putting into circulation the new goods. The further fluctuations of a variable *cash*, correspond to reducing of money resources and amplification of influence of not price factors. So at *cash = 2* oligopolist aspires to raise quality of the available goods, improving thus its image and increasing trust of clients. At *cash=3* advertising improves, and at *cash = 4* - conditions of delivery, etc

### MULTI-AGENT SIMULATION MODEL OF MARKETING SYSTEM

Multi-agent simulation model is made of the *basis MDL-components* describing a condition status and dynamics of behaviour of elements of system, *the organizational components* specifying structure of interrelations between basis components, and *mobile components* - for the description of the reports placed in *accumulative arrays* and forming turns on servicing. *The MDL-description* of structure of model organizational component HIGH LEVEL COMPONENT according to the scheme of connections of basis components (fig. 1) has the following appearance:

HIGH LEVEL COMPONENT *Market\_HIGH*

#### SUBCOMPONENTS

Firm1, Firm2, Firm3, buyer, Market

#### COMPONENT CONNECTIONS

Firm 1.*QueryStock1*--> Market.*Qstock 1*;

Firm 2.*QueryStock2*--> Market.*Qstock 2*;

Firm 3.*QueryStock3*--> Market.*Qstock 3*;

Buyer. criterion {i OF 1..9} {j OF 1..100} -->

Market. Criterion [i][j] ;

Firm1.parameters1 {i OF 1..9}-->Firm3.parameters1 [i]

Firm2.parameters2 {i OF 1..9}-->Firm3.parameters2 [i]

Market. g --> Firm1.g , Firm2.g , Firm3. g

# Quantity of buyers

Market. *Cust F1* --> Firm1.*Cust F1*;

Market. *Cust F2* --> Firm2. *Cust F2*;

Market. *Cust F3* --> Firm3. *Cust F3*;

# Non-realized remnants

Market. *balances[1]* --> Firm1.*balances[1]*;

Market. *balances [2]* --> Firm2. *balances[2]*;

Market. *balances [3]* --> Firm3. *balances[3]*;

.....  
Market. *EQUIPOISE* --> Firm3. *EQUIPOISE*;  
END OF *Market\_HIGH*

Two self-contained components are consolidated via communication line on which the behaviour of a variable from one component is taken into account in the other. Thus in section COMPONENT CONNECTIONS the name of a component and his imported sensor variable directed on a sensor input of other component is underlined.

*The behaviour of the agent-buyer* is described by basis component *Buyer* which should choose the seller and establish a volume of orders by criteria of seller's option and a product. The actions of agents-sellers described in basis components *Firm1*, *Firm2*, *Firm3*, are connected to a choice of price and not price strategy of each firm - oligopolist with allowance for strategy and variables of a condition status of other oligopolists-competitors transmitted on sensor communications.

Work of model during the set interval of time *TNext* consists of lines of consecutive conditional events (fig. 2). Transition from one event to another is managed with special signals-indicators. After the end of simulation, it is signaled for making up of the summary information on past transactive to the period then the new reference point of time *Tnext* for new transactive the period is established.

### SIMULATION RESULTS

Results of simulation of price strategy of two oligopolists - competitors with the subsequent occurrence of the third in universal simulation system *Simplex3* are submitted on fig. 3 - 7. The chart on fig. 3 shows process of elastic pricing at the big distinction of the initial prices and stabilization of the market at occurrence in him the new competitor with the price strategy. On fig. 4 the competition is shown in fluctuation by a market share (number of buyers) of each firm and possible allocating of shares after occurrence in the market new oligopolist is shown. The Fig. 5 - 7 show dynamics of a level of demand on production of each firm as chart of fluctuation of volumes of tenders, sales and remnants for the sale, developing in modeled processes of competition and dynamic interaction of intellectual agents

### CONCLUSIONS

*Multi-agent simulation of marketing system* opens and explains the mechanism of strategy of pricing of the competing parties in disputed situations of counteraction of active elements in the marketing environment. Decision-making is directed on studying and forecasting of processes of stabilization of the market at various economic and social indignations, to a choice of optimum marketing strategies, balancing of supplies and demands in the current conditions. Therefore development of multi-agent simulation

models reflecting variety of behaviour and interaction of self-contained links of marketing system is socially significant and actual for an estimation of complex

situations and computer support of accepting of critical decisions.

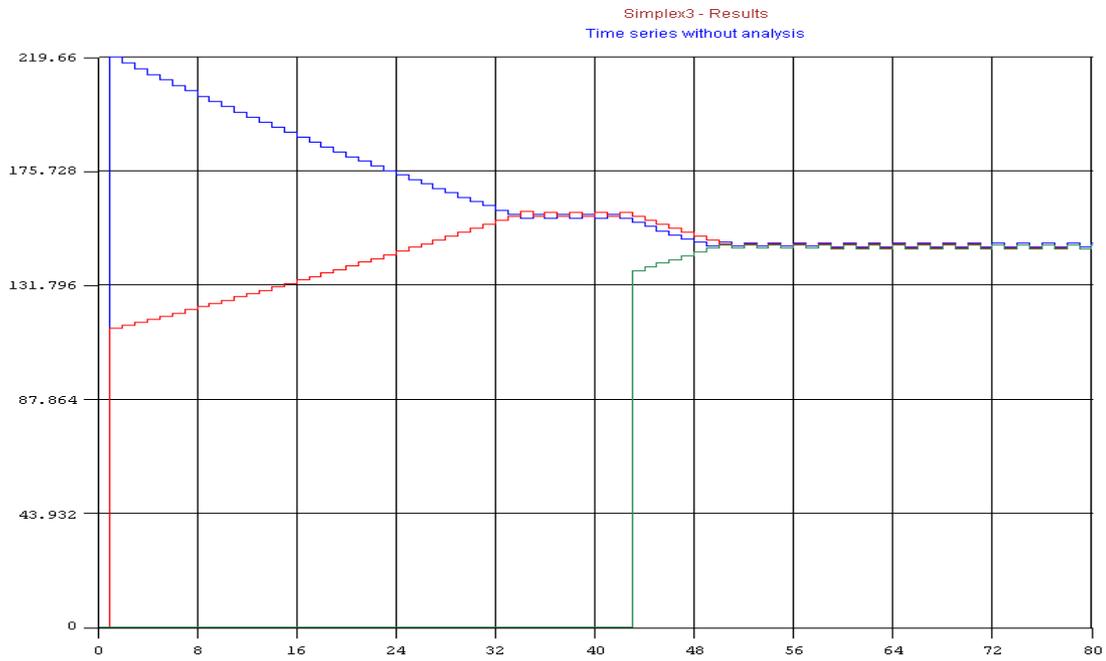


Figure 3: Process of stabilization of the market at strategy of elastic pricing

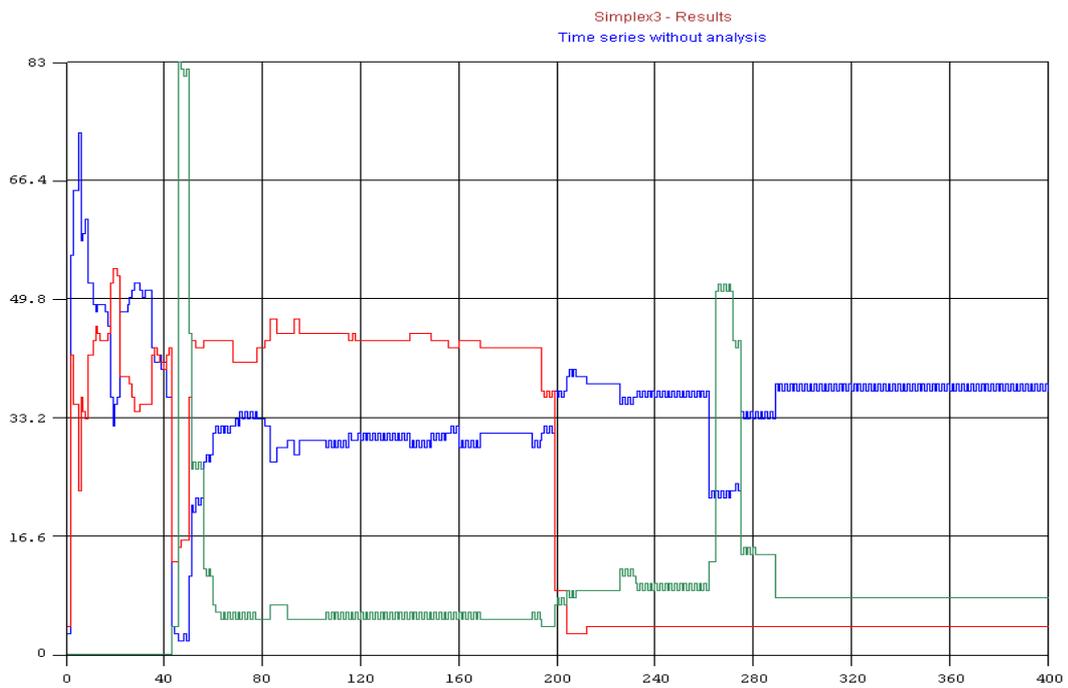
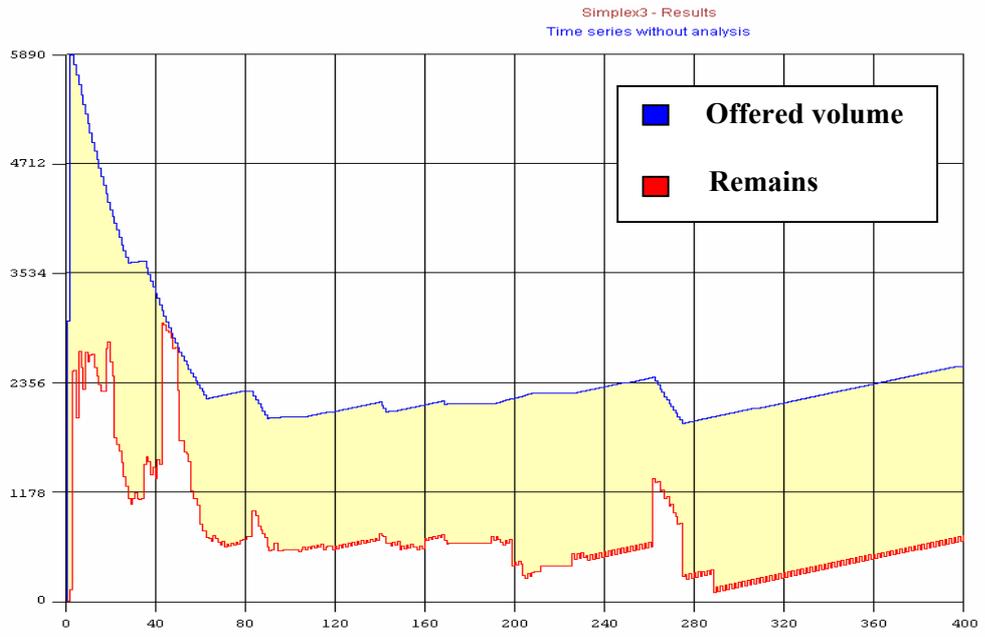
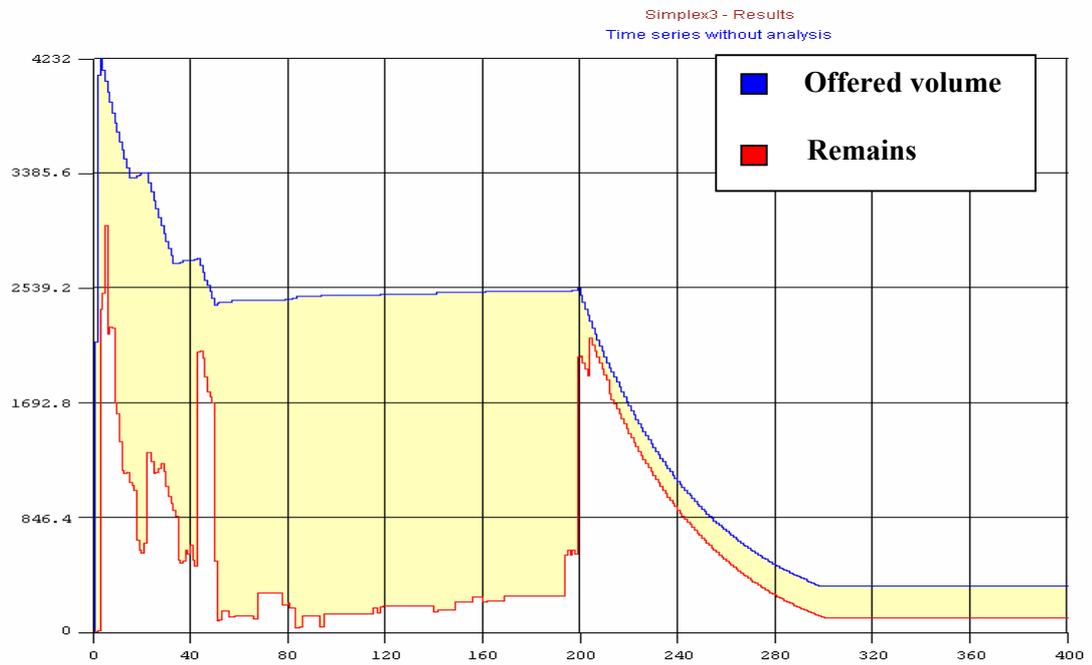


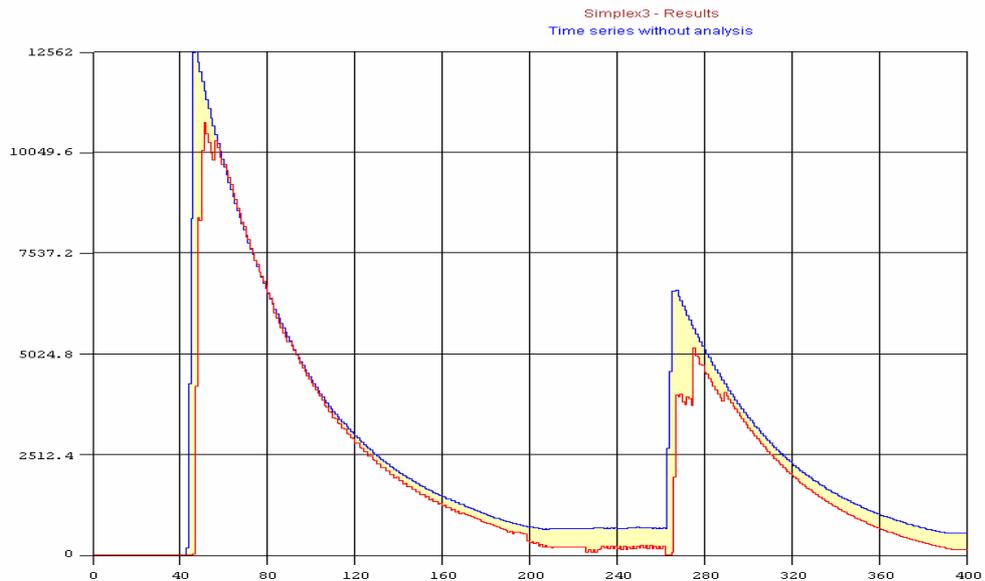
Figure 4: Quantity of buyers of each firm



**Figure 5: An offered volume and the remains of the goods of first firm**



**Figure 6: An offered volume and the remains of the goods of second firm**



**Figure 7: An offered volume and the remains of the goods of third firm**

## REFERENCES

- B. Schmidt "The Art of Modelling and Simulation. Introduction to the Simulation System Simplex 3", Ghent, Belgium, 2001.
- B. Schmidt. Simplex2. Modellbeschreibungs-sprache SIMPLEX-MDL - Universität Passau, 1996.
- Gorni Ch. Ein Referenzmodell für die Simulation intelligenter Agenten. Simulation in Passau. Heft 2, 1996, Universität Passau, Germany.
- Schmidt B. Die Modellierung menschlichen Verhaltens. SCS-Europe BVBA, Ghent, Belgium, 2000.- 105 s.
- Yu. A. Ivashkin Structural-parametrical modeling of intellectual agents and system. 2<sup>nd</sup> Workshop on agent-based simulation. Passau, Germany 2001, p. 71-76.
- Weppner H.: Individuenbasierte Simulation eines oligopolistischen Marktes auf Basis des Referenzmodells PECS; Lehrstuhl für Operations Research und Systemtheorie, Universität Passau 1998.
- Shcherbakov Anton V.  
Scientific collaborator, aspirant of the chairs "Computer Technologies and Systems" of the Moscow state university of applied biotechnology; 1969 of birth. Scientific interests: Operations Research, agent-based technology, marketing, modeling of the marketing System and price strategy.
- Rogozhkina Elena A.  
Scientific collaborator of the chairs "Computer Technologies and Systems" of the Moscow state university of applied biotechnology; 1984 of birth. Scientific interests: the system analysis and systems of Simulation and Control.

## AUTHORS BIOGRAPHY

Ivashkin Yuri A.

Doctor of engineering science, professor, head of the chairs "Computer Technologies and Systems" of the Moscow state university of applied biotechnology; The valid member of the International academy of information and International academy of system researches; 1940 of birth. Scientific interests: the system analysis and theory of making decisions; modeling of the intellectual agents and multi-agent systems.

# A MULTI-AGENT SIMULATOR FOR TESTING AGENT MARKET STRATEGIES

Maria João Viamonte, Carlos Ramos,  
Fátima Rodrigues  
GECAD – Knowledge Engineering and  
Decision Support Group  
Institute of Engineering, Polytechnic of  
Porto, Portugal  
E-mail: {viamonte,csr,fr}@dei.isep.ipp.pt

José Carlos Cardoso  
Department of Electrical  
Engineering, University of Trás-os-  
Montes e Alto Douro, Vila Real,  
Portugal  
E-mail: jcardoso@utad.pt

## KEYWORDS

Agent-based simulation, Electronic markets, Dynamic agent strategic behaviour, Data mining.

## ABSTRACT

We envision a future in which the global economy and the Internet will host a large number of interacting software agents. Most of them will be economically motivated, and will negotiate a variety of goods and services. It is therefore important to consider the economic incentives and behaviours of economic software agents, and to use all available means to anticipate their collective interactions. This paper addresses this concern by presenting a multi-agent market simulator designed for analysing market strategies based on a complete understanding of buyer and seller behaviours, preference models and pricing algorithms. The results of the negotiations between agents will be analysed by Data mining tools in order to extract rules that will give the agents feedback to improve their strategies.

## INTRODUCTION

As the result of technological developments electronic commerce is emerging as the new way of doing business. We believe that, over the course of the next decade, the global economy and the Internet will merge into a global market with a large amount of autonomous software agents that exchange goods and services with humans and other agents. Agents will represent or support consumers, producers, and intermediaries. When interactions among agents become sufficiently rich, a crucial qualitative change will occur. New classes of agents will be designed specially to serve the needs of the other agents. The agents we are envisaging will not be just assistants to the business process. They will add value to their activities by, synthesising, filtering, translating, and mining. However, it would be dangerous to assume that theories and intuitions based on centuries of human experience in business processes will be directly applicable to understand, anticipate, and control the behaviour of markets in which software agents participate.

To study electronic markets behaviour and evolution, we developed ISEM (Viamonte et al., 2004) (Viamonte et al., 2003), a multi-agent market simulator, designed for analysing agent market strategies. This simulator has been selected to be included in a book about the application of agents in electronic commerce in Europe (Viamonte and Ramos, 2001) and was recently selected as a worldwide case study in simulation of negotiation agents (Viamonte et al., 2006). ISEM ideas are also currently being applied under the scope of the project Agent&Markets (POSI/EIA/56260/2004) supported by the Portuguese Agency for Scientific Research (FCT). The main objectives of ISEM are: first, the ISEM system addresses the complexity of on-line buyer's behaviour by providing a rich set of behaviour parameters; second, the ISEM system provides available market information allowing sellers to make assumptions about buyer's behaviour and preference models; third, the different agents customise their behaviour adaptively, by learning user's preference models and business strategies. The agent learning ability is achieved through data mining techniques applied on-line during the market sessions of ISEM simulator.

## ISEM CONCEPT

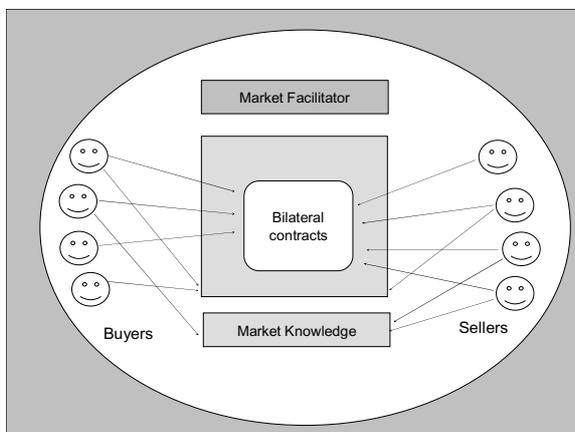
The underlying structure of ISEM is that a simulation-based approach can model more diverse and complex scenarios, rather than the general case. By using a simulator prior to conducting marketing experiments, suppliers and consumers can develop an intuitive understanding of the theoretical findings and use this knowledge to develop a more sophisticated strategy implementation. Our investigation of agent-mediated electronic commerce focuses specially on agent market strategies for an extremely common type of market: a market with finite time horizon, seller inventory, and buyer population, such as airline tickets, hotel rooms and seasonal retail. Also inherent to the finite nature of these markets is an increased importance of fluctuations in consumer demand. In order to take advantage of these demand changes we are interested in investigate dynamic agent market strategies. Our strategy algorithms make assumptions about the behaviour of

the buyers or the type of buyers, based on available market information that is obtained with data mining tools. Our approach opens some interesting research directions to study user modelling with knowledge discovery tools.

As decision support tool, we developed a market simulator that creates real-life bargain situations and is based on the model proposed by Fatima et al. (Fatima et al., 2004). Seller and buyer agents negotiate over the price of a good or service in order to established bilateral contracts and in addition to attempting to obtain the best price, agents need to ensure that negotiations ends before a certain deadline; moreover, agents make assumptions about buyers behaviour and preference models based on available market information. The simulator probes the conditions and the effects of market rules, by simulating the participant's strategic behaviour; moreover, agents can adapt their strategies as the simulation proceeds on the basis of previous efforts successes or failures. ISEM is flexible since the user completely defines the model he or she wants to simulate, including the number of agents, each agent's type and strategies.

### ISEM MARKETPLACE MODEL

ISEM works like an open market where buyer and seller agents meet in the marketplace. It includes these types of agents: a market facilitator, sellers, buyers and market knowledge, figure 1.



Figures 1: Multiple Agents in ISEM

The market facilitator agent coordinates the simulated market and ensures that it functions correctly. It knows the identities of all the agents in the market, regulates negotiation, and assures that the market operates according to established rules. Before entering the market, agents must first register with the market facilitator, specifying their role and services.

Seller and buyer agents are the two key players in the market, so we devote special attention to them, particularly to their business objectives and strategies to reach

them. In order to be competitive in today's economic markets, buyer and seller agents need not only to be efficient in their business field, but also to be able to quickly react and adapt to new environments as well as to interact with other available entities. The control architecture adopted for the design of those agents should meet these requirements, having a similar structure but with a kind of symmetrical behaviour (due to their antagonistic business objectives). The structure comprises four functional modules: communication; individual knowledge; decision making & coordination; and execution.

The user completely defines the number of seller and buyer agents in each scenario and must specify their intrinsic and strategic characteristics. "Intrinsic characteristics" refer to the agents' individual knowledge related to product list, limit prices, preferred prices, profile and available capacity or consumption needs. "Strategic characteristics" refer to the strategies the agent will use to reach the objective of selling the available inventory at the best price (seller) or buying the needed items with less costs (buyer). Sellers will compete with each other because they are all interested in selling their inventory at the highest possible values. On the other hand, sellers will cooperate with buyers to establish an agreement that is profitable for both. This is a rich domain for which it is possible to develop and test several decision algorithms and strategies for cooperation and competition.

The market knowledge agent is a special agent included in the ISEM system, which plays the role of "power" agent. This agent has access to market knowledge, which contains information about the organisational and operational rules of the market, as well as information about all different running agents, their capabilities and historical information. The market previsions and agent behaviour models are obtained through data mining algorithms, using data resulting from agent negotiations that support agents' market strategies. In practice, usually, after a confidential negotiation period, the market facilitator agent discloses information about past transactions and agents' characteristics (if possible); all agent interactions are logged at a transaction level of detail, which provide a rich source of business insight that can help to customise the business offerings to the needs of the individual buyers. With this functionality it is possible to discover sub-groups that behave independently and associations between products. For that, ISEM uses clustering, classification and association operations.

To carry out the clustering operation a Two-Step clustering algorithm (Zhang et al., 1996) is used to target buyers with similar characteristics in the same agent group. Then, to obtain more relevant information that describes the consumption patterns of each cluster population, a rule-based modelling technique, using

C5.0 classification algorithm, an evolution of C4.5 algorithm (Quinlan, 1993), is used to analyse those clusters and to obtain descriptions based on a set of attributes, collected in the individual agents' knowledge module. These models are transferred to the market knowledge agent and offer a set of market information, such as: preferred sellers; preferred marks; favourite products and reference prices, which support the process of agents' strategy implementation. To discover associations between buyer details and purchases, data from multiple agent negotiations are manipulated to create "basket" records showing product purchases. This permits the observation of the behaviour of each buyer agent. This data is combined and manipulated by the "Apriori algorithm" (Agrawal et al., 1996), to discover associations between buyer details and purchases. The best association rules, those with a strong support and confidence, are extracted and transferred to the market knowledge agent. With this kind of knowledge it is possible to provide insight into the sellers' agents about the profiles of buyer agents with certain purchase pro-pensities, showing associations between products, prices, style, etc.

After these operations, to get confident data, agents can request the services provided by the market knowledge agent, in order to support their strategic behaviour. Only players with more sophisticated behaviour will take advantage of this new knowledge; since the user can determine which seller agents have access to this facility. The user can also determine if the agents' information will be private or public; public information is available to market analysis with the data mining functionality. However the market can get knowledge about an agents' behaviour even if they are set as a private information agent. This situation occurs, by the simple fact of being on the market.

The ISEM facilitates agent meeting and matching, besides supporting the negotiation model. In order to have results and feedback to improve the negotiation models and consequently the behaviour of user agents, ISEM simulates a series of negotiation periods,  $D = \{1, 2, \dots, n\}$  where each negotiation period is composed by a fixed interval of time  $T = \{0, 1, \dots, m\}$ .

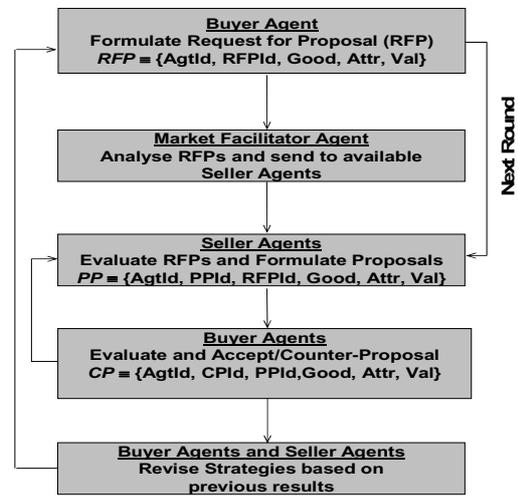
Moreover, each agent has a deadline  $D_{\max}^{Agt} \in D$  to achieve its business objectives. At a particular negotiation period, each agent has an objective that specifies its intention to buy or sell a particular good or service and on what conditions.

### Negotiation Model

The negotiation model used in ISEM is bilateral contracting where buyer agents are looking for sellers that can provide them the desired products at the best price. We adopt what is basically an alternating protocol (Fatima et al., 2004) (Gallego and Ryzin, 1994). Let  $Agtb$  denote the buyer agent,  $Agts$  the seller agent and

let  $[P_{i_{\min}}, P_{i_{\max}}]$  denote the range of values for price that are acceptable for agents. A seller agent has the range  $[P_{si_{\min}}, P_{si_{\max}}]$ , which denotes the scale of values that are comprises by the minimum value that the seller is disposed to sell to the optimal value. A buyer agent has the range  $[P_{bi_{\min}}, P_{bi_{\max}}]$ , which denotes the scale of values that are comprises by the optimal value to buy to the maximum value.

Negotiation starts when a buyer agent sends a request for proposal (RFP), figure 2. In response, a seller agent analyses its own capabilities, current availability, and past experiences and formulates a proposal (PP). Sellers can formulate two kinds of proposals: a proposal for the product requested or a proposal for a related product, according to the buyer preference model (see section Seller Behaviour for details).



Figures 2: Sequence of Bilateral Contracts

$PP_{g_i}^{DT, Agts \rightarrow Agtb}$  represents the proposal offered by the seller agent  $Agts$  to the buyer agent  $Agtb$  at time  $T$ , at the negotiation period  $D$  for the  $good_i$ . The buyer agent evaluates the proposals received with an algorithm that calculates the utility for each one,  $U_{PP_{g_i}^{DT, Agts \rightarrow Agtb}}^{Agtb}$  (see section Buyer Behaviour for details); if the value of  $U_{PP_{g_i}^{DT, Agts \rightarrow Agtb}}^{Agtb}$  for  $PP_{g_i}^{DT, Agts \rightarrow Agtb}$  at time  $T$  is greater than the value of the counter-proposal (CP) that buyer agent will formulate for the next time  $T$ , in the same negotiation period  $D$ , then the buyer agent accepts the offer and negotiation ends successfully in an agreement; otherwise a counter-proposal  $CP_{g_i}^{DT, Agtb \rightarrow Agts}$  is made by the buyer agent to the next time  $T$ . The seller agent will accept a buyer counter-proposal if the value of  $U_{CP_{g_i}^{DT, Agtb \rightarrow Agts}}^{Agts}$  is greater than the value of the counter-proposal (CP) that seller agent will formulate for the next time  $T$ ; otherwise seller agent

rejects. On the basis of the bilateral agreements made among market players and lessons learned from previous bid rounds, both agents revise their strategies for the next negotiation rounds and update their individual knowledge module.

### **AGENTS STRATEGIC BEHAVIOUR MODEL**

Agents use time-dependent strategies to change their price during a negotiation period: offers and counter-offers are generated by linear combinations of simple functions, for simple criteria, the time; at this work, we have also used the time-dependent strategies to model different attitudes towards time, during a negotiation period; an agent that gains utility, with the time, and has the incentive to reach a late agreement (within the remaining time until the end of a negotiation period) is considered a strong or patient player; an agent that loses utility with time and that tries to reach an early agreement is considered a weak or impatient player. Agents use behaviour-dependent strategies to adjust parameters (price, demand) for the next negotiation period according to the results obtained in the previous ones.

Buyers and seller agents develop their behaviour and strategies based on a combination of public information, available through requesting for market knowledge agent services; and private information, available only to the specific agent at their individual knowledge module. It is expected that each agent develops the individual knowledge module with historical information, since they have different behaviours and consequently different results. On the basis of results from ISEM simulations, the agents can build a profile of the other agents with expected proposed prices, limit prices, needs and capabilities. On the other hand, requests for market knowledge agent services also provide a great support for agents that have more sophisticated behaviour.

#### **Buyer Behaviour**

Over the course of the market, the collective behaviour of buyer agents is defined by three variables: the lifetime, the maximum price, and its strategy; the lifetime parameter indicates how many days they are disposed to wait in the market, continuously looking for the best deal. Indirectly, the lifetime of buyers determines the number of buyers in the market each day. Pre-existing buyers return if they were unable to purchase in the previous days and their specified lifetime has not expired. Each buyer has a set of products that it wants to buy, and for each one it has information about attributes and products alternatives, if any. Buyers will analyse the seller's proposals with an advanced algorithm which analyses the different proposals, evaluates the expected returns and then apply a decision method (decision buyer algorithm) to decide when accept a bilateral contract. The advanced

algorithm, first sorts the proposals for the requested product by price and selects the best one, which will be compared to its own values. If it finds a seller proposal satisfactory then the buyer will contact directly the seller in question; otherwise if the buyer has a preferred "seller" then it can increase the reserve price (e.g. plus 10%); finally it will analyse the proposals for related products, if it finds proposals for alternative products accordingly to the user preference model, then it will start a similar analysis. Buyer agents can choose from four different time-dependent tactics: Determined, Anxious, Moderate and Gluttonous (Viamonte et al., 2003): these strategies depending on both the point in time when the agent starts to modify the price and the amount it changes; and can use two complementary behaviour-dependent tactics: the Modified Goal Directed for Buyers (MGDB) and the Fragmented Demand (FD). The MGDB strategy (Viamonte et al., 2003) is based on two consecutive objectives; the first one is buying the consumption needs and then reducing payoff. Following this strategy, buyers will offer a higher price if they didn't meet their consumption needs in the previous period and offer less if they succeeded in meeting their needs. The FD strategy (Viamonte et al., 2003), adjusts the demand per day by attempting to reach the goal of buying its entire needs by the last day of the market, and not before, this strategy paces its purchases over the market, with the goal of buying all the units needed but with less costs. This strategy allow buyers to save money, however, some times buyers are not capable of buying all the needed units, because while waiting to buy till the last day of market they lose the chance of buying.

#### **Seller Behaviour**

The user defines the behaviour of the sellers in the market, both in terms of their behaviour over time (business objectives and agent risk characterisation), and their behaviour on a per day basis (negotiation strategies). Every day each seller has a set of products that it wants to sell. The seller will analyse the request for proposals received and formulates a proposal with an advanced algorithm. Two kinds of proposals are possible: a proposal for the requested product, if the seller has the requested product or a proposal for a related product. It is expected that seller agents be proactive, by asking for the services provided by the market knowledge agent to suggest a feasible alternative product. A seller formulates an alternative proposal supported by an overall utility function, which reflect the business objectives of the user that it represents and agent risk characterization (Viamonte et al., 2003), which will determine how the sellers will behave. To define the next period parameters according to the results obtained in the previous ones sellers choose between two different behaviour-dependent strategies: the Modified Goal Directed for Sellers (MGDS) (Viamonte et al., 2003), that adjusts its price by attempting to reach the goal of selling the entire

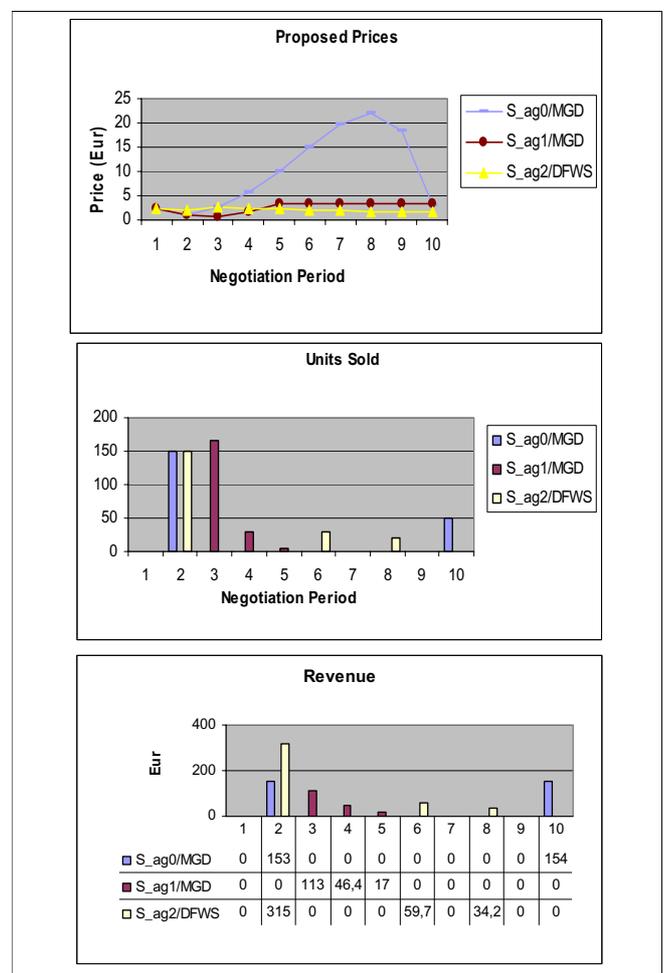
inventory by the last day of the market, by lowering prices when sales in the previous day are low and raising prices when the sales are high; and the Derivative Following (DF) strategy (Viamonte et al., 2003) that can be weighted by Seller Satisfaction (DFWS) or by the Previewed Demand for a specific product (DFWPD), this strategy adjusts its price by looking to the amount of revenue earned on the previous day as a result of the previous day price change. If in the previous day, the price change produced more revenue, then the strategy makes a similar change in price, otherwise the strategy makes an opposite price change. This strategy calculation is an adjustment of the Derivative Following strategy analysed in (Greenwald et al., 1999). We modified the DF strategy for a finite market by attempting to reach the goal of selling its entire inventory by the last day of the market, instead of adjusting the price each day, the change is scaled by a ratio based on the progress through the market and can be based on the percentage of Buyers that we expect to satisfy (% Satisf) or based in the value for Previewed Demand. Seller agents can obtain these values through requesting for market knowledge agent support; and permits to do changes that will be done accordingly to buyer loyalty and to demand expected for a given product.

### DYNAMIC STRATEGIES ANALYSE

We use the above referred strategies, which are already implemented, to present the following example, which illustrates some differences in how behaviour-dependent strategies performed. Consider a simple scenario with few sellers and few buyers using time-dependent and behaviour-dependent strategies. In every trial we present, the market has 10 days, each seller has 200 units and each buyer wants 150 units of the same good (ex: mobile phone). We test these strategies under a comparison-shopping and with preferences for certain sellers over others. All sellers start with the same price and each buyer are able to pay different prices. Moreover, all of the agents have the last day of functioning of market as deadline to do their transactions. We pretend to analyse which behaviour-dependent strategy is appropriate under these specific conditions. In a competitive market, the adaptive pricing strategies react to the others strategies in the market in addition to the buyers demand.

As we can see, figure 3, all the sellers achieve their goal, to sell almost everything. After carefully analysing the results, we can observe that the DFWS strategy produces a high amount of revenue and often sells more units than the other seller agents using the MGDS. The success of a DFWS depends on the starting price it chooses, and the percentage of buyers satisfied. When DFWS sells approximately the same amount of inventory as MGDS, it usually produces more revenue than the MGDS strategy, and frequently occurs that, even when DFWS sells a smaller amount of inventory,

it usually produces more revenue than MGDS, since this one makes a dramatic price change; this occurs because the MGDS strategy spreads out its sales, including selling on the last days when prices approach the minimum. Another important issue is that MGDS does not take into account the percentage of buyers that are satisfied when making price changes. Furthermore, we can conclude that when the demand is less than the most competitive sellers' available capacity, the seller will lose money when using the MGDS; the seller will decrease the price and try to sell more, which may not be possible because of insufficient demand. However, MGDS strategy can be valuable, particularly to increase market share when two or more sellers are competing directly because of similar proposed prices. Buyers using the FD strategy frequently buy the requested units, with fewer costs. Although these strategies are computationally straightforward, they are surprisingly robust under extremely different market conditions.



Figures 3: The Modified Goal Directed for Sellers and Derivative Following Weighted by Satisfaction Strategies

### CONCLUSIONS

ISEM seems to be a valuable framework for studying market evolution. The multi-agent technology allied to

an objected-oriented implementation enables easy future improvements and model enlargement. Market participant's strategic behaviour is very significant in the context of competition. In addition, the availability of new market knowledge obtained with data mining algorithms is vital for supporting marketing and sales. ISEM works as a platform for evaluation, enriched with the ability to segment the buyer population into different sub-groups that behave independently. Another important particularity of ISEM simulator is the inclusion of a buyer behaviour-dependent strategy, able to adapt based on observed market changes. Although we implemented some valuable and promising strategies, we must increase and improve the portfolio of agents' strategies and behaviours. Directions of our future work include evaluating additional dynamic market strategies; based on different value-added services, for sellers and more sophisticated behaviour-dependent strategies for buyers.

## REFERENCES

- Viamonte M. J., C. Ramos, F. Rodrigues, and J. C. Cardoso. "Learning User Preferences Models and Business Strategies for E-Commerce" in *eAdoption and the Knowledge Economy*, Paul Cunningham, Miriam Cunningham editors, IOS Press, ISBN 1586034707, pp. 1123-1130, 2004.
- Viamonte M. J., C. Ramos, F. Rodrigues, and J. C. Cardoso. "A Simulation-Based Approach for Testing Market Strategies in Electronic MarketPlaces" in *Proceedings of the International Conference on Web Intelligence (WIC)*, Halifax, Canada, pp. 490-494, 2003.
- Viamonte M. J., C. Ramos. "A Model for an Electronic MarketPlace" in *Agent Mediated Electronic Commerce, European AgentLink Perspective, Lecture Notes in Artificial Intelligence 1991*, Frank Dignum and Carlos Sierra, editors Springer, ISBN 3-540-41671-4, pp. 115-125, 2001.
- Viamonte M. J., C. Ramos, F. Rodrigues, and J. C. Cardoso. "ISEM: A Multi-Agent Simulator For Testing Agent Market Strategies" to be published in the *IEEE Transaction on Systems, Man and Cybernetics, Part C, Special Issue on Game-theoretic Analysis and Stochastic Simulation of Negotiation Agents*, 2006.
- Fatima S., M. Wooldridge, N. Jennings. "An agenda-based framework for multi-issue negotiation" in *Artificial Intelligence Journal* vol. 152, nº1, pp.1-45, 2004.
- Zhang T., R. Ramakrishnon, and M. Livny. "BIRCH: An Efficient Data Clustering Method for Very Large Databases" in *Proceedings of the ACM SIGMOD Conference on Management of Data*, p. 103-114, Montreal, Canada, 1996.
- Quinlan J. "C4.5: Programs for Machine Learning" Morgan Kaufmann Publishers, San Francisco, USA, 1993.
- Agrawal R., H. Manilla, R. Srikantand, H. Toivonen, I. Verkamo "Fast discovery of association rules". AAAI Press 1996.
- Gallego G. and G. V. Ryzin, "Optimal Dynamic Pricing of Inventories with Atochastic Demand Over Finite Horizons" in *Management Science* Vol. 40 No.8, pp. 999-1020 1994.
- Greenwald A., J.Kephart and G. Tesauro, "Strategic Pricebots Dynamics" in *Proceedings of the First ACM Conference on Electronic Commerce*, Denver, USA, pp. 58-67, 1999.

## AUTHOR BIOGRAPHIES

**MARIA JOÃO VIAMONTE** is an assistant professor of computer engineering at the Polytechnic Institute of Porto's Institute of Engineering. Her research interests are in multiagent simulation, agent mediated electronic commerce and decision support systems. She received her PhD in electrical engineering from the University of Trás-os-Montes and Alto Douro. Contact her at Instituto Superior de Engenharia do Porto, Departamento de Engenharia Informática, Porto, Portugal; [viamonte@dei.isep.ipp.pt](mailto:viamonte@dei.isep.ipp.pt)

**CARLOS RAMOS** is a coordinator professor of computer engineering and director of the knowledge Engineering and Decision Support Research Group at the Institute of Engineering of Porto's. His research interests are artificial intelligence and decision support systems. He received his PhD in electrical engineering from the University of Porto. Contact him at Instituto Superior de Engenharia do Porto, Departamento de Engenharia Informática, Porto, Portugal; [csr@dei.isep.ipp.pt](mailto:csr@dei.isep.ipp.pt)

**FÁTIMA RODRIGUES** is a coordinator professor of computer engineering at the Polytechnic Institute of Porto's Institute of Engineering. Her research interests are in multiagent simulation, agent mediated electronic commerce and decision support systems. She received her PhD in electrical engineering from the University of Minho. Contact her at Instituto Superior de Engenharia do Porto, Departamento de Engenharia Informática, Porto, Portugal; [fr@dei.isep.ipp.pt](mailto:fr@dei.isep.ipp.pt)

**JOSÉ C. CARDOSO** is an Associated Professor of engineering department at University of Trás-os-Montes e Alto Douro, Portugal. His research interests include telecommunications, networks and digital signal processing. He received his PhD in electrical and electronics engineering from the University of Wales, Bangor, U.K. Contact him at Departamento de Engenharia, Universidade de Trás-os-Montes e Alto Douro, Vila Real, Portugal. [jcardoso@utad.pt](mailto:jcardoso@utad.pt)

# TESTING THE SCENARIO ANALYSIS ALGORITHM OF AN AGENT-BASED SIMULATOR FOR COMPETITIVE ELECTRICITY MARKETS

Isabel Praça, Carlos Ramos and Zita Vale  
GECAD: Knowledge Engineering and  
Decision Support Group  
Institute of Engineering - Polytechnic of Porto  
R. António Bernardino de Almeida, 4200-072 Porto  
E-mail: isp@dei.isep.ipp.pt

Manuel Cordeiro  
Secção de Engenharias  
University of Trás-os-Montes e Alto Douro  
Vila Real

## KEYWORDS

Agent-based Simulation, Electricity Markets, Dynamic Behaviour.

## ABSTRACT

The main goal of MASCEM – Multi-Agent Simulator of Competitive Electricity Markets, is to give electricity market entities a tool to support their decisions and to obtain knowledge about market behaviour and evolution. Intelligent agents with strategic behaviour represent entities from electricity markets. In this paper we propose some illustrative scenarios to explain and take some conclusions about MASCEM agent's Scenario Analysis Algorithm behaviour.

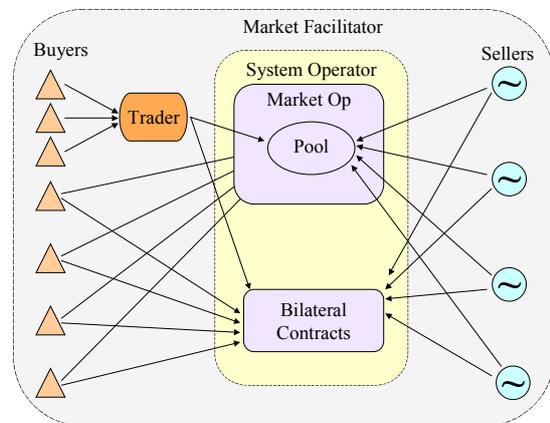
## INTRODUCTION

The electricity industry is becoming competitive; a market environment is replacing the traditional centralized-operation approach. New entities have emerged and the role of existing ones has changed. There are different market rules, based on different negotiation mechanisms. In this context, new modelling approaches that simulate how electric power markets might evolve over time and how market participants might react to the changing environment are welcome. Electricity market entities are heterogeneous and autonomous, have their own objectives and follow their business strategies in order to reach them; they interact among them in a dynamically changing environment. All these characteristics led us to develop MASCEM – Multi-Agent Simulator of Competitive Electricity Markets, and give electricity market entities a tool to support their decisions and to obtain knowledge about market behaviour and evolution. With MASCEM several experiences have already been made, leading us to achieve some conclusions and define future developments.

## MASCEM: MULTI-AGENT MODEL

There are different types of agents in our model: Market Facilitator Agent, Seller Agents, Buyer Agents, Trader Agents, Market Operator Agent and System Operator Agent. In this section we will briefly describe their

roles, functionalities and the interactions between them, for details please consult (Praça et. al 2003). Figure 1 illustrates MASCEM Multi-Agent model.



Figures 1: Multi-Agent Model

The Market Facilitator plays the role of market coordinator of the Electricity Market. It knows the identities of all the agents present in the market, regulates the negotiation process and assures the market is functioning according to the established rules. Agents before entering the market must first carry out the registration with the Market Facilitator, specifying their market role and services.

Seller and Buyer agents are the two key players in the market, so a special attention is devoted to them, and particularly to their objectives and strategies they can use to reach them. Seller agents represent entities able to sell electricity in the market, e.g. generating companies holding electricity production units. Buyer agents represent electricity consumers and electricity distribution companies. The number of Seller and Buyer Agents in each scenario is completely defined by the user, who must also specify their intrinsic and strategic characteristics. By intrinsic characteristics we mean the individual knowledge related to reservation and preferred prices, and also to the available capacity (or consumption needs if it is a Buyer agent). By strategic characteristics we mean the type of strategies the agent will employ to reach the objective of selling the

available capacity at the best price, if the agent is a Seller, or to buy the needed power if the agent is a Buyer. Seller Agents will compete with each other, since they are all interested in selling all their available capacity and in obtaining the highest possible market quote. On the other hand Seller Agents will cooperate with Buyer Agents while trying to establish some agreement that is profitable for both. This is a rich domain where it is possible to develop and test several algorithms and negotiation mechanisms for both cooperation and competition.

The increase in competitiveness creates opportunities for many new players or agents to enter the market; one of these players is the Trader. The introduction of this new entity, with well-defined responsibilities, allows liberalization and competition in the electricity industry to be developed and simplifies the way the whole process works with producers and consumers on the market and the relationship with the Market Operator. This entity participates in the market on behalf of consumers. It is an intermediary between them, who delegate on the Trader the purchasing of their needs, and the suppliers. One important feature of our simulator is the inclusion of this type of agent, usually not considered in related works.

The System Operator Agent is specific to the application domain, i.e. Electricity Markets, representing the responsible for the transmission grid and all the involved technical constraints. Every contract established, either through Bilateral Contracts or through the Pool, must be communicated to it, which analyses its technical viability from the Power System point of view (e.g. feasibility of Power Flow to attend all needs).

The Market Operator is responsible for the Pool. A Pool is based on an auction mechanism. Market price is established taking into account the auction mechanism, the previewed demand and the submitted bids. This agent is only present in simulations of Pool or Hybrid markets. It will receive the bids of Sellers and Buyers (according to the type of Pool – Symmetric or Asymmetric), analyse them and establish the marginal price and accepted bids. The process of determining the accepted bids is done according to the technical validation made by the System Operator. After, the Market Operator communicates to Sellers and Buyers the acceptance, or not, of their bids and, optionally, the market price.

### Seller and Buyer Agents Structure

Seller and Buyer agents are the two key players in the market. These agents have similar structure and a kind of symmetrical (due to their antagonistic objectives) behaviour, for this reason they are both treated in this section, however, whenever necessary the differences between them will be pointed.

The structure of these types of agents comprises three functional modules: Events Handler, Negotiation Management and Strategic Decision Making, plus one knowledge-based module: the Market & Individual Knowledge module. Figure 2 illustrates this structure.



Figures 2: Seller and Buyer Agents Structure

The Events Handler Module is responsible for all processes related with messages handling. Incoming messages are ordered by degree of importance and time of arrival. Out coming messages are sent only to those agents that are known to be possibly interested in that particular piece of information. Agents use ICL – Interagent Communication Language – to exchange messages between themselves.

During a negotiation period agents analyse and formulate several proposals. The Negotiation Management module contains all the processes related to this subject. The proposals received are analysed taking into account issues such as the price, quantity of energy and viability of the transaction (based on the technical analyses made by the System Operator). The process of formulating proposals results from the interaction of this module with the Strategic Decision Making module.

The Strategic Decision Making module is the most complex one. This is a module that analysis previous results and determines how to quote bids and which strategy to use. This module contains several dynamic strategies.

The market simulator is organised in several negotiation periods and Seller, Buyer and Trader Agents have strategic behaviour to define their desired price. These agents have time-dependent strategies, to change the price according to the remaining time until the end of the negotiation period; and behaviour-dependent strategies, to define the next period price according to the results obtained in the previous ones.

MASCEM implements four types of strategies to change the price during a negotiation period: Determined, Anxious, Moderate and Gluttonous. The

difference between these strategies is the time instant at which the agent starts to modify the price and the amount it changes. Determined agents maintain their prices constant during the negotiation period. Anxious agents start modifying the prices early in the negotiation period but by small amounts. Moderate agents will start changing the prices in the middle of the period by a small amount, and Gluttonous agents will only start changing the prices at the end of the negotiation period but by major amounts.

Although time-dependent strategies are simple to understand and implement, they are very important since they allow the simulation of important issues such as: emotional aspects and different risk behaviours. For example: an agent using a Determined Strategy is a risk indifferent one; while Gluttonous agents exhibit the behaviour more risk disposable, since they maintain the same price until very close to the end of the negotiation period, taking the risk of not selling.

To adjust price between negotiation periods, also referred as behaviour-dependent strategies, two different strategies were implemented: one called Composed Goal Directed and another called Adapted Derivative Following, see details in (Praça et al. 2004). These are important strategies that use the knowledge obtained with past experiences to define bid prices for the next periods.

To obtain an efficient decision support, Seller and Buyer agents also have the capability of using an algorithm, the Scenario Analysis Algorithm, described later.

The Market & Individual Knowledge module contains information about the organisational and operational rules of the market, as well as other agent commitments and capabilities, and about the agent itself: agent own capabilities, current availability, past experiences and strategies. Through the analysis of historical market results, this module constructs the profile of each agent in the market, particularly in what concerns their capabilities, reservation prices and expected prices.

### MASCEM: INTERFACE

MASCEM aim is to study the electricity spot market, where negotiation is divided into several negotiation periods, usually 24 or 48 periods, meaning next day's 24 hours or 48 mean-hours.

In spot markets the process of negotiation can be of many different forms. So, our simulator includes the possibility of negotiating through bilateral contracts, through a Pool, with either a single or a double uniform auction, and through a mixed market, where the agent must decide whether to negotiate in the auction and/or establish a bilateral agreement.

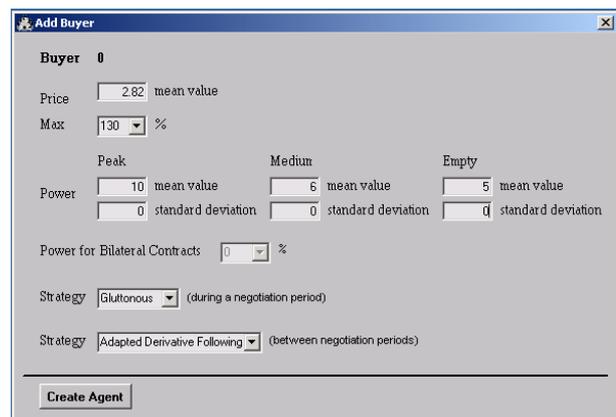
This is an important characteristic giving to the simulator a high degree of flexibility and usefulness, since the same scenario can be analysed through

different negotiation mechanisms (Praça et al. 2005). Figure 3 shows the interface screen to settle simulation parameters.



Figures 3: Scenario Definition

The user may also define Seller, Buyer and Trader Agents intrinsic and strategic characteristics. Figure 4 illustrates the parameters that characterize a Buyer Agent.



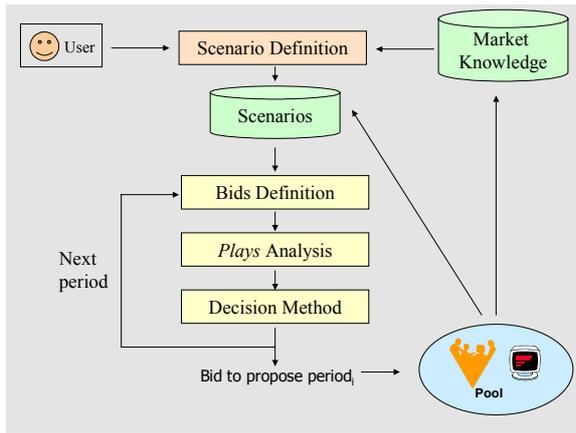
Figures 4: MASCEM screen for Buyer Agent input data

One important issue is the fact that Buyers consumption diagram is based on peak, empty and full consumption periods, and even within these periods, demand needs is allowed to fluctuate, while being represented by a probability distribution function.

### SCENARIO ANALYSIS ALGORITHM

This algorithm is based on the analysis of several bids under different scenarios. The aim of this algorithm is to help agents organize and extract knowledge from the information they gathered.

Agents use the information contained on their Individual and Market Knowledge Module to build the most probable scenarios for the next negotiation periods, and analyse them in order to take the most suitable behaviour to overcome their competitors, while assuring their needs will also be fulfilled. Figure 5 illustrates the algorithm steps.



Figures 5: Scenario Analysis Algorithm

### Scenario and Bid Definition

Each agent has historical information about market behaviour and about other agents' characteristics and behaviour. To get warrantable data, each agent uses techniques based on statistical analysis and knowledge discovery tools, which analyse the historical data.

With the information gathered agents can build a profile of other agents based on their expected proposed prices, limit prices, and capacities. With these profiles, and based on the agent own objectives, several scenarios, and the possible advantageous bids for each one, are defined. The agent should analyse the incomes that result from bidding its limit, desired prices, and competitive prices—those that are just slightly lower (or higher, in the Buyer's case) than its competitors' prices.

We call a *play* to a pair bid-scenario. After defining all the scenarios and bids, market simulation is applied to build a matrix with the expected results for each *play*.

### Decision Method

The matrix analysis with the simulated plays' results is inspired by the game theory concepts for a pure-strategy two-player game, assuming each player seeks to minimize the maximum possible loss or maximize the minimum possible gain (Fudenberg and Tirole 1991).

A Seller—like an offensive player—will try to maximize the minimum possible gain by using the MaxiMin decision method. A Buyer—like a defensive player—will select the strategy with the smallest maximum payoff by using the MiniMax decision method. In Buyers' matrix analyses, they select only situations in which they can fulfil all their consumption needs. They avoid situations in which agents will accept reduced payoff but can't satisfy their consumption needs completely.

This analysis not only provides the agent with decision support about the bid to propose in a Pool but also helps improve the negotiation mechanism for establishing bilateral contracts. With this information, the agent can evaluate a bilateral contract's potential benefits;

compare them to the benefits expected in a Pool, and make counterproposals.

### Scenario Analysis

The analysis of each period's results will update the agent's market knowledge and the scenarios to study. After each negotiation period, instead of considering how they might increase, decrease, or maintain their bid, agents use knowledge rules that restrict modifications on the basis of other agents' expected behaviour.

The knowledge rules update agents' bids in each scenario, but the number of scenarios remains the same. If at the end of a negotiation period the agent concludes — by analysing market results — that it incorrectly evaluated other agents' behaviour, it will fix other agents' profiles on the basis of the calculated deviation from real results.

## EXPERIMENTS

Several experiences were made to evaluate the benefits of the Scenario Analysis Algorithm (SAA). In this section we describe a small and simple scenario to illustrate them. The scenario used is very simple to let the reader better understand it. Table 1 presents Seller agent intrinsic characteristics and Table 2 those for Buyer Agents.

Table 1: Seller Agents

	Limit Price Cent/kWh	Pretended Price Cent/kWh	Energy kWh
S1	1.56	1.95	13
S2	2.86	3.57	12
S3	3.15	3.93	14

Table 2: Buyer Agents

	Limit Price	Pretended Price	Empty kWh	Full kWh	Peak kWh
B1	3.67	2.82	5	6	10
B2	3.30	2.54	6	6	10

Seller S1 is the most competitive one, having prices smaller than the other agents. This agent may increase its profits by raising its pretended price, without being overcome by competitors. The agent can use the SAA to take this conclusion. Let's see what happens in the Asymmetric Pool and in the Symmetric Pool.

### Asymmetric Market

In this type of market only Sellers are able to compete by presenting bids to the Pool. The Pool mechanism is usually the First Price Sealed Bid Auction. According to this mechanism, market price will be established based

only on Seller bids and previewed demand. So, agents will consider only profiles of other Seller agents.

Agent S1 may use SAA to build a profile of other agents and test bids that approach their expected bids but are sufficiently smaller to overcome them. Table 3 shows S1 profit fluctuation when compared to the same scenario simulated without the use of SAA.

Table 3: Seller S1 Profits in Asymmetric Market

	S1 Profit Fluctuation
Only S1 uses SAA	25 %
All Sellers use SAA	3 %
All Sellers except S1 use SAA	- 6 %

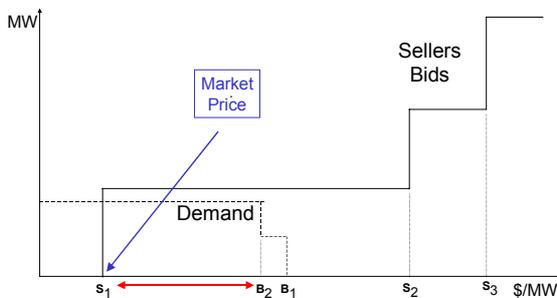
As we can see the major profit increase happens when S1 is the only agent using SAA. That makes sense since the other agents, which are less competitive, are not using SAA to conclude it and so they will keep trying to obtain their desired prices instead of making smaller bids, like those based on their limit prices. When the other Sellers also use SAA they conclude they cannot overcome S1, but they may compete with each other and so they reduce their pretended prices. In periods where S1 is not able to satisfy all consumption needs, such as in peak periods, the market price will be established by S2 or S3. Since they decrease their bids, market price will also be decreased, and so S1 profits will, indirectly, be reduced.

### Symmetric Market

In Symmetric Markets, the Pool functions according to a Double Uniform Auction, so both Sellers and Buyers are able to compete by presenting bids to the Pool.

When using SAA agents will also analyse bids that approach Buyers proposals that means that when studying Symmetric Markets the number of *plays* to analysed is higher.

As we can see in Figure 6, since Buyer Agents pretended prices are smaller than those presented by Sellers, the S1 Agent will conclude he is not able to increase bids as much as in the Asymmetric Market.



Figures 6: Symmetric Market Mechanism

Table 4 shows S1 profit fluctuation when compared to the same scenario simulated without the use of SAA.

Table 4: Seller S1 Profits in Symmetric Market

	S1 Profit Fluctuation
Only S1 uses SAA	12 %
All Sellers use SAA	-5 %
All Sellers except S1 use SAA	- 13 %

As expected S1 profits are smaller than the obtained in the Asymmetric Market. On one hand, according to the Pool mechanism, Buyer Agents are able to submit bids, so they are also able to influence market price. On the other hand, since Buyers pretended prices are smaller than Sellers prices, S1 concludes, through the SAA, that he can not raise its price, as much as in the Asymmetric Market, since in that market only Seller profiles are considered.

### MASCEM RELATED WORK

MASCEM ideas seem very promising and innovative, when compared to other approaches to study Electricity Markets, such as: as PowerWeb (PowerWeb-URL), where demand is always fixed and just single uniform auctions are studied; the Auction Agents for the Electric Power Industry, which only implements a Dutch Auction (EPRI-URL); the SEPIA-Simulator for Electric Power Industry Agents (Harp et al. 2000), which only implements a bilateral contracts market. Very relevant is the work of John Bower (Bower and Bunn 2001), Monclar e Quatrain (Monclar and Quatrain 2001), and Nicolaisen (Nicolaisen et al. 2001), however they are interested in studying only a particular market, the England and Wales market. However, our work is intended as a Decision Support Tool for the analysis and comparison of the negotiation mechanisms most used in Competitive Electricity Markets.

On the other hand, MASCEM has been selected, to illustrate agent technology application to electricity markets, as one of the six worldwide reference systems illustrating the use of agent based simulation in different types of markets (Praça et al. 2003).

MASCEM ideas are currently being applied to other types of markets, in the perspective of Electronic Commerce applications, under the scope of the project *Agents&Markets* (POSI/EIA/56260/2004) supported by the Portuguese Agency for Scientific Research (FCT).

### CONCLUSIONS

This work describes the use of Multi-Agent Simulation to study and understand the restructuring process of the electric power industry. In this context of recent and strong transformations, decision support tools are very welcome.

MASCEM is an agent-based simulator to study electricity spot markets. MASCEM agents include a Scenario Analysis Algorithm, that analyses market information and help agents define their bids and preview market results. Some experiences are reported suggesting the importance of this scenario analysis.

Although this is a very rich domain for illustration, there are many other areas where these ideas could also be fruitfully applied.

## REFERENCES

- Bower J., and Bunn, D. 2001 "Agent-based Simulation – An Application to the New Electricity Trading Arrangements of England and Wales". *IEEE Transactions on Evolutionary Computation*, 5, N°5 (Oct.), 493-503.
- EPRI-(Electric Power Research Institute), "Using Intelligent Agents to Implement an Electronic Auction for Buying and Selling Electric Power". Available: <http://www.agentbuilder.com/Documentation/EPRI/>.
- Fudenberg, D. and Tirole, J. 1991. "Game Theory". MIT Press.
- Harp, S.A., Brignone, S. and Wollenberg, B.F. 2000. "SEPIA: Simulator for Electric Power Industry Agents", *IEEE Control Systems*, 20, N°4 (August), 53-69.
- Monclar, F.R. and Quatrain, R. 2001. "Simulation of Electricity Markets: A Multi-Agent Approach", in *Proc. of the 2001 International Conference on Intelligent Systems Application to Power Systems*, 207-212.
- Nicolaisen, J., Petrov, V. and Tesfatsion, L., 2001 "Market Power and Efficiency in a Computational Electricity Market with Discriminatory Double-Auction Pricing", *IEEE Transactions on Evolutionary Computation*, 5, N° 5 (Oct.), 504-523.
- PowerWeb-URL. <http://stealth.ee.cornell.edu/powerweb/>.
- Praça, I., Ramos, C., Vale, Z. and Cordeiro, M. 2003. "MASCEM: A Multiagent System That Simulates Competitive Electricity Markets." *IEEE Intelligent Systems – Special issue on Agents and Markets* 18, No.6 (Nov/Dec), 54-60.
- Praça, I., Ramos, C., Vale, Z. and Cordeiro, M. 2004. "Intelligent Agents for the Simulation of Competitive Electricity Markets." *International Journal of Modelling and Simulation* 24, No.2, 73-79.
- Praça, I., Ramos, C., Vale, Z. and Cordeiro, M. 2005. "Intelligent Agents for Negotiation and Game-based Decision Support in Electricity Markets." Selected for publication in an upcoming special issue of the *Engineering Intelligent Systems Journal*.

## AUTHOR BIOGRAPHIES



**ISABEL PRAÇA** is an assistant professor of computer engineering at the Polytechnic Institute of Porto's Institute of Engineering. Her research interests are in multiagent simulation and decision

support systems. She received her PhD in computer engineering from the University of Tás-os-Montes e Alto Douro. Contact her at: [isp@dei.isep.ipp.pt](mailto:isp@dei.isep.ipp.pt).



**CARLOS RAMOS** is a coordinator professor of computer engineering and director of the Knowledge Engineering and Decision Support Research Group at the Polytechnic Institute of Porto's Institute of Engineering. His research interests are artificial intelligence and decision support systems. He received his PhD in electrical engineering from the University of Porto. Contact him at: [csr@dei.isep.ipp.pt](mailto:csr@dei.isep.ipp.pt).



**ZITA VALE** is a coordinator professor of electrical engineering at the Polytechnic Institute of Porto's Institute of Engineering. Her research interests are decision support and artificial intelligence. She received her PhD in electrical engineering from the University of Porto. She is a member of the IEEE, IEE and ACM. Contact her at: [zav@dee.isep.ipp.pt](mailto:zav@dee.isep.ipp.pt).



**MANUEL CORDEIRO** is a full professor at the University of Tás-os-Montes e Alto Douro. His research interests are rational energy use, grounding systems and power systems applications. He received his PhD in electrical engineering from the University of Tás-os-Montes e Alto Douro. Contact him at: [cordeiro@utad.pt](mailto:cordeiro@utad.pt).

# AGENT BASED SIMULATION FOR GROUP FORMATION

Goreti Marreiros, Ricardo Santos and Carlos Ramos

GECAD – Knowledge Engineering and Decision  
Support Group  
Institute of Engineering – Polytechnic of Porto  
R. Dr. António Bernardino de Almeida  
4200-072 Porto, Portugal  
{goreti, csr}@dei.isep.ipp.pt

José Neves

University of Minho  
Largo do Paço  
4700-320 Braga, Portugal  
jneves@di.uminho.pt

## KEYWORDS

Group Decision Making; Agent Based Simulation

## ABSTRACT

Group decision making plays an important role in today's organisations. The impact of decision making is so high and complex, that rarely the decision making process is made just by one individual. The simulation of group decision making through a Multi-Agent System is a very interesting research topic. The purpose of this paper is to specify the actors involved in the simulation of a group decision, to present a model to the process of group formation and to describe the approach made to implement that model. In the group formation model it is considered the existence of incomplete and negative information, which was identified as crucial to make the simulation closer to the reality.

## INTRODUCTION

The problem of group decision-making has gained great relevance in the scope of Decision Support Systems, which were initially designed as individual tools. Quickly those tools have demonstrated to be limited, in the sense that in today's organizations several persons, entities or agents are involved in most of the decision processes. In that way, the decision problems are considered from different points of view, with different opinions about the importance of the decision criteria (for example, in the purchase of a car we will be able to consider criteria like price, technical characteristics, design or manufacture). Groups of individuals have access to more information and more resources (Nunamaker et al., 1991), and that will (probably) allow reaching "better" and quicker decisions.

In the last years Group Decision Support Systems (GDSS) research focused in asynchronous (different-time) and ubiquitous (different-place) tools, and several web-based GDSS have been developed (Marreiros et al, 2004a; Karacapilidis and Papadias, 2001; Marreiros et al, 2005).

Despite of the quality of developed GDSS, they present some limitations like, for instance, the modelling of the group decision making problem through a Multi-Agent System.

The use of Multi-Agent Systems seems very suitable to simulate the behaviour of groups of people working together and, in particular, to group decision making modelling, because it allows:

- Individual modelling – each participant of the group decision making can be represented by an agent that will interact with other agents. Agents can be modelled with social and emotional characteristics in order to become more realistic.
- Flexibility – with this approach it is easy to incorporate or remove entities. It is also possible to change the characteristics of the individuals, for instance, in order to analyze its impact in the group behaviour.
- Data distribution – frequently, in group decision making, participants are geographically distributed. Agents that represent participants, with this approach, may be running in different machines.

If the group decision making problem is to be resolved by a group of intelligent agents then it is, first of all, necessary to constitute the group.

In Multi-Agent literature the references to tasks related to team formation are in the area of cooperative problem solving. Wooldridge and Jennings identified four stages in the resolution of the cooperative problem solving process (Wooldridge and Jennings, 1999):

- Recognition of the problem - an agent identifies the potential for cooperation.
- Team formation - the agent solicits assistance for the identified problem.
- Plan formation - the newly formed collective attempts to construct an agreed joint plan.

- Execution - members of the collective play out the negotiated roles.

Several authors consider that the focus of team formation is the agent's mental state and its motivation to form teams and collaborate (Cohen et al, 1997; Wooldridge and Jennings, 1999). Dignum and his colleagues present a theory for agent team formation that is based on structured dialogues, with an emphasis on persuasion (Dignum, 2000).

This paper aims to specify the different tasks involved in the formation of a group of agents that will be the main actors in the simulation of a group decision making meeting. This work is included in ArgEmotionAgents project (POSI / EIA / 56259 / 2004 - Argumentative Agents with Emotional Behaviour Modelling for Participants' Support in Group Decision-Making Meetings), which is a project supported by FCT (Science & Technology Foundation – Portugal) envisaging the use of Multi-Agent Systems approach for simulating Group Decision-Making processes, where Argumentation and Emotion components are specially important.

The paper is organised as follows. The following section presents the main participants in the process of a group decision making simulation. After that, it is presented a model for multi-agent group formation, where it is considered the incomplete information handling and the existence of explicit negation. Implementation details are discussed before the final section that presents some conclusions about the proposed model and point out some directions for future work.

## INTERVING AGENTS IN SIMULATION OF GROUP DECISION MAKING AND THEIR ROLE

In our opinion, the simulation of group decision making through a multi-agent system implies the need for different kind of agents (Marreiros et al, 2004b):

- Participant Agents (AgP) – these agents will simulate the role of persons in the group decision making process. The agents are dotted of social and emotional characteristics that will personalize its behaviour. Each agent will have a model of himself and a model of the others, that will be refined with the information received during simulations.
- Facilitator Agent (AgF) – this agent will help the responsible for the simulation in its organization. According to coordinator instructions, will require the formation of a group of agents with skills to understand and resolve a specific problem. This agent will also interview during the simulation, sending for instance stimulus messages to the less participative agents, and will summarize the results of the simulation.

- Register Agent (AgR) – for an agent to become part of the community of participant agents (AgP), it should first make a registry, making available some public information about its profile.
- Voting Agent (AgV) – experience tells that almost all the group decision making meetings have one or more voting phases. This agent will be responsible for the tasks related with the voting simulation process.
- Information Agent (AgI) – the agent that detains information about the different proposals (alternatives) that will be evaluated by the group of agents during the group decision making simulation.

This paper focus in the first three kinds of agents (AgP's, AgF and AgR), because these agents will be directly involved in the group formation process. In the following section we will present a model to support the tasks associated with the group formation process.

## MODEL FOR GROUP FORMATION

After the identification of the main actors in the simulation of a group decision making process and the characterization of its role, it is necessary to establish the steps for the creation of a community of participant agents. The importance of maintaining a community of agents during several simulations is directly related to the need of obtaining information about the credibility, the reputation of the agents, as well as past behaviours, promises that have been made, etc. This information will be very useful when agents are participating in the simulation of the group decision meeting.

### Inclusion in the Community of Participant Agents

The selection of agents to participate in the simulation of a group decision making is made from a community of participant agents (AgP). First of all; the agent must be registered to be selected. Some information about potential participant should be available in order to allow the acceptance of this participant agent by the Register Agent (AgR):

*Agent (Id):: area\_of\_expertise,  
interest\_topics,  
availability.*

Where *Id*, *area\_of\_expertise*, *interest\_topics* and *availability* represent respectively the identification of the agent, the set of areas where the agent is expert, the interest topics for the agent and its availability at that moment.

The community of participant's agents is a set of N agents, AgP<sub>1</sub>, AgP<sub>2</sub>, ..., AgP<sub>N</sub>, denoted by AgP. The

availability of each agent, in the community, can be classified according three states: uncommitted, committed, or in action. An **uncommitted** agent is available to participate in a simulation of a group decision making. An agent **in action** is already involved in a simulation that is running. At last, a **committed** agent has agreed to be part of a group, but the simulation has not yet started.

### Incomplete Information

The Knowledge Base (KB) of the Register Agent was defined in the previous section. But agents do not have a way to represent explicit negative information in the KB, as for instance, topics that do not have any interest to the agent. In other words, instead of being based on the Closed-World Assumption (which tell us that any missing information in the KB is false), the knowledge that something is false must be explicitly represented in the KB. In this sense, the KB has two different types of knowledge: the positive knowledge (what is known to be *true*), and the negative knowledge (what is known to be *false*). All the rest is *unknown*. Suppose that in the KB of the AgR the information related to the areas of expertise of the AgP<sub>i</sub> identified as Michel is represented in program 1:

```
area_of_expertise('Michel', tourism).
¬area_of_expertise('Michel',cousine).
```

Program 1- representation of the information related to the expertise areas of a specific agent

If the KB is questioned about if the area of expertise of Michel is Pharmacy the answer should be unknown, because there is no information related to that.

Following the approach described in Analide and Neves (2002) situations of incomplete information may involve two kinds of nulls. The Extended Logic Programming (ELP) will be the approach followed for the knowledge representation.

The first type of Null value is an undetermined unknown value, which means that, there is a missing value but the possible instantiations for that value are completely unknown. Suppose that one of the agents belong to the community AgP, and in the moment of register he does not specify his interest topics, just inform that he has interest topics. This means that the interest topics of this agent are unknown. Program 2 represents the use of this kind of Null.

```
¬topic_of_interest(A,B):-
    not topic_of_interest(A,B),
    not exceptiontopic_interest(A,B).
exceptiontopic_interest(A,B):-
    topic_interest(A, something).
topic_of_interest('Jonh', something).
```

Program 2 : representation of information related to the agent interest topics

The other type of Null value represents information of an enumerated set. Following the previous example suppose that an agent does not give information related to his availability, then in this case there are three exceptions allowed: uncommitted, committed or in action. This could be represented as in program 3:

```
¬availability(A,B):-
    not availability(A,B),
    not exceptionavailability(A,B).
exceptionavailability(A,B):-
    availability(A, availability).
exceptionavailability('John',committed).
exceptionavailability('John',uncommitted).
exceptionavailability('John',in_action).
```

Program 3 : representation of information related to the agent availability

### Multi-Agent Model for Group Formation

The simulations of group decision making processes will be coordinated by the decision maker agent. This agent will be responsible by the phase identified as problem recognition (Wooldridge and Jennings, 1999). The coordinator of the simulation has to identify the quantity of participants that will be needed to form the group and the lifetime of the simulation.

After identifying the above referred aspects, the coordinator, asks the Facilitator Agent (AgF) to form a team. The Facilitator Agent will send a request to the Register Agent (AgR) to contact potential interested agents:

```
Request_form_group (AgF, AgR, K, Si, expertise_areas,
lifetime)
```

In the previous message *K* is the necessary quantity of agents, *S<sub>i</sub>* is the identification of the future simulation, *expertise\_areas* are the areas of expertise that agents must have and lifetime is the number of periods of the simulation. After receiving this request, the Register Agent will send a request to the community of participant agents with the information of the required expertise and the lifetime of the simulation.

```
Request_form_group (AgR, AgP, Si, expertise_areas, lifetime)
```

The agents of the community may answer to the facilitator with three possible types of answers: *interest\_in\_participate*, *not\_interested* or they could simply ignore the request.

```
interest_in_participate(AgPj, AgF, Si)
not_interested(AgPj, AgF, Si)
```

To realize the selection of the interested agents, the AgR will analyse the received answers and verify if they are in accordance with his KB. If the AgR identified agents that did not answer or say that are not interested and, as being an agent with special interest to the group, he could establish a direct contact.

```
Request_form_group (AgR, AgPj, Si, expertise_areas, lifetime)
```

The formation of the final group is based in the received answers and in the inference process realized through the KB of the register agent, with the objective of maximizing the knowledge of the group.

The AgR will send a message to the AgF with a possible group, which should be approved by the simulation coordinator.

*Proposed\_group(AgR, AgF, {AgP<sub>x</sub>..., AgP<sub>y</sub>})*

If the simulation coordinator approves the group, the AgF will inform the agents and the simulation start. Otherwise, the facilitator will request other potential group to the AgR.

## IMPLEMENTATION ISSUES

A prototype of the multi-agent model for group formation is being developed in order to validate the model and also to generate simulation results that can be analyzed. In this section we review the multi-agent platform that is being used to implement the proposed model, and present some details of our implementation.

### Multi-Agent platform

The prototype is being developed in Open Agent Architecture (OAA) and Java. OAA (Cohn et al., 1994; Martin et al, 1999) developed at the Artificial Intelligence Center of Stanford Research Institute, is a framework for integrating a community of heterogeneous software agents in a distributed environment. The main features of OAA platform are:

- Openness – agents may be written in various languages (Prolog, Java, ANSI C/C++, LISP) and operating systems. The language and operating systems barrier are in this way minimized.
- Distributive – agents may be distributed by multiple networked machines. This is a very important advantage to increase simulation runs, specially in scenarios where are involved a great number of agents.
- Extensible – it is possible to add or remove agents at run-time, allowing the creation of flexible and robust scenarios.
- Mobile – simple OAA based user interfaces can run on personal digital assistants (PDA).

The OAA framework is composed by two distinct types of agents: the OAA facilitator agent and client agents. Usually there is only one facilitator by application, but it is also possible to have multiple facilitators. The OAA facilitator agent is responsible for tasks related to coordination and communication. The OAA Facilitator has, in somehow, behaviour similar to a router in the sense that it is responsible for distributing data and messages among its set of client agents. The first communication between two clients agents is

necessarily made through the facilitator, after what they can communicate directly if necessary. The other type of agent is the client agent that, when invoked, creates a connection with a facilitator and informs him of what services he provides.

OAA has an Inter-agent Communication Language (ICL) that is shared by all agents independently of the language in which they are programmed or the operating system of the machine where the agents reside. The ICL language is close to KQML (Mayfield et al., 1996).

OAA is widely accepted. There are more than 30 applications developed based on this platform (Cheyer and Martin, 2001).

### Prototype

At this moment the various agents are created in java as a java thread, but we are considering the implementation of the AgR in PROLOG because, as being responsible for the selection of the group elements, this agent has to reason based in incomplete information.

In figure 1 it is possible to see the public profile of one of the agents that belong to the community of participant agents (AgP).

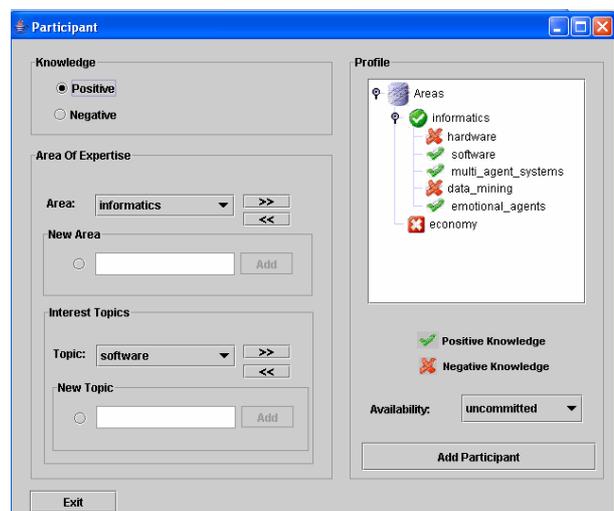


Figure 1: Creation of participant agents (Profile definition)

In this case the agent is expert in informatics, particularly in topics related to software, multi-agent systems and emotional agents, and he has no knowledge in topics related to data mining and hardware (negative information). The agent has no knowledge also in economic area.

In figure 2 it is possible to see all the community of participant agents. The agent created above belongs to this community.

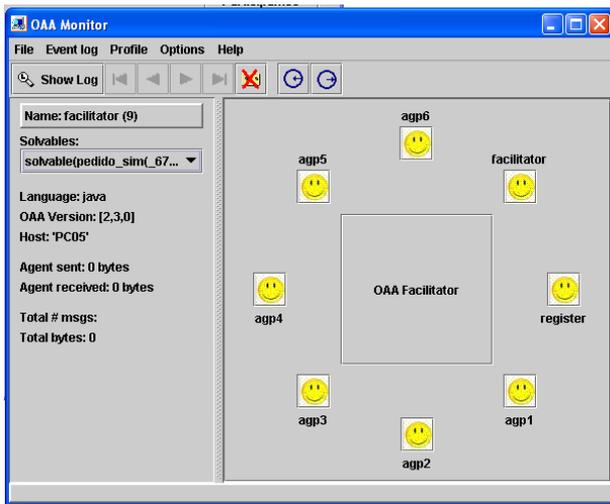


Figure 2: AgP Community

The AgR works closely to the OAA facilitator agent. The OAA facilitator is responsible for all the communication and coordination tasks, and the AgR is responsible for the participants profile maintenance and for the group formation process.

## CONCLUSION

The simulation of group decision making through a multi-agent system allows, in particular, studying the behaviour of agents along the time. The process of group formation could be done by the person responsible for the simulation, or through automatic selection of the group members, where the agents manifest interest or not in the participation in a specific group decision simulation.

This paper proposes a multi-agent model to support the process of group formation, where it is considered the existence of incomplete and negative information about the skills of the potential group members. For the selection of the effective group it is considered, the total knowledge of the group, besides the abilities of each one of the elements.

Future work will include the continuation of the implementation of this model as a first step to the simulation of a multi-agent group decision making. It is also our propose to incorporate some of the group decision simulation results in the selection phase of a future simulation. An example could be for instance the group willingness characteristic. As result of several simulations it could be estimated the willingness of a specific agent to group work. The agent reputation is another characteristic that will be considered in the group formation process.

## REFERENCES

- Analide, C. and J. Neves. 2002. "Antropopatia em Entidades Virtuais". I Workshop de Teses e Dissertações em Inteligência Artificial (WTDIA'02), Porto Galinhas, Brazil.
- Cohen, P., H. Levesque, I. Smith. 1997. "On Team Formation", In J. Hintika and R. Tuomela, Contemporary Action Theory, Synthese.
- Dignum, B. Dunin-Keplicz, and R. Verbrugge. 2000. "Agent Theory for Team Formation by Dialogue". Agent Theories Architectures and Languages, 150-166.
- Karacapilidis, N. and D. Papadias. 2001. "Computer supported argumentation and collaborative decision making: The Hermes system", Information Systems, Vol. 26 No. 4, 259-277.
- Marreiros, G.; J. P. Sousa and C. Ramos. 2004(a). "WebMeeting - a group decision support system for multi-criteria decision problems". International Conference on Knowledge Engineering and Decision Support ICKEDS04, 63-70.
- Marreiros, G.; C. Ramos and J. Neves. 2004(b). "Defining a model for agent-based participant support in group decision meeting". International Conference on Knowledge Engineering and Decision Support ICKEDS04, 71-77.
- Marreiros, G.; C. Ramos and J. Neves. 2005. "Modelling group decision meeting participants with an Agent-based approach". Selected for publication in an upcoming special issue of the International Journal of Engineering Intelligent Systems.
- Nunamaker, J.F., A.R. Dennis, J.S. Valacich, D.R. Vogel, J.F. George. 1991. "Electronic Meeting Systems to support group work". Communications of the ACM, vol. 34, No. 7, 42-58
- Wooldridge, M and N.R. Jennings. 1999. "The Cooperative Problem Solving". Journal of Logic and Computation. Vol 9, No. 4, 563-593.
- Mayfield, J.; Labrou, Y. and Finin, T. 1996. "Evaluation of KQML as an Agent Communication Language" Intelligent Agents Volume II -- Proceedings of the Workshop on Agent Theories, Architectures, and Languages. Lecture Notes in Artificial Intelligence, Springer-Verlag.
- Cohen, Philip R. and Cheyer, Adam J. and Wang, Michelle and Baeg, Soon Choel. 1994. "An Open Agent Architecture", in AAAI Spring Symposium, 1-8.
- Martin, D. and Cheyer, A and Moran, Douglas B. (1999). "The open agent architecture: a framework for building distributed software systems", Applied Artificial Intelligence, Vol. 13, No. 1-2, 91-128.
- Cheyre, A. and Martin D. "The Open agent Architecture", Journal of Autonomous Agents and Multi-agent systems, Vol. 4 No.1, 143-148.

## ACKNOWLEDGEMENTS

This work is partially supported by ArgEmotionAgents project (POSI / EIA / 56259 / 2004 - Argumentative Agents with Emotional Behaviour Modelling for Participants' Support in Group Decision-Making Meetings), which is a project supported by FCT (Science & Technology Foundation – Portugal).

## **AUTHOR BIOGRAPHIES**

**GORETI MARREIROS** is an assistant professor of Computer Engineering at the Institute of Engineering – Polytechnic of Porto (ISEP/IPP) and researcher at the Knowledge Engineering and Decision Support Research Group (GECAD). She is preparing her PhD thesis on multi-agent simulation to support argumentation and group decision making. She received her MSc in Information Management from the University of Porto in 2002.

**RICARDO SANTOS** is a researcher at the Knowledge Engineering and Decision Support Research Group (GECAD). He received his BcS in Computer Science from the Polytechnic of Porto (IPP).

**CARLOS RAMOS** is a Coordinator Professor of Computer Engineering and director of the Knowledge Engineering and Decision Support Research Group (GECAD) at the Institute of Engineering – Polytechnic of Porto (ISEP/IPP). His research interests are Artificial Intelligence and Decision Support Systems. He received his PhD in Electrical and Computer Engineering from the University of Porto in 1993.

**JOSÉ NEVES** is Full Professor of Computer Science at the University of Minho. He got his PhD in Computer Science from Harriet Watt University, Edinburgh, Scotland in 1984. His current research activities span the fields of Extended Logic Programming, Knowledge Representation and Reasoning systems and Multi-agent Systems.

# Flexible Models of Service Systems Based on the ABAsim Architecture

Michal Lekýr  
Norbert Adamko

Faculty of Management science and Informatics  
University of Žilina  
Univerzitná 8215/1, 01026 Žilina  
E-mail: lekyr@utcpd.sk, Norbert.Adamko@fri.utc.sk

## KEYWORDS

Agent simulation, ABAsim architecture, service system modeling, CASE tool

## ABSTRACT

In the process of designing new service systems and studying operation of existing systems computer simulation becomes very useful for its flexibility and similarity of modeled systems with the real world. This paper gives a brief overview of agent oriented architecture ABAsim, which provides its users with high quality founding base for modeling large-scale service systems. ABAsim architecture is based on hierarchically organized rather reactive than proactive agents. This architecture was applied to model the operation of various types of service systems. These models have been designed according to the general model of a service system described in this article. Specialized CASE tool for design and creation of simulation models developed under the architecture is also discussed.

## INTRODUCTION

Computer simulation is an experimental method which is often used to study the behavior of various kinds of service systems. The process of modeling such systems is rather complicated task with high potential for errors, especially when the modeling is carried out by scientists holding their expertise in other fields than computer science. To reflect real system behavior in the modeled system requires experience even beyond the field of computer science. Defining the complex methodology for service systems simulation and modeling and creating a CASE tool supporting model design and maintenance based on the methodology is the effort to make this process easier. This paper briefly describes the ABAsim architecture of simulation models. This architecture provides flexible base for creation of simulation models of large service systems. The methodology of modeling such systems is the main focus of this article. It also gives an overview of the

supporting tool for this architecture called the ABAbuilder.

## THE ABAsim ARCHITECTURE

Architecture *ABAsim* was developed mainly for simulations of large service systems. The simulation models of simple real systems could be composed of only one agent; however the simulations of complex service systems are obviously connected with *multi-agent approach* using the agents within some organizational structures. Let us remark that the philosophy of ABAsim architecture was also partly inspired by the paradigm of reactive agents, which is based on the society of rather reactive than proactive agents (Nwana, H. 1996). The intelligence of that society *emerges* when observing the whole community and not the separate (relatively little intelligent) members.

Multi-agent hierarchical system can be demonstrated on the example shown on Figure 1. On the top of the hierarchy is agent *A-0* (agent of simulation, also called boss). This agent utilizes the set of subordinated specialized agents in order to fulfill defined tasks connected with surroundings (agent *A-1*) or with service system itself (agent *A-2*). Agent *A-1* is responsible for the management of whole surroundings of the system (arrival of customers from surroundings, departure of customers from the system, etc.). The agent divides the management to three partial managements of different transportation modes (road, railway and water). We say that agent *A-1 delimits* the surroundings management to three peer agents *A-3*, *A-4* and *A-5*, which inform their superior agent about the important facts connected with their mode of transportation. Agent *A-1* can also send important multi-modal information to its *subordinates*. It is expected that all three mentioned modal agents carry out the same kinds of management operations.

The agent responsible for the service system itself is *delegating* some portion of its competences to hierarchically lower ranked colleagues. It is obvious from the mentioned figure that agent *A-2* utilizes its subordinate agents *A-6* to *A-9* to manage all required tasks.

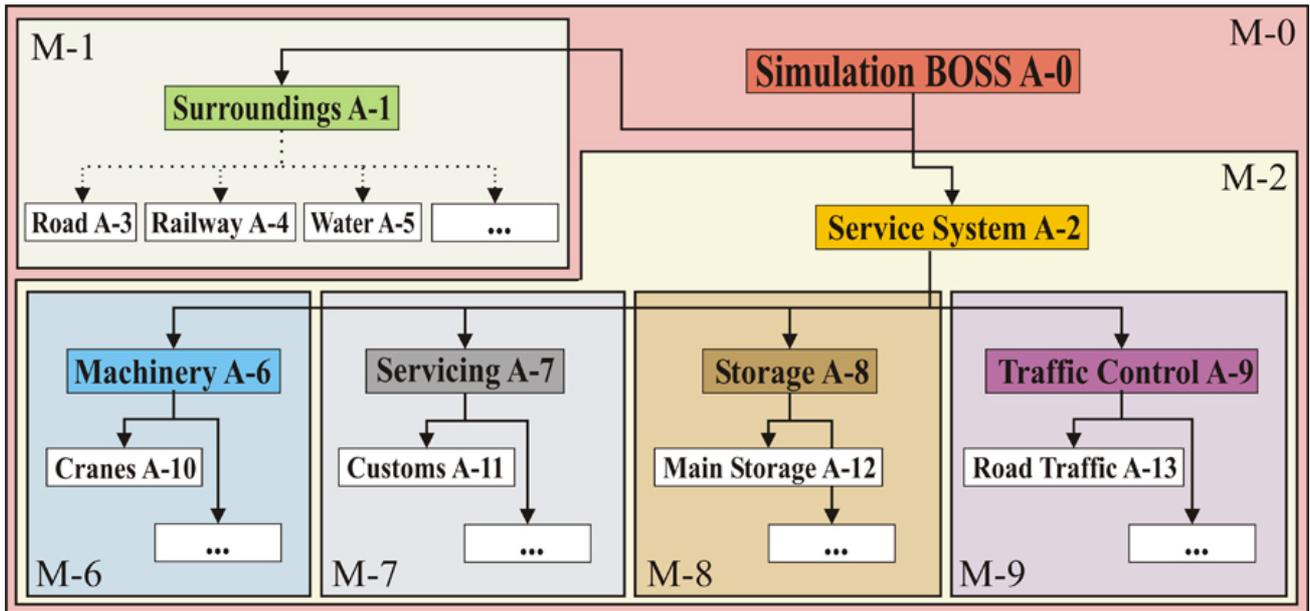


Figure 1: Agent Hierarchy in a Simulation Model of a Service System

Another point of view is that agent  $A-0$  is a representative (boss) of the whole model  $M-0$  (simulation model of service system). That fact is denoted as:  $A-i = boss(M-i)$  and  $M-i = model(A-i)$ . The models  $M-1$  and  $M-2$  represent sub-models of model  $M-0$ . The models on the lowest hierarchical level (the leaves) are always realized as one-agent models. Let us finally denote the structure (down to the third level) of model  $M-0$  in the form of algebraic expression:  $(M-1[M-3, M-4, M-5], M-2(M-6, M-7, M-8, M-9))$ , where the pairs of brackets encapsulate the models, which were created as an act of delimitation and the pairs of parentheses encapsulate the models with delegated competences.

### Agent decomposition

Each of agents can be decomposed to four following groups of *internal components*:

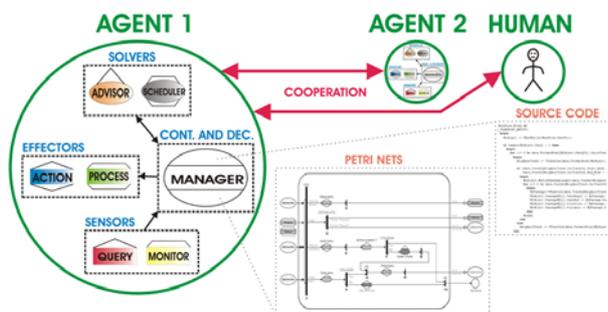


Figure 2: Agent Decomposition

- a) The first *control* and *decision making* component, called *manager*, is responsible for making decisions and inter-agent communication. In addition, the manager represents the central agent component because it initiates the work of other internal components and also can communicate with all of them.
- b) The group of *sensors* is specialized to information mining from the state space. That group is composed of two kinds of components - the *query* delivers the required kinds of information instantly and the *monitor* makes scanning of state space during some time interval and continuously brings important pieces of information to the manager.
- c) The next group of *solvers* provides problem solutions for the manager, which can accept them or asked for alternative ones. The *advisor* is passive component that is able by return to react only to the manager's request for delivery of problem solution proposal. The typical forms of an advisor can be represented e.g. by optimization algorithms, neural networks or human operator. On the other hand the *scheduler* (focused on the restricted scope of problems) works for manager continuously on the base of either a priori created rosters/schedules (e.g. connected with allocations of resources) or making its own dynamic forecast for defined time interval.
- d) The last component group includes *effectors* (actuators), which make changes of system status after getting corresponding instructions from the manager. The other kinds of agent components are not allowed to make the mentioned changes. The *action*-component makes instant state changes (e.g.

switch a traffic light over, close the train doors), while the *process* (e.g. the crane movement) makes them continuously till its tasks are finished.

The effectors, sensors and solvers are briefly called manager's *assistants* and can be further distinguished as:

- *Continual assistants*, activities of which take some interval of simulation time (processes, monitors and schedulers).
- *Instant assistants* (actions, queries and advisors), which are active only within one instant of simulation time.

### Inter-agent communication

Let us present basic principles of solving the above-mentioned problem within ABASim architecture profiting by hierarchical organization of sub-models. The model  $M = \{M-0, M-1, M-2, \dots, M-s\}$  represents a set of sub-models reflecting a set of agents  $A = \{A-0, A-1, A-2, \dots, A-s\}$ . For all  $I = 1 \dots S$  we denote:

- $I_{Ag}(A-i)$  – a set of messages directly addressed to an agent  $A-i$ , which is able to respond to them (a set of *direct-messages* of an agent  $A-i$ ),
- $I_{Sub}(A-i)$  – a set of direct messages of all agents, which are subordinated to  $A-i$  (a set of *mediated-messages* by an agent  $A-i$ ),
- $I_{Mod}(A-i) = I_{Ag}(A-i) \cup I_{Sub}(A-i)$  – a set of messages, which can be elaborated by a *model(A-i)* regardless of its structure (a set of *model-messages* in respect of *model(A-i)*).

Each message kind can be utilized in three following variants:

- Message of kind  $mess(A-i, A-j, agent)$  is sent by an agent  $A-i$  to an agent  $A-j$ , which accepts it only if  $mess \in I_{Ag}(A-j)$  – it is a standard *addressed message*.
- Another message type  $mess(A-i, A-j, model)$  is dispatched by a sender in case it is not known a concrete target addressee/agent (or due to flexibility it is not desired to be determined). However, it is evident that *model(A-j)* is qualified for its elaboration. That message is treatable if  $mess \in I_{Mod}(A-j)$  and is called *partially addressed message*.
- Finally a message kind  $mess(A-i, *, *)$  represents a *non-addressed message*, which is elaborated in case that  $mess \in I_{Mod}(A-0)$ .

### MODELING SERVICE SYSTEMS

A *service system* is understood as a system focused on elaboration of *orders* (attendance of *customers*) and

execution of *services* related to them. The mentioned services can further initiate another set of orders. The service systems include a wide class of various systems e.g. factories, any transportation and logistic nodes/junctions, repair shops etc. The natural and technical systems do not belong to that domain.

The service system structure can be considered (from the viewpoint of order elaboration) as strictly *hierarchical*. The order (the customer) entering the system (e.g. produce a car) calls recursive sequence of suborders according to the rules of competence redistribution. All system elements (subsystems) work synergically, unlike the majority of natural systems, with the common goal to elaborate the order. Thus, the architecture is not determined to work with the processes of the following kinds: evolution, competition and parasitism. The entities of a service system (orders/customers and resources) can be divided into specialized classes with the same behavioral rules for included entities. It means that the responsibility for the behavior of entities is taken by their superior subjects (agents) and hence there is no reason to consider entities as agents.

The service systems represent usually large-scaled systems. It is mostly necessary to transfer service resources to the customer (or vice versa), in order to realize the service activity. Hence, there are typical frequent complex transposition processes within those systems.

### System Entities

The main function of a service system is to serve customers which enter and leave the system. To fulfill this task, resources of various kinds are necessary. The customers and all the resources are represented by system entities, which are controlled by system agents. Each entity is controlled by one agent at a time. Agents can relegate the control over specific entity to another agent. Entities are not capable of managing their own acts. There are several dimensions to classify system entities:

Firstly entities may be classified by their mobility, i.e. by their ability to be moved around some network by their controlling agent:

- *Static entities* – the change in their position in space and time is not possible (e.g. building, gate, parts of the infrastructure, etc.)
- *Dynamic entities* – the position change can be achieved by their controlling agent. They are further subdivided into:
  - *Mobile entities* – an agent controls their movement without support of another entity (e.g. car, plane, vessel, etc.)

- *Stationary entities* – their movement has to be supported by another entity (e.g. container, chassis, etc.)

Second classification is according to their appearance in the system:

- *Endogenous entities* – they originate inside the system (e.g. train sets, etc.)
- *Exogenous entities* – they enter and leave the system, originate outside of the system (e.g. car, plane, etc.)

Entities in the system are stored in so called *the entity layer* which is part of the ABAsim architecture. Each entity has *access rights* defined, so that only authorized agents are allowed to change their status. This approach concentrates all of the system entities at a single location and makes them easily accessible to all of the system agents.

### The Model Design

To design the system architecture, the system has to be divided according to its main tasks. Therefore, a different kind of agent for each one of the main tasks to be done has been modeled. Unlike in (Rebollo et al. 2000) where each system entity is controlled by a separate agent, in our system each agent controls all the entities belonging to the same category. For example in a system with multiple cranes, such as gantry cranes, the operation of these cranes would be controlled by one agent called *Agent GantryCranes*.

The ABAsim architecture, which is designed for modeling large service systems, allows us to divide the problem into subproblems. Each subproblem can be solved by a specific agent, which uses its subordinate agents to accomplish desired tasks. This approach enhances flexibility of the whole system.

According to our understanding of service system, we propose the hierarchical architecture of agents in a simulation model of a service system shown on Figure 1. The whole model *M-0* is represented by its boss agent *Simulation BOSS A-0*. This agent has two subordinated agents: *Surroundings A-1* and *Service System A-2*.

Submodel *M-1* represented by agent *A-1* is used to model the behavior of the surroundings related to the modeled service system. Agent *A-1* can have various number of underling agents. The number of these agents depends on the service system characteristics. For example, when modeling a container terminal, agents *Water* and *Road* will be used to supply “customers” for the service system. Agent *Railway* would be alternatively used in case railway directly interacts with the system. Agents in the *M-1* submodel are responsible for management of *exogenous entities* before entering and after leaving the system.

Agent *Service System A-2* represents the submodel *M-2* which is the model of the service system itself. It’s delegating its competences to hierarchically lower ranked colleagues: *Machinery A-6*, *Servicing A-7*, *Storage A-8* and *Traffic Control A-9*. Both *endogenous* and *exogenous* entities are being managed by *M-2*. The main features of agents in *M-2* are specified in the following section.

*Machinery A-6* is the head of the subsystem *M-6* and it utilizes all of the agents managing the operation of *endogenous* entities (e.g. cranes, reach stackers, straddle hoists, etc.) Each *M-6* member controls entities of the same category and it’s capable of making decisions about the usage of each entity it manages. Decision making of these agents is supported by their *advisors*, which are commonly implemented as some kind of optimization algorithm. The way the logic is implemented influences the function of all controlled entities.

Agent *Servicing A-7* is responsible for submodel *M-7* which has been divided according to the different services. Each service type has assigned one agent which is responsible to fulfill desired tasks. Agents in this group are responsible for serving “customers” at all checkpoints in the system.

*M-8* is a subsystem with its leading agent *Storage A-8*. This part of the model is responsible for managing system storage resources. Agent *A-8* has a function of the director of all the storage areas in the system. Subordinated agents provide rules for storing entities in different storage sections of the system.

Finally the traffic control subsystem *M-9* is controlling all the traffic in the entire system. The process of transportation of an entity from starting point to its destination point is guided by corresponding agent. For example, when truck is moving from the entrance gate to the control bridge, this process is controlled by the *Road Traffic A-13* agent. This agent is responsible for safe journey, avoiding all possible collisions with other entities and positioning in the queue of waiting vehicles.

### Modularity and Model Re-Use

Structure of a simulation model as well as of individual agents represents such features, which highly support a flexibility of simulation model within ABAsim architecture. In spite of that fact the message-oriented communication mechanism does not enable such level of flexibility, which would allow changing a sub-model structure without a need of additional changes within other model parts.

### THE ABAbuilder TOOL

Using traditional methods of model design introduces high risk of making errors; therefore the ABAbuilder tool which was created upon the existing architecture

brings a great improvement in this process. ABAbuilder is a CASE tool based on the ABAsim architecture and it supports several aspects of model development including system analysis, design of communication, and use of methodologies, prototyping, model maintenance and re-engineering (Figure 3).

One of the greatest contributions of the architecture is the flexibility of model configuration. ABAbuilder is capable of creating large palette of alternative internal components, which can be used to construct the simulation scenario. The experimenter is allowed to change executive characteristics of each of the system agents by using different components with pre-programmed behavior and decision making algorithms. Even less experienced programmers can design their own experiments. This is due to similarity of modeled system with the real world.

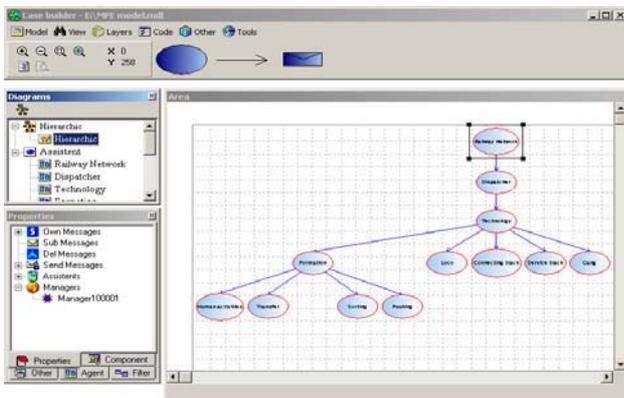


Figure 3: Screen Capture of the ABAbuilder Tool

A simulation model can be described by the hierarchy of system agents. From the ABAbuilder point of view, not only agents, but also their internal components are considered as model building base. There are several phases each model has to undergo, before experiments can be carried out:

1. design of the hierarchy and communication of system agents
2. implementation of the simulation model
3. verification of the model
4. parameters setup
5. source code generation

To be able to accomplish these particular phases the CASE tool had to be divided into four interacting modules: The *visual environment* displays the simulation model from different perspectives. Its main purpose is to display the agent hierarchy and internal components of each agent. It allows users add new

agents to the hierarchy, edit their internal components and define the message flow inside the model. Visual Petri nets editor is used to define the reaction of agent's manger to received messages. Users are allowed to look at all aspects of the hierarchy at once, or they can apply various sets of filters.

The *error localization* module works continuously and it checks the model each time a change has been applied. It provides user with two kinds of messages: *error messages* and *warnings*. Errors messages appear when user tries to do something which is not allowed by the methodology of creating models based on the ABAsim architecture. Warnings inform users of possible unwanted problem solutions.

When the decision is made by the user that the model is ready for deployment the *source code generation* module comes into use. This module is capable of creating the framework of the simulation model. It generates source files in chosen destination language (only Object Pascal is currently supported). These files contain all agent classes, their internal components, communication and the hierarchy. User then finalizes the simulation model in external development environment (e.g. Borland Delphi). Libraries containing the simulation kernel, which allow compiling a standalone application, are provided.

To be able to reconstruct the model from source files to its graphical representation a *source code parser* module is necessary. This module analyzes given files, which were previously generated and it provides data for the visual environment.

ABAbuilder tool enables to construct new models and modify existing ones with the help of prefabricated model parts, which are selected from bases of agent components (assistants, managers), agents, sub-models and their various configurations. It makes the whole process of modeling much easier to accomplish and more understandable for more as well as less experienced user.

## CONCLUSIONS

In this paper the agent-oriented architecture ABAsim was presented and its openness for creating flexible models of large-scale service systems. The general model of a service system presented in this paper is being customized to build more models of real world systems. Once these models are configured they are being verified and validated and experiments are being carried out. ABAbuilder tool makes process of model customization much easier for developers. One of the goals to achieve in the future is to try to make modeled systems cooperate by the methods of distributed simulation which are also supported by this architecture.

## REFERENCES

- Gulyas, L.; Koszik, T. 1999. "Model Design Interface – A case tool for the Multi-Agent Modeling Language", In: *Journal of Artificial Societies and Social Simulation* vol. 2, no. 3
- Jennings, N. R. 2001. "An agent-based approach for building complex software systems", *Communications of the ACM*, Vol. 44, No.4, pp.35-41
- Kavička, A.; Adamko, N.; Klima, V. 2000. "Simulation support for railway junctions infrastructure and processes economization", In: *Proceedings of MI 2000 conference*, Zilina, Slovak republic, 2000
- Klima, V.; Kavička, A. 2000. "Simulation support for railway infrastructure design and planning processes", In: *Proceedings of COMPRAIL 2000 conference*, Bologna, Italy, 2000, pp.447-456
- Kovacs, G.; Kopacsi, S.; Nacsa, J.; Haidegger, B.; Groumpos, P. 1999. "Application of software reuse and object-oriented methodologies for the modelling and control of manufacturing systems", In: *Computers in Industry 39 1999*, pp. 177-189
- Nwana, H. 1996. "Software Agents: An Overview", In: *Knowledge Engineering Review*, Vol. 11, No 3, pp.1-40
- Rebollo, M.; Julian, V.; Carrascosa, C.; Botti, V. 2000. "A Multi-Agent System for the Automation of a Port Container Terminal".
- Wooldridge, M.; Jennings, N. 1995. "Intelligent Agents: Theory and Practice", *The Knowledge Engineering Review 10 (2)*, pp. 115-152

## AUTHOR BIOGRAPHIES

**MICHAL LEKÝR** - got his Master degree at the University of Zilina in 2003, where he is now doing his PhD. study.

**NORBERT ADAMKO** - graduated at the University of Zilina in 1997, where he in the same year started PhD. study. Currently he is working as an assistant professor, teaching courses on Modeling and Simulation, Data Structures and Discrete Simulation.

## ACKNOWLEDGEMENT

This work has been supported by the Slovak grant foundation under grant No. 1/1049/04 "Agent Oriented Architecture of Service Systems Simulation Models"

# AGENT-BASED MODELING AND SIMULATION OF CYBER-WARFARE BETWEEN MALEFACTORS AND SECURITY AGENTS IN INTERNET

Igor Kotenko

St. Petersburg Institute for Informatics and Automation  
39, 14<sup>th</sup> Liniya, St. Petersburg, 199178, Russia  
E-mail: ivkote@iias.spb.su

## KEYWORDS

Agent-based Modeling and Simulation, Computer network attacks, Information Assurance.

## ABSTRACT

The paper considers an approach to modeling and simulation of cyber-wars in Internet between the teams of software agents. Each team is a community of agents cloned on various network hosts. The approach is considered by an example of modeling and simulation of "Distributed Denial of Service" (DDoS) attacks and protection against them. Agents of different teams compete to reach antagonistic intentions. Agents of the same team cooperate to realize joint intentions. The ontologies of DDoS-attacks and mechanisms of protection against them are described. The variants of agents' team structures, the mechanisms of their interaction and coordination, the specifications of hierarchy of action plans as well as the developed software prototypes are determined.

## 1. INTRODUCTION

The modern society greatly depends on various distributed computer systems, which have different possibilities and are characterized by high complexity. Vulnerabilities of these systems, permanently magnified quantity, variety and complexity of cyber-attacks and gravity of their consequences highlight urgent necessity for information assurance and survivability of computer systems. Especially it is fair in connection with integration of computer systems on the basis of open networks (such as the Internet), permanently modified and magnified, not having state boundaries, centralized control and uniform security policy. Hackers characterize the current state of counteraction of malefactors' systems to security systems as "a game of network cats and mice" (Nomad 2002).

Experienced malefactors realize sophisticated strategies of cyber-attacks. These strategies can include:

- (1) Information gathering about the computer system under attack, detecting its vulnerabilities and defense mechanisms;
- (2) Determining the ways of overcoming defense mechanisms (for example, by simulating these mechanisms);
- (3) Suppression, detour or deceit of protection components (for example, by using slow ("stretched" in

time) stealthy probes, separate coordinated operations (attacks) from several sources formed complex multiphase attack, etc.);

- (4) Getting access to resources, escalating privilege, and implementation of thread intended (violation of confidentiality, integrity, availability, etc.) using the vulnerabilities detected;

- (5) Covering tracks of malefactors' presence and creating back doors in order to use them later.

Protection mechanisms should support real-time fulfillment of the following operations:

- (1) Implementing the protection mechanisms appropriated to the security policy (including proactive intrusion prevention and attack blocking, misinformation, concealment, camouflage, etc.);

- (2) Vulnerability assessment, gathering data and analysis of the current status of the computer system defended;

- (3) Intrusion detection and prediction of the malefactors' intentions and actions;

- (4) Direct incident response, including deception of the malefactors, their decoy with the purpose of disclosure and more precise determining the malefactors' purposes, and reinforcement of critical protection mechanisms;

- (5) Elimination of intrusion consequences and detected vulnerabilities, adaptation of the information assurance system to the next intrusions.

Unfortunately, the existing theoretical base for information assurance in large-scale systems does not correspond to the indicated tendencies.

We think that the majority of problems in information assurance is caused by immaturity of logical foundations for construction of integrated adaptive security systems operating in adversarial environments (Ellison et al. 1997; Linger et al. 2001).

To our opinion, it is stipulated mainly by insufficient attention to fundamental works, which, on the one hand, consider information assurance as a complex task of organizational and technical cyber warfare between security systems and malefactors' attacking systems (Bell and Santos 2002; Carmel and Markovitch 1996; Garstka 2000; Geib and Goldman 2001; Goldman 2002; Modeling and Simulation Activities 1997), and, on the other hand, are based on *exploratory modeling and simulation* of indicated processes (Klein 2002; Stytz and Banks 2001; Yuill et al. 2000).

The issues of modeling and simulation of information assurance are actively researched during more than thirty years. The various formal and informal models of particular protection mechanisms are developed, but practically there are not enough works formalizing complex antagonistic character of information assurance.

This state of research is explained by complexity of information security problems (Evans et al. 2001; Faatz 2000; Waag et al. 2000).

Understanding of an information assurance as uniform holistic system is extremely hampered. It depends on great many interactions between different cyber warfare processes and is determined by dynamic character of these processes and different components of computer systems (Bell and Santos 2002).

Especially it is fair in conditions of the Internet evolution to a free decentralized distributed environment in which a huge number of cooperating and antagonistic software components (agents) interchange among themselves and with people by large information contents and services (Information Dynamics and Emergent Behavior 2001; Kephart et al. 1998). Modeling and simulation of these aspects of information assurance is supposed to put as a basis of our research. This will allow developing an integrated approach to construction of network security systems operating in aggressive antagonistic environment.

Information assurance modeling and simulation is considered in the paper as modeling and simulation of processes realized, at least, by two opposing sides, namely, malefactors' attacking and security systems trying to bypass, deceive and suppress each other. Such modeling and simulation can allow developing main principles of construction of intrusion-aware distributed network security systems, which operate by prediction of intentions and actions of malefactors.

One of the most harmful classes of attacks aiming at destruction of network resources availability is "Denial of Service" (DoS). The purpose of DoS is isolation of a victim host, i.e. creation of a situation in which a remote host cannot communicate with an external world. The basic feature of "Distributed Denial of Service" (DDoS) attacks is coordinated use of enormous remote hosts-"zombies" for generation of ill-intentioned traffic (Mirkovic et al 2002; Nouredien 2002; Jeon 2001). DDoS attacks are preceded by breakings of tens (and sometimes hundreds, and even thousand) computers in which the special DDoS-software is established thus allowing to carry out coordinated DoS attacks against victim hosts.

The goal of our research is development of mathematical basis for adaptive co-evolving agent-based adversarial modeling and simulation for cyber warfare. This paper considers an approach to agent-based modeling and simulation of cyber warfare by an example of modeling and simulation of DDoS attacks (Gorodetsky et al. 2003) and protection against them.

The rest of the paper is structured as follows. *Section 2* outlines suggested common approach for modeling and simulation of cyber warfare by imitating counteraction of malefactors and defense agents' teams. *Section 3* describes the ontology of DDoS attacks and defense mechanisms. *Section 4* presents specifications of structure and scheme of operation of DDoS agents' team. *Section 5* outlines structure and functioning of defense agents' team. *Section 6* depicts mechanisms for action coordination, monitoring and restoration of agent functionality, and maintenance of communication selectivity. *Section 7* describes the suggested formal model of computer network. *Section 8* determines architecture and main user interfaces for software prototypes developed. *Conclusion* outlines the results of the paper.

## 2. TEAMWORK-BASED APPROACH FOR MODELING AND SIMULATION

Agent-based modeling and simulation of information assurance assumes that cyber warfare is represented as a large collection of semi-autonomous interacting agents. The aggregate system behavior emerges from evolving local interactions of agents in a dynamically changing environment specified by computer network model.

We assume to select two agents' subsystems (teams) effecting on computer network as interconnected set of objects (resources):

(1) Adversary attacking system - a team of malefactor's agents (for automatic generation of distributed coordinated attacks);

(2) Security (defense) system - a team of security agents (for intrusion protection, data sensing and information fusion, intrusion detection, adversary intentions and actions prediction, and incident response).

So the task of agent-based modeling and simulation for information assurance is represented as modeling and simulation of the malefactor and security agents-teams' interaction including the security (defense) team response on adversary activity.

Thus, adversary and defense systems are represented as antagonistic teams of agents. The purpose of adversary team consists in defining vulnerabilities of computer network and defense system and implementation of security threats (confidentiality, integrity, availability, etc.) by executing distributed coordinated attacks. The purpose of defense team is protection of computer network and own components from attacks.

Agents of different teams compete to reach opposite intentions. Agents of the same team cooperate to achieve common intention (to fulfill attack on computer network or to defend the computer network). Competing agents must deal with probabilistic knowledge, risk management, deception, and opponent modeling.

The security agents protect the defended network from attacks, observe adversary's actions, fulfill information fusion, try to deceive adversary team, estimate enemy actions, status of the network and components of the

security system, predict malefactors' intentions and actions, react to enemy actions, restore resources of the network (Gorodetski et al. 2004).

It is said that the agents' team realizes teamwork, if the team members (agents) fulfill joint operations for reaching the common long-time goal in a dynamic external environment at presence of noise and counteraction of opponents. Now the research on teamwork is an area of steadfast attention in multi-agent systems. A set of approaches to formalization and simulation of the agents' teamwork is known (Cohen and Levesque 1991; Grosz and Kraus 1996; Jennings 1995; Tambe 1997; Tambe and Pynadath 2001; etc).

For implementing teamwork, we use the main ideas stated in works on the joint intention theory (Cohen and Levesque 1991), the shared plans theory (Grosz and Kraus 1996) and the combined theories of agents' teamwork (Jennings 1995; Tambe 1997; Tambe and Pynadath 2001; etc).

In our approach it is offered that the agents' teamwork is organized by the group (team) plan of the agents' actions. In result, a team has a mechanism of decision-making about who will execute particular operations. As in the joint intention theory, the basic elements, allowing the agents' team to fulfill a common task, are common (group) intentions, but its structuring is carried out in the same way as the plans are structured in the shared plans theory (Kotenko 2003). The common (group, individual) intention and commitment are associated with each node of a general hierarchical plan. These intention and commitment manage execution of a general plan, providing necessary flexibility. During functioning each agent should possess the group beliefs concerning other team-mates. For achievement of the common beliefs at formation and disbandment of the common intentions the agents should communicate. All agents' communications are managed by means of common commitments built in the common intentions. For this purpose it is supposed to use the special mechanism for reasoning of agents on communications. Besides it is supposed, that agents communicate only when there can be an inconsistency of their actions (Tambe 1997). It is important for reaction to unexpected changes of network environment, redistributing roles of the agents which failed or unable to execute the general plan, and also at occurrence of not planned actions. The mechanisms of the agents' interaction and coordination can be based on three groups of procedures (Tambe and Pynadath 2001):

- (1) Coordination of the agents' actions (for implementation of the coordinated initialization and termination of the common scenario actions);
- (2) Monitoring and restoring the agents' functionality;
- (3) Communication selectivity support (for choice of the most "useful" communication acts).

The specification of the plan hierarchy is carried out for each role. The following elements of the plan should be described: initial conditions, when the plan is offered for fulfillment; conditions for finishing the plan

execution (these conditions can be as follows: plan is fulfilled, plan is impracticable or plan is irrelevant); actions fulfilled at the team level as a part of the common plan. For the group plans it is necessary to express joint activity.

Assignment of roles and allocation of plans between the agents is fulfilled in two stages: at first the plan is arranged in terms of roles, and then the roles are put in correspondence to the agents. Agents' functionalities are generated automatically according to the roles specified.

The adversary (malefactors') team co-evolves by generation of new attack patterns to overcome defenses. On the other hand, defense team co-evolves by generating new protective actions against attacks, suppression of malefactors' team and recovery of destructed and compromised components of the computer network.

Interaction among agents can be represented as a two-player game ("game of network cats and mice"), where the agents' objective is to look for a strategy that maximizes their expected sum of rewards in the game.

We assume that agents' strategies can be modeled by means of the family of stochastic attribute formal grammars (and their interpretation by state machines) and hidden markov models.

To cope with the information heterogeneity and distribution of intrusion sources and agents used we apply ontology-based approach and special protocols for specification of shared consistent terminology.

The suggested technology for creation of the malefactors-agents' team (that is fair for other subject domains) consists in realization of the following chain of stages (Gorodetsky, et al. 2003; Kotenko 2003):

- (1) Formation of the subject domain ontology;
- (2) Determination of the agents' team structure;
- (3) Determination of agents' interaction-and-coordination mechanisms (including roles and scenarios for roles exchanges);
- (4) Specification of agents' plans as a hierarchy of stochastic formal grammars;
- (5) Assignment of roles and allocation of plans between agents;
- (6) State-machine based implementation of teamwork.

Modeling in any subject domain assumes development of its conceptual model, i.e. a set of basic concepts of the subject domain, relations between the concepts, and also data and algorithms interpreting these concepts and relations. So formation of the subject domain ontology is an initial stage of the agents' team creation.

The agents' team structure is described in terms of a hierarchy of group and individual roles. Leaves of the hierarchy correspond to the roles of individual agents, but intermediate nodes - to group roles. One agent can execute a set of roles. Agents can exchange roles in progress of plan execution.

For agents' team operation in real-time a hierarchy of state machines is used. These state machines are built as a result of interpretation of a hierarchy of attribute

stochastic formal grammars which set the plan hierarchy specification. The state machines implement a choice of the plan which will be executed and a fulfillment of the established sub-plans in a cycle “agents’ actions - environment responses”.

Agents’ coordination is carried out by message exchange. As the agents’ teams operate in antagonistic environment agents can fail. The lost functionalities are restored by redistributing the roles of failed agents between other agents and (or) cloning new agents.

detail. In this ontology, the hierarchy of nodes representing notions splits into two subsets according to the macro- and micro-layers of the domain specifications. All nodes of the ontology on the macro- and micro-levels of specification are divided into the intermediate (detailable) and terminal (non-detailable) (Gorodetski and Kotenko 2002).

The ontology notions of an upper layer can be interconnected with the corresponding notions of the lower layer through one of three kinds of relationships:

(1) “Part of” that is decomposition relationship (“Whole”-“Part”); (2) “Kind of” that is specialization relationship (“Notion”-“Particular kind of notion”); and (3) “Seq of” that is relationship specifying sequence of operation (“Whole operation” - “Sub-operation”).

High-layer notions (corresponding to the intentions) form the upper layers of the ontology. They are interconnected by the “Part of” relationship. Attack and defense actions realizing agent’s intentions (they presented at lower layers as compared with the intentions) are interconnected with the intentions by “Kind of” or “Seq of” relationship.

The “terminal” notions of the macro-level are further elaborated on the micro-level, and on this level they belong to the set of top-level notions detailed through the use of the three relationships introduced above.

In micro specifications of the ontology, besides three relations described (“Part of”, “Kind of”, “Seq of”), the relationship “Example of” is also used. It serves to establish the “type of object-specific sample of object” relationship.

The developed ontology includes the detailed description of the domain of DDoS attacks and defense mechanisms in which the notions of the bottom layer (“terminals”) can be specified in terms of network packets, OS calls, audit data, etc.

The fragment of the DDoS-attacks sub-ontology is depicted in Figure 1. The nodes specifying a set of software exploits for generation of DDoS attacks

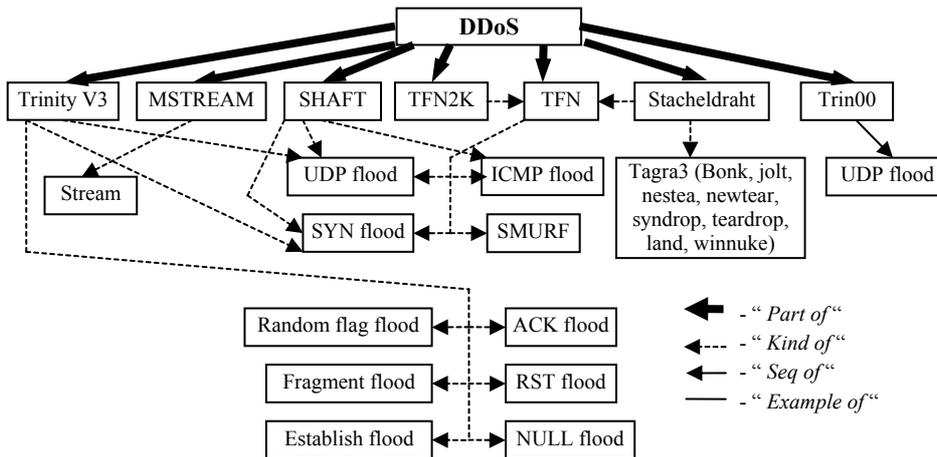


Figure 1: Fragment of DDoS Attacks Ontology on Macro-level

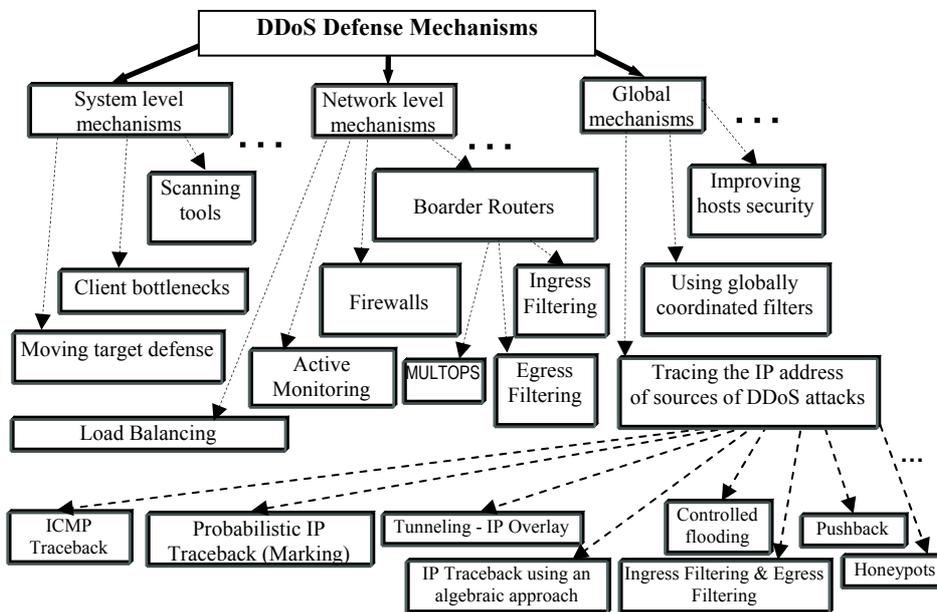


Figure 2: Fragment of DDoS Defense Ontology on Macro-level

### 3. ONTOLOGY OF ATTACKS AND DEFENSE MECHANISMS

The developed ontology comprises a hierarchy of notions specifying activities of team of malefactors and defense agents directed to implementation of DDoS attacks and protection from them in different layers of

(Trinity V3, MSTREAM, SHAFT, TFN2K, Stacheldraht, Trin00) make up a top level of the ontology fragment (Figure 1). At lower levels of the fragment different classes of DoS-attacks are detailed. For example, "Smurf" attacks consist in sending broadcasting ICMP ECHO inquiries on behalf of a victim host, therefore hosts accepted such broadcasting packages answer to the victim host, that results in essential capacity reduction of a communication channel and, in some cases, in full isolation of an attacked network. Other example is "Land" attack. It is sending an IP-packet with equal fields of port and address of the sender and the receiver, i.e. Source Address (SA) = Destination Address (DA), Source Port Number (SPN) = Destination Port Number (DPN).

The nodes determining the high levels of defense mechanisms (system, network, global) make up the top level of DDoS defense mechanisms ontology (Figure 2). At the bottom levels of the ontology these nodes are described by particular defense mechanisms (Mirkovic at al. 2002; Noureldien 2002).

Different types of nodes corresponding to system level defense mechanisms can be used. For example, scanning tools check presence of DDoS-agents in the host file system, and also scan the ports frequently used by attackers. Mechanisms of client bottlenecks are directed on creating bottleneck processes on the zombie hosts used for DDoS-attacks to limit their attacking ability. Mechanisms of moving target defense consist in changing IP address to avoid being attacked.

Mechanisms of a network level can be represented by the following nodes: (1) mechanisms for defeating DDoS attacks at boarder routers (Ingress Filtering, Egress Filtering, MULTOPS bandwidth attack detection, etc.); (2) firewalls (intended for eliminating the packets implementing DoS-attacks); (3) active monitoring (for continuous supervision on network state, checking TCP/IP traffic and reaction to critical situations); (4) load balancing (on the basis of configurable input and output queues on hosts).

Global mechanisms implement coordinated defense tools on different hosts in the Internet. For example, the goal of tracing the source IP address is to observe the intruders' path back to the zombie computers or the original attacker.

#### 4. ATTACK AGENTS' TEAM

We limit the *team of DDoS agents* by three-level structure. The team consists of the "client" supervising a subteam of "masters". Each master, in turn, manages a group of "demons". "Demons" execute immediate attack actions against victim hosts. "Demons" include two subsets – scouts  $D_R$  and attackers  $D_A$ .

So a set of agents can be described as a pair  $A = \{M, D\}$ , where  $M = \{m_1, m_2, \dots, m_r\}$  – set of "masters";  $D = \{D_R, D_A\} = \{d_1, d_2, \dots, d_n\}$  – set of "demons" ("scouts" and "attackers"). Each agent can be represented as follows (Georgeff and Rao 1995; Rao and Georgeff 1998):  $a_N = \langle K, B, De, I, P, G, C \rangle$ , where

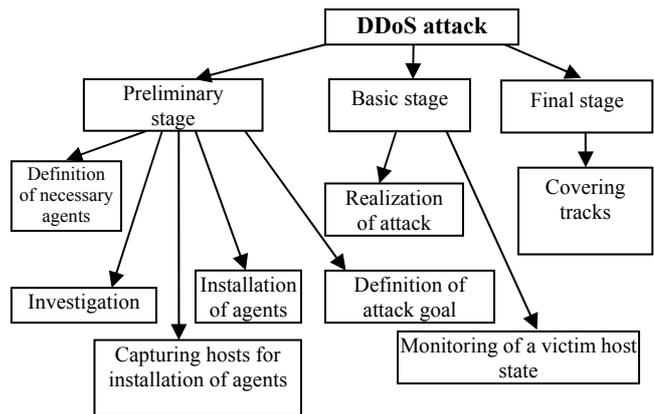


Figure 3: Upper level of DDoS Agents' Plans Hierarchy

$N$  – an identifier of the agent;  $K$  – knowledge;  $B$  – beliefs;  $De$  – desires;  $I$  – intentions;  $P$  – a set of parameters determining a mode of agent's operation, for example, minimal reaction time ( $RT$ ), etc.;  $G = \{L_R, f_R\}$  – a set of agent's goals and actions,  $L_R$  – hierarchy of possible goals and actions of the agent (reactions to influences),  $f_R$  – anytime-function for selecting a current goal or action from  $L_R$  according to  $K, B, De, I, P$  and  $C$  for time not exceeding  $RT$ ;  $C$  – commitments of the agent concerning other agents.

Common formal plan of attacks implemented by team of DDoS agents has three-level structure: (1) Upper level is a level of intention-based scenarios of malefactors' team specified in terms of sequences of intentions and negotiation acts; (2) Middle level is a level of intention-based scenarios of each malefactor specified in terms of ordered sequences of sub-goals; (3) Lower level is a level of malefactor's intention realization specified in terms of sequences of low-level actions (commands).

The upper level of hierarchy of DDoS agent plans is depicted in Figure 3. DDoS-attack includes three stages: preliminary, basic and final. Main operations of the preliminary stage are investigation (reconnaissance) and installation of agents. The content of the basic stage is realization of DDoS attack by joint actions of agents. Having received as a result of a chain of messages a "victim" address, agents-attackers begin to defeat a chosen host. At this time agents-scouts monitor a victim state. At detection of success of attack agents-scouts inform other agents about this fact. In case of prevarication of a host (for example the host been switched off from a network) or impossibility of defeating it, the operation is terminated or a new victim for DDoS attack is chosen.

The fragment including the upper and middle levels of hierarchy of agent plans for preliminary and basic stages is depicted in Figure 4. As an example, only two types of DDoS-attacks are represented – SMURF and Land attacks.

Mathematical model of attacks is specified in terms of a set of formal grammars interconnected through "substitution" operations (Kotenko and Man'kov 2003):

$M_A = \langle \{G_i\}, \{Su\} \rangle$ , where  $\{G_i\}$  – the formal grammars,  $\{Su\}$  – the “substitution” operations. The sequences of symbols (“strings”, “words” – in formal grammar terminology) generated by each of such grammars correspond to the sequences of time ordered malefactor’s intentions and/or actions. It is assumed that every sequence of a malefactor’s actions viewed as a “word” in a formal language is specified through a family of enclosed context-free grammars recognizable by a corresponding family of state machines. At the scenario specification layer (it was earlier called macro-layer) such sequences correspond to the specification of scenarios in terms of the malefactor’s intentions and actions.

The formal model of attack scenarios in terms of formal

the steps of an attack scenario),  $V_T$  is the set of its terminal symbols (that designate the steps of a lower-level attack scenario),  $S \in V_N$  is the grammar axiom (an initial symbol of an attack scenario),  $P$  is the set of productions (production rules) that specify the refinement operations for the attack scenario through the substitution of the symbols of an upper-level node by the symbols of the lower-level nodes, and  $A$  is the set of attributes and algorithms of their computation.

Attribute component of each grammar serves for several purposes. The first of them is to specify randomized choice of a production at the current inference step if several productions have the equal left part non-terminals coinciding with the active non-terminal in the current sequence under inference.

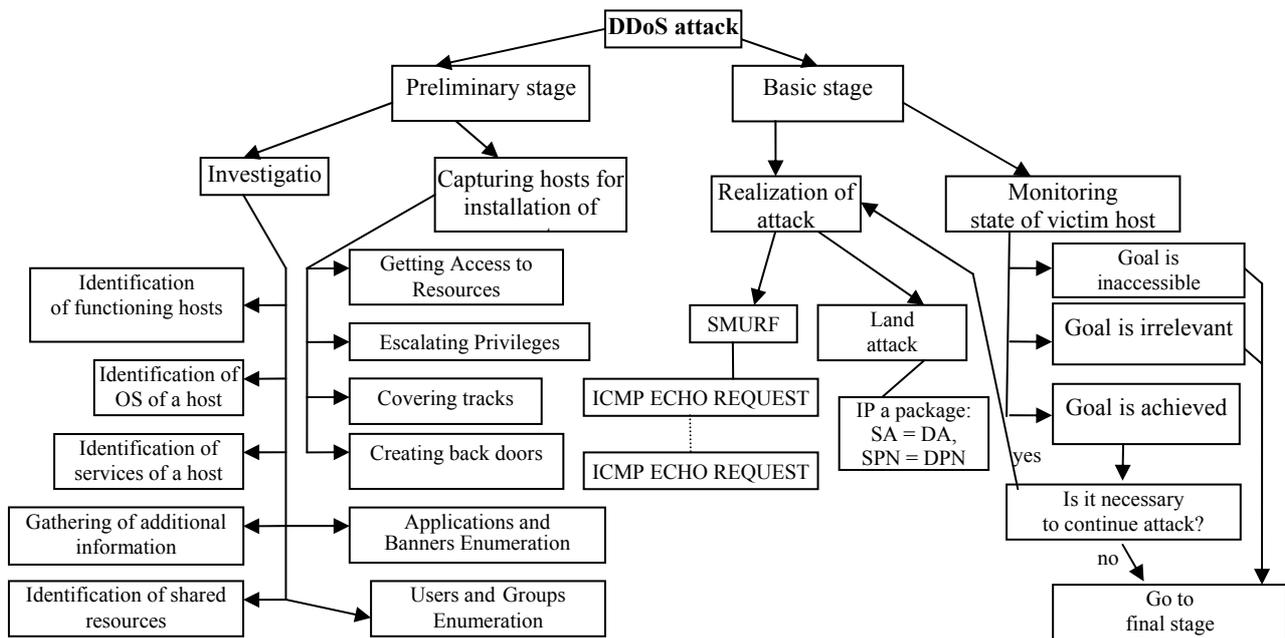


Figure 4: Fragment of Upper and Middle Levels of Hierarchy of DDoS Agents’ Plans

grammars are based on the attacks ontology described above. It is noteworthy to notice that each node of the ontology that is not “terminal” one is mapped to particular grammar, which is capable to generate only admissible sequences realizing this intention in terms of symbols, corresponding to the ontology nodes of the immediately lower layer. Depending on the required level of detail, these nodes may be represented by the terminal nodes of the macro or micro-level. In the former case, the grammar may be used to visualize the malefactor’s actions, and in the latter case – for attack simulation in the lowest layer terms (if the “terminal” nodes of the micro-level are represented by network packets, or messages of the tcpdump program, OS commands and/or calling applications with specified parameters).

Every formal grammar is specified by quintuple (Kotenko and Man’kov 2003):  $G = \langle V_N, V_T, S, P, A \rangle$ , where  $G$  is the grammar identifier (name),  $V_N$  is the set of non-terminal symbols (that are associated with the upper and the intermediate levels of representation of

Also the attribute component is used to check conditions determining the admissibility of using a production at the current step of inference. These conditions depend on (1) task specification (general attack goal), (2) configuration of the attacked computer network (host) and its resources and (3) results of the malefactor’s previous actions.

Assignment of roles and distribution of plans between agents are carried out as follows: roles of the agents necessary for the given goal are selected, the chosen roles are appointed to agents, agents of corresponding types are installed (cloned, employed).

## 5. DEFENSE AGENTS’ TEAM

*Defense agents* are installed on hosts placed in critical points of a network. The team of defense agents on a host can consist of one or several agents of various classes (Figure 5).

*Sensor agent (SA)* carries out preliminary processing of network traffic fixing events which are significant for

defense. *Identification and authentication agent (IAA)* is responsible for determining sources of messages and acknowledgement of their authenticity. *Access control agent (ACA)* regulates access of users to network resources according to their privileges. Agents *IAA* and *ACA* detect non-authorized actions on access to information resources of a host and interrupt connections attributed to number non-authorized. *Intrusion detection agents (IDA)* are responsible for detection of particular "suspicious" events or obvious attacks and decision-making concerning reaction to these events. *Intelligent intrusion detection agents (IIDA)* realize higher level of processing and generalization of facts discovered. They makes decision on the basis of messages about detected suspicious behavior and obvious attacks from different agents. *Reaction agents (RA)* are responsible for suppressing

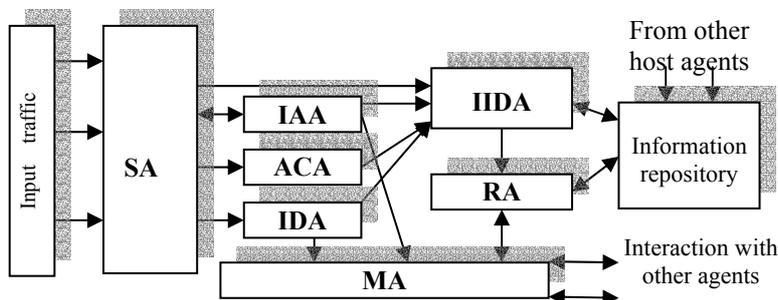


Figure 5: Structure of Defense Agents' Team

attacks. *Meta-agent (MA)* controls agents' behavior and detects complex distributed attacks which particular actions directed on different hosts.

The hierarchy of action plans of defense agents consists of three basic levels of processing:

- (1) Agents *SA* fulfill primary processing of input traffic;
- (2) Agents *IAA*, *ACA* and *IDA* carry out in real time the preliminary analysis of the received data detecting obvious attacks and reacting on them;
- (3) Agents *IIDA* and *MA* detect and react to the multiphase distributed attacks. These agents also implement forecasting the subsequent actions of users, using known scripts of attacks.

## 6. INTERACTION AND COORDINATION MECHANISMS

As it was described in section 2, it is offered to use three groups of mechanisms (procedures): maintenance of action coordination, monitoring and restoration of agent functionality, and supporting communication selectivity (Tambe 1997).

The mechanisms of first class are intended for realization of coordinated initialization and termination of actions of some general plan. Coordinated initialization means that all team-mates begin execution of the same plan at defined time. It assumes an appointment of specific agents for fixed roles in specific scenario, their notification about the appointed scenario

and role, and also reception of confirmations on their readiness to play the defined role in the scenario.

The coordinated termination of a common action (refusal of the common intention) demands also mutual informing of team-mates about the action at presence of corresponding conditions. Such conditions can be determined by achieving the common goal, finding-out of unattainability of the goal or its prevarication by even one member of the team.

For example, the attack goal "increase of authority up to a level of superuser" is achieved, if a malefactor managed to penetrate into a target host and to increase his authority up to a level of superuser. Also the purpose is unattainable, if one of obligatory actions on penetration into a target host is not executed. And the purpose is irrelevant, if the target host is switched off from the network.

Mechanisms of monitoring and restoration of agent functionality should provide supervision of some agents over others that it was possible to establish loss of capacity for work by the agent or a group of agents. It is directed on fast restoration of functionality of the team at the expense of reassignment of the "lost" roles to those team-mates which can perform corresponding job.

For example, if one of the malefactor-agents who are carrying out intention "Identification of operating system of a host" is blocked by firewall of a target network or other obstacle for realization of this intention takes place, this agent (or other malefactor-agent who found out state of nonoperability of "colleague") should send this information to a "leader" of the scenario. If there will be other agent, capable to solve the task this role should be assigned to it. Checking of rules and realization of reasoning should entail corresponding communications of agents by means of some communication protocol. Mechanisms of maintaining communication selectivity order the communication act when the probability and cost of agents' coordination loss is great enough. They are based on calculating importance of the message in view of the "costs" and benefits of this message. It is necessary to guarantee that the benefit of the message exchange for maintenance of agents' coordination surpasses a "cost" of the communication act (for example, a network security system, having intercepted agents' messages, can detect and "suppress" an attack). Therefore it is very important to choose those communication acts which will bring the greatest benefit to the team.

## 7. MODEL OF COMPUTER NETWORK

The attack development depends on the malefactor's "skill", information regarding network characteristics, which he/she possesses, some other malefactor's attributes (Gorodetski and Kotenko 2002), security

policy of the attacked network, a power of the defense agents' team, etc.

An attack is being developed as interactive process, in which the attacked network and security agents are reacting on the malefactor's action. Computer network plays the role of the environment for attacker and security agents, and therefore its model must be a part of the attack simulation tool.

Model of the attacked computer network is represented as quadruple  $MA = \langle M_{CN}, \{M_{Hi}\}, M_P, M_{HR} \rangle$ , where  $M_{CN}$  is the model of the computer network structure;  $\{M_{Hi}\}$  are the models of the host resources;  $M_P$  is the model of computation of the attack success probabilities;  $M_{HR}$  is the model of the host reaction in response of attack. Let us determine the model  $M_{CN}$  of a computer network structure  $CN$  as follows:  $M_{CN} = \langle A, P, N, C \rangle$ , where  $A$  is the network address;  $P$  is a family of protocols used (e.g., TCP/IP, FDDI, ATM, IPX, etc.);  $N$  is a set  $\{CN_i\}$  of sub-networks and/or a set  $\{H_i\}$  of hosts of the network  $CN$ ;  $C$  is a set of connections between the sub-networks (hosts) established as a mapping matrix. If  $N$  establishes a set of sub-networks  $\{CN_i\}$ , then each sub-network  $CN_i$  can in turn be specified by the model  $M_{CN_i}$  (if its structure needs to be developed in detail and if information is available about this structure). Each host  $H_i$  is determined as a pair  $M_{Hi} = \langle A, T \rangle$ , where  $A$  is the host address,  $T$  is a host type (e.g., firewall, host, etc.).

Models  $\{M_{Hi}\}$  of the network host resources serve for representing the host parameters that are important for attack simulation.

Success or failure of any attack action (corresponding to terminal level of the attack ontology) is determined by means of the model  $M_P$  of computation of the attack success probabilities. This model is specified as follows:  $M_P = \{R^{SPr_j}\}$ , where  $R^{SPr_j}$  is a special rule that determines the action success probability depending on the basic parameters of the host (attack target). The rule  $R^{SPr_j}$  includes IF and THEN parts. The IF part contains action name and precondition (values of attributes constraining the attack applicability). The THEN part contains value of success probability ( $SPr$ ).

The result of each attack action is determined according to the model  $M_{HR}$  of the host reaction. This model is determined as a set of rules of the host reaction:  $M_{HR} = \{R^{HR_j}\}$ ,  $R^{HR_j}: Input \rightarrow Output [\& Post-Condition]$ ; where  $Input$  – the malefactor's activity,  $Output$  – the host reaction,  $Post-Condition$  – a change of the host state,  $\&$  – logical operation "AND",  $[]$  – optional part of the rule. The Input format:  $\langle Attack\ name \rangle: \langle Input\ message \rangle: \langle Attack\ objects \rangle [; \langle Objects\ involved\ in\ the\ attack \rangle]$ . The Output format:  $\{\langle Attack\ success\ parameter\ S \rangle [; \langle Output\ message \rangle]; \{\langle Attack\ success\ parameter\ F \rangle [; \langle Output\ message \rangle]\}$ . The Attack Success Parameter is determined by the success probability of the attack that is associated with the host (attack target) depending on the implemented attack type. The values of attack success parameter are Success (S), and Failure (F). The part of output message

shown in the  $\langle \rangle$  is taken from the corresponding field of the host (target) parameters. The part of output message shown in quotation marks "" is displayed as a constant line. The Post-Condition format:  $\{p_1=P_1, p_2=P_2, \dots, p_n=P_n\}$ , where  $p_i - i^{th}$  parameter of the host (for instance,  $SP, SR, TH$ , etc.) which value has changed,  $P_i$  – the value of  $i^{th}$  parameter.

## 8. PROTOTYPES AND THEIR EVALUATION

Using Java, Visual C ++ and MASDK (Gorodetski et al. 2002) several prototypes of particular components of multi-agent system intended for simulation cyber-war of agents' teams in computer networks were developed. We have implemented three software components: a component for simulation of DDoS-agents' operation, a component of simulation of DoS-attacks and a component of intrusion detection and reaction to DDoS-attacks. Now these components are used for validation of the accepted basic ideas and formal framework. These components allow to show all stages of DDoS-attacks in the evident form, and also to simulate different situations depending on security parameters of attacked networks. The model of agents' team has three-level structure consisting of "client" managing "masters" which supervise "demons" (Figure 6). "Demons" are subdivided into two roles - scouts and attackers. Figure 7 illustrates capturing hosts and installing DDoS agents.

The visualization of particular DoS attacks is shown in Figure 8. The information presented in the figure is divided in four groups: (1) the attack task specification placed in the left top-most part of the screenshot; (2) the attack generation tree visualized in the right hand part of it; (3) the strings of the malefactor's actions placed in the left hand part of the screenshot and below the attack task specification; (4) a tag of success (failure) as green (black) quadrangle and data obtained from an attacked host (a host response) depicted on the right hand part of each information warrior's action.

The main objective of the experiments with the prototypes is to evaluate the tool's efficiency for simulation of different DDoS attacks and attacked network configurations. These experiments were carried out for various parameters of the attack task specification and an attacked computer network configuration.

The influence of the following input parameters on attacks efficacy was explored: a malefactor's intention, a degree of protection afforded by the network and personal firewall, a degree of security of attacked host, and the degree of malefactor's knowledge about a network.

The following parameters of attack realization outcome have been selected:  $NS$  (Number of attack Steps) – number of terminal level attack actions;  $PIR$  (Percentage of Intention Realization) – percentage of the hacker's intentions realized successfully (for "Reconnaissance" it is a percentage of objects about which the information has been received; for

“Implantation and threat realization” it is a percentage of successful realizations of the common attack goal on all runs); *PAR* (Percentage of Attack Realization) – percentage of “positive” messages (responses) of the Network Agent on attack actions (the “positive” messages are designated in attack visualization window by green lines); *PFB* (Percentage of Firewall Blocking) – percentage of attack actions blockage by firewall (red lines in attack visualization window); *PRA* (Percentage of Reply Absence) – percentage of “negative” messages (responses) of the Network Agent on attack actions (gray lines in attack visualization window).

Let consider two generated dependences of parameters *PIR*, *PAR*, *PFB*, *PRA* from different input parameters values under intention *GAR* (“Gaining Access to Resources”) realization. This intention is implemented on the stage “Capturing hosts for installation of agents”. For construction of these dependences the following values of the attacked network configuration were used as x-coordinate parameters: 1 – both network and personal firewalls are active; 2 – only network firewall is active; 3 – only personal firewall is active; 4 – none of firewalls is active.

The main parameters changes under maximal protection of attacked host (“Strong”) and maximal hacker’s knowledge about a network (“Good”) are depicted in Figure 9.

The main parameters changes under minimal protection of attacked host (“None”) and maximal hacker’s knowledge about a network (“Good”) are depicted in Figure 10.

The simulation-based exploration has demonstrated its efficacy for accomplishing various attack scenarios against networks with different structures and security policies implemented.

## 9. CONCLUSION

In the paper we developed basic ideas of the modeling and simulation of cyber warfare by teamwork approach and formal grammars. The technology for creation of the agents’ team was suggested and described. We

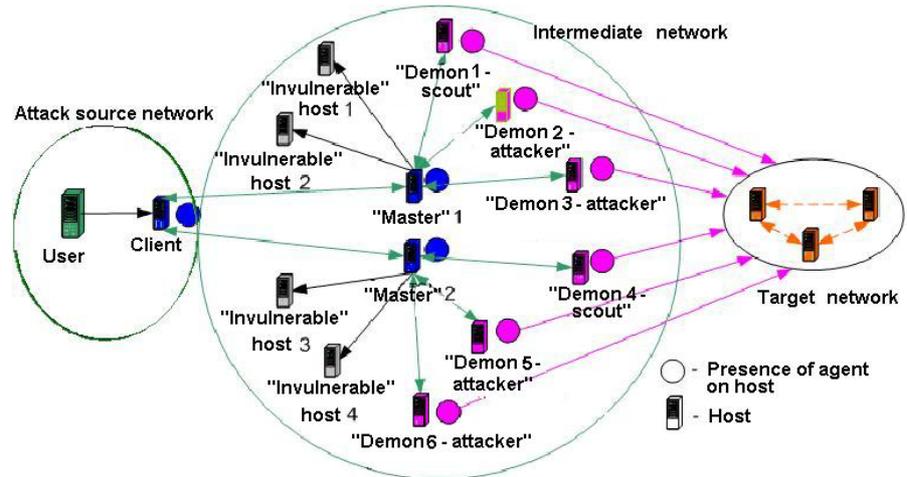


Figure 6: Diagram of DDoS Attacks Simulation

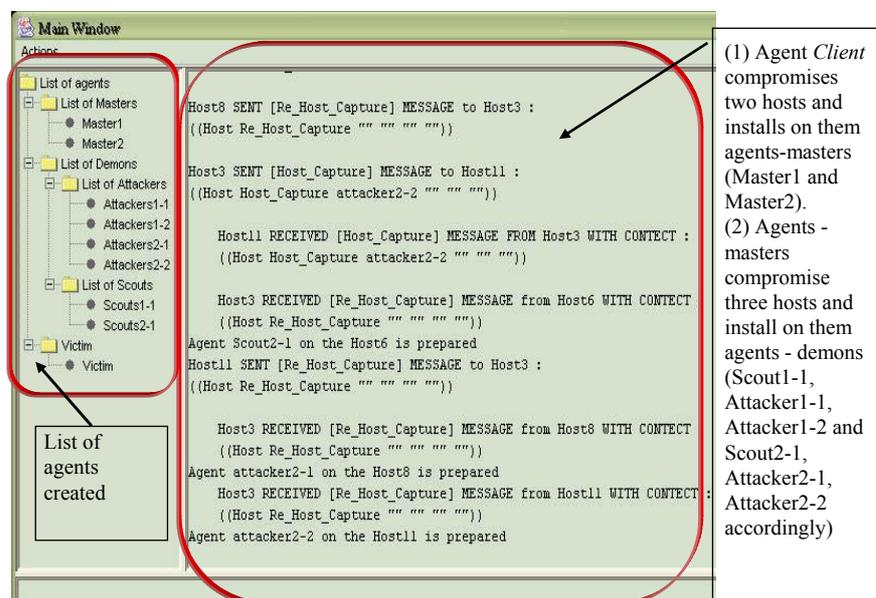


Figure 7: Capturing Hosts and Installing DDoS Agents

developed the approach to be used for conducting experiments to both evaluate computer network security and analyze the efficiency and effectiveness of security policy against different DDoS attacks.

We presented the structure of a team of agents, specifications of hierarchies of agent plans, agent interaction-and-coordination mechanisms, and agent role-assignment mechanisms.

Software prototypes were developed. They allow imitating a wide spectrum of real life DDoS attacks and defense mechanisms. Software code is written in terms of Visual C++ 6.0 and Java 2 languages. Experiments with the prototypes have been conducted, including the investigation of attack scenarios against networks with different structures and security policies.

The further development of our modeling and simulation framework and software tools will consist of joining different software components implemented, improving capabilities of the attack and defense agents

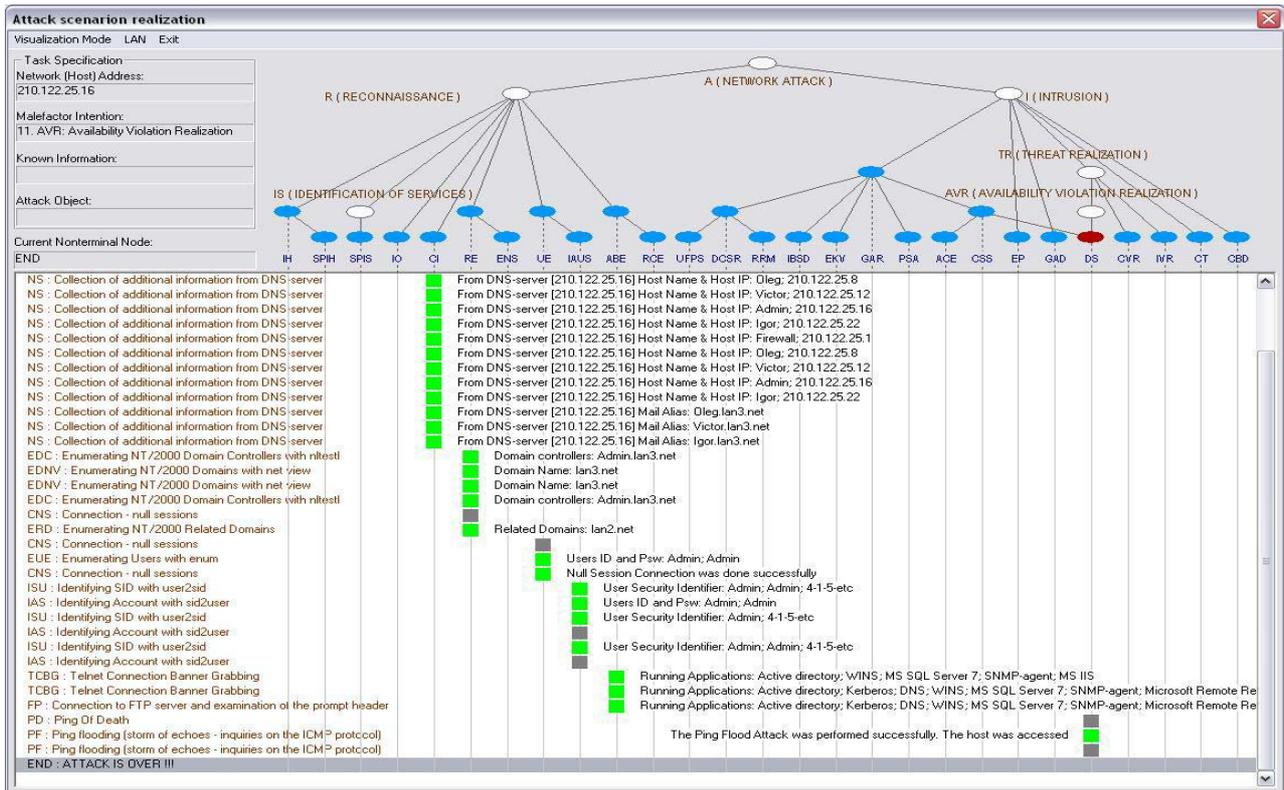


Figure 8: Visualization of DoS Attack Generation

teams by expansion of the attack and defense mechanisms classes, and implementing more sophisticated attack and defense scenarios.

## 10. ACKNOWLEDGEMENT

This research is being supported by grants 04-01-00167 of Russian Foundation of Basic Research and partly funded by the EC as part of the POSITIF project (contract IST-2002-002314).

## REFERENCES

- Bell, B. and E. Santos Jr. 2002. "Making Adversary Decision Modeling Tractable with Intent Inference and Information Fusion". *Proc. of the 11th Conf on Computer Generated Forces and Behavioral Representation*, Orlando FL.
- Carmel, D. and S. Markovitch. 1996. "Opponent modeling in multi-agent systems". *Adaption and Learning in Multi-Agent Systems*, Lecture Notes in Artificial Intelligence. Vol.1042.
- Cohen, P.R. and H.J. Levesque. 1991. "Teamwork". *Nous*, 25(4).
- Ellison, R.J.; D.A. Fisher; R.C. Linger; H.F. Lipson; T. Longstaff; N.R. Mead. 1997. *Survivable Network Systems: An Emerging Discipline*. Technical Report. November.
- Evans, S.; S.F. Bush; J. Hershey. 2001. "Information Assurance through Kolmogorov Complexity". *DARPA Information Survivability Conference and Exposition II*. Anaheim, California.
- Faatz, D. 2000. "Information System Security Assurance Evaluation". *Information Assurance Technology Symposium*.
- Garstka, J. 2000. *Network Centric Warfare: An Overview of Emerging Theory*. PHALANX.
- Geib, C.W. and R.P. Goldman. 2001. "Plan recognition in intrusion detection systems". *DARPA Information Survivability Conference and Exposition*, DARPA and the IEEE Computer Society.
- Georgeff, M.P. and A.S. Rao. 1995. "The semantics of intention maintenance for rational agents". *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI-95)*, Montreal, Canada.
- Goldman, R.P. 2002. "A Stochastic Model for Intrusions. Recent Advances in Intrusion Detection". *Fifth International Symposium. RAID 2002*. Proceedings. Lecture Notes in Computer Science, V.2516. Springer Verlag.
- Gorodetski, V.; O. Karsayev; I. Kotenko; A. Khabalov. 2002. "Software Development Kit for Multi-agent Systems Design and Implementation". Lecture Notes in Artificial Intelligence, Vol. 2296.
- Gorodetski, V. and I. Kotenko. 2002. "Attacks against Computer Network: Formal Grammar-based Framework and Simulation Tool". *Fifth International Symposium. RAID 2002*. Proceedings. Lecture Notes in Computer Science, V.2516. Springer Verlag.
- Gorodetskiy, V.I.; I.V. Kotenko; J.B. Michael. 2003. "Multi-agent Modeling and Simulation of Distributed Denial of Service Attacks on Computer Networks". *Third International Conference "NAVY AND SHIPBUILDING NOWADAYS"* (NSN'2003). Proceedings. St.Petersburg, Russia.
- Gorodetski, V.; O. Karsayev; I. Kotenko, V.Samoilov. 2004. "Multi-Agent Information Fusion: Methodology, Architecture and Software Tool for Learning of Object and Situation Assessment". *The 7th International*

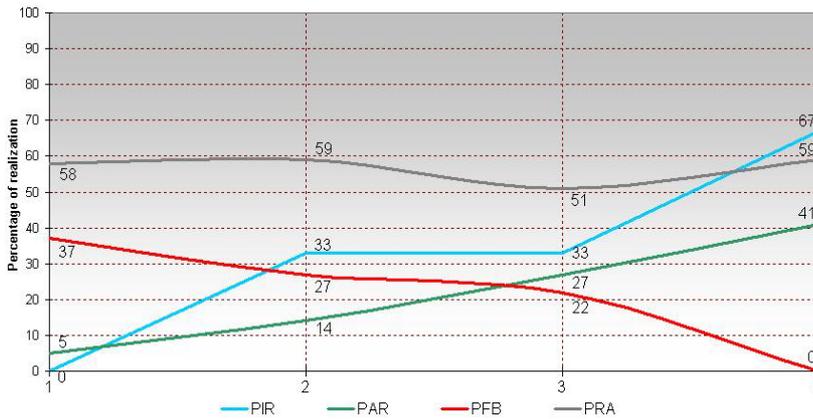


Figure 9: Changes of Simulation Parameters (1)



Figure 10: Changes of Simulation Parameters (2)

Conference on Information Fusion. Proceedings. Stockholm, Sweden. June 28 - July 1, P.346-353.

Grosz, B. and S. Kraus. 1996. "Collaborative plans for complex group actions". *Artificial Intelligence*, Vol.86.

Information Dynamics and Emergent Behavior. 2001. *Information Dynamics and Emergent Behavior of Heterogeneous-Agent Systems*. Arizona State University. DARPA ITO Sponsored Research. Project Summary.

Jennings, N. 1995. "Controlling cooperative problem solving in industrial multi-agent systems using joint intentions". *Artificial Intelligence*, No.75.

Jeon, DeokJo. 2001. *Understanding DDOS Attack, Tools and Free Anti-tools with Recommendation*. SANS Institute.

Kephart, J.O.; J.E. Hanson; J. Sairamesh. 1998. "Price-War Dynamics in a Free-Market Economy of Software Agents". *Proceedings of the Sixth International Conference on Artificial Life*. Editors MIT Press.

Kotenko, I. 2003. "Teamwork of Hackers-Agents: Modeling and Simulation of Coordinated Distributed Attacks on Computer Networks". *Lecture Notes in Artificial Intelligence*, Vol.2691.

Kotenko, I. and E. Man'kov. 2003. "Agent-Based Modeling and Simulation of Computer Network Attacks". *Fourth International Workshop "Agent-Based Simulation 4 (ABS 4)"*. Proceedings. Jean-Pierre Muller, Martina-M.Seidel (Editors). April 28-30. Montpellier. France.

Klein, G.L. 2002. *The Future for Intelligent Simulation Models*. [http://www.mitre.org/pubs/edge/january\\_02/klein.htm](http://www.mitre.org/pubs/edge/january_02/klein.htm)

Linger, R.C. and A.P. Moore. 2001. *Foundations for Survivable System Development: Service Traces, Intrusion Traces, and Evaluation Models*. Report CMU/SEI-2001-TR-029.

Mirkovic, J.; J. Martin; P. Reiher. 2002. *A Taxonomy of DDoS Attacks and DDoS Defense Mechanisms*, Technical report #020018. Computer Science Department, University of California, Los Angeles.

Modeling and Simulation Activities. 1997. *Modeling and Simulation Activities in Support of Information Assurance*. Report IATAC TR-97-002.

Nomad. 2002. *Nomad Mobile Research Centre*. [www.nmrc.org](http://www.nmrc.org).

Noureldien, A.N. 2002. "Protecting Web Servers from DoS/DDoS Flooding Attacks. A Technical Overview". *International Conference on Web-Management for International Organisations*. Proceedings, Geneva.

Rao, A. S. and M. Georgeff. 1998. Decision procedures of BDI logics. *Journal of Logic and Computation*, 8(3).

Stytz, M.R. and S.B. Banks. 2001. *Requirements and Issues in Cyberwarfare Simulation*. <http://www.sisostds.org/doclib/>.

Tambe, M. 1997. "Towards Flexible Teamwork". *Journal of Artificial Intelligence Research*, No.7.

Tambe, M. and D.V. Pynadath. 2001. "Towards Heterogeneous Agent Teams". *Lecture Notes in Artificial Intelligence*, Vol.2086.

Waag, G.L.; J.M. Feinberg; L.J. Painchaud; R.K. Heist. 2000. *Information Assurance Modeling & Simulation. State of the Art Report - A Summary*. Modeling and Simulation Information Analysis Center, Alexandria. VA.

Yuill, J.; F. Wu; J. Settle; F. Gong; R. Forno; M. Huang; J. Asbery. 2000. Intrusion-detection for incident-response, using a military battlefield-intelligence process. *Computer Networks*, 34.

## BIOGRAPHY



**IGOR KOTENKO** graduated with honors St.Petersburg Academy of Space Engineering (Department of Telecommunication and Computer-Aided Systems) and St.Petersburg Signal Academy (Department of Computer-Aided systems). He obtained the Ph.D. degree in 1990 and the National degree of Doctor of Engineering Science in 1999. He is Professor of computer science and Leading researcher of St. Petersburg Institute for Informatics and Automation. He is now leading a research group in the field of multi-agent modeling and simulation for computer network security. His e-mail address is [ivkote@iias.spb.su](mailto:ivkote@iias.spb.su) and his Web-page can be found at <http://space.iias.spb.su/ai/kotenko/kotenko.jsp>.

# COUPLING THE FARMING SYSTEM MODELLING TOOL 'OLYMPE' WITH THE MULTI-AGENT-SYSTEM SOFTWARE SYSTEM 'CORMAS' TO UNDERSTAND THE USE OF RESOURCES IN COMPLEX AGRICULTURAL SYSTEMS

Bruno Bonté

Ecole des Mines de Douai

Eric Penot

CIRAD-TERA

73 av JB Breton, 34 398

Montpellier Cedex France

Jean François Tourrand

CIRAD-TERA

73 av JB Breton, 34 398

Montpellier Cedex France

## KEYWORDS

Model, Management of Farming resources, Multi-Agent-System

## Abstract

This paper describes how the farming system modelling software 'Olympe' (developed in collaboration by INRA/CIRAD/IAMM) can be coupled with the agent-based simulation platform 'Cormas' to better characterize and analyse farming systems identified as major centres of decision in agriculture. Cormas (developed by CIRAD) enables representation of complex situations and takes into account interactions between different stakeholders. First we describe the two models and the potential benefits of combining the tools, then we describe how we developed a new platform based on the two software systems, and finally we illustrate how the new platform can be used with a simple example developed for educational purposes. The advantages and limits of this type of approach are discussed along with possible further developments.

## INTRODUCTION

Models can never be completely objective since different models can be used to describe the same phenomena from a different point of view. However it is sometimes useful to combine different types of models representing different stakeholders or different scales to better understand a phenomenon as a whole. To this end, we evaluated the advantages of combining two kinds of representation of the exploitation of agricultural resources: a « systemic representation of farming systems » and a « multi-agent model of resource management ».

The goal of this paper is to propose coupling the farming system modelling system 'Olympe' and the agent-based simulation platform 'Cormas'. Olympe (developed in collaboration by INRA/CIRAD/IAMM)

to characterize and analyse farming systems identified as major centres of decision in agriculture. Cormas (developed by CIRAD) enables representation of complex situations taking into account interactions between various stakeholders.

The objective of the study was to couple accurate economic information from the farming system modelling tool Olympe with a MAS to obtain a better definition of the 'farmer' agent as well as to profit from a potential data base created under Olympe which includes information on various types of farming systems. However the two software systems are written in different languages using different formats. A link was created for educational purposes to show the advantages of coupling the two tools, i.e. to better explain complex systems and the impact of technical changes in situations where the farmer is one of the major stakeholders and whose needs consequently have to be analysed from an economic perspective. MSA allows analysis of interactions and this is not possible with Olympe. The aims of the two software systems may be different, but in fact they are complementary.

This initial study paves the way for further work in coupling both tools at the different levels of complexity required for the analysis of different agrarian situations.

We first describe the two models and highlight the benefits of combining the tools. We then describe how we developed the new joint platform, which we present using a simple educational example. Finally, we discuss the advantages and limits of such an approach and identify possible future developments.

## THE OBJECTIVE OF THE STUDY

### Farming Systems Modelling

#### *The model, its origin and its working principle*

This model is frequently used by agro-economists and is based on the systemic representation of farming systems. It represents farming activities as a set of specific and clearly characterized systems (*figure 1*). For each techno-economic scale, a different system is defined in which sub-systems evolve. The model

defines a certain number of notions which are used to study farming systems from a technical and economic point of view.

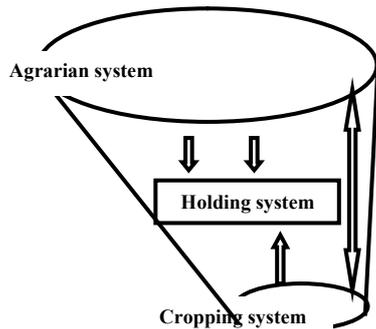


Figure 1: systemic representation of farming systems

This model is used as a tool to support decision making processes and communication and provides an economic synopsis of the complexity of farming systems. Each kind of system is defined on the basis of the ‘general system’ proposed by J.L. Le Moigne (Le Moigne 1990), by its function, its context, the transformations it generates and its finality. P. Jouve (1992), following R. Badouin (1985), provided a general framework for the diagnosis and analysis of farming systems. Indeed, each kind of system plays a specific role in production, conservation or in other activities which have to be integrated in the organization of the whole farm.

*The ‘Olympe’ software system and its uses*

Olympe software focuses on one particular stakeholder: the farmer and on his holding system which manages production factors (land, labour and capital as well as information) through activities grouped in cropping and livestock systems (figure 2). It can be used as a decision support tool by a range of different stakeholders: by farmers to manage their holding, by developers to understand the impact of agricultural policies, and by researchers to understand farmers’ strategies and pathways. Olympe can also be used by groups of farmers for regional analysis to measure flows (of inputs/outputs as well as capital and externalities). Primary data are obtained through classical surveys of farming systems. Each ‘entity’ is pre-defined in the software and the user needs only change parameters and enter data to build his own model. There are no internal regulations or particular rules generated by the software with conditionality.

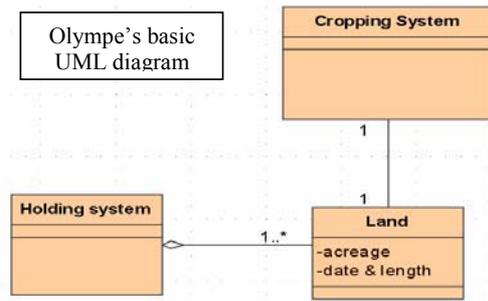


Figure 2: simplified diagram of Olympe

The different types of systems illustrated above were selected as being significant in the farm’s present situation. This allows the study of characteristic values which, in the context of farming management, are relevant at different scales. The farmer is considered to be the main decision-making agent. Technical innovation consists in modifying cropping practices (with associated costs and benefits) or replacing one cropping/livestock system by another. Thus, we need to understand the strategies the farmer uses to link determining economic factors (investments, cash flow, funding capacity etc.) with the optimal combination of cropping/livestock systems, taking into account other sources of income such as hunting, fishing, handicrafts and non-agricultural activities; a diversity which characterizes many farmers. The interplay between the elements that make up a given farming system is called a pathway.

**Multi-agent modelling of resources management: a simple way to deal with complex systems.**

*Framework, origin and working principle of the model*  
Multi-agent modelling means complex systems can be represented as a set of individual agents which interact in a particular environment and this representation results in the emergence of the picture of the whole system (Ferber 1995). Each ‘agent’ takes his own decisions based on his own goals and his own representation of the environment (figure 3). This kind of modelling can be adapted to represent social organisation and that is why it is currently being used by the Cirad to tackle problems inherent in resources management in a participative modelling approach.

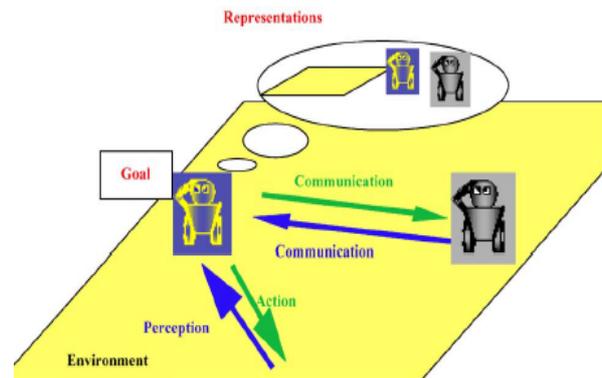


Figure 3: multi-agent system (Ferber 1999)

This type of representation enables us to study the consequences a given change will have for the system as a whole, whether the change concerns the rules of cooperation, the perception an agent has of his environment, or a different quantity of resources.

#### The 'Cormas' model and its use

Cormas is an agent-based simulation platform which was created to be used in a companion-modelling approach. Cormas is a computer tool which uses object programming to define each entity of the model as belonging to a 'class' which is defined by the modeller. In fact, certain basic classes (an 'agent', the active entity, 'a spatial entity', the elementary part of the topological back-up, and 'a passive entity') are pre-defined with basic characteristics and functions, and the user can create his own classes from these pre-defined entities to build his own model. He can for example define a 'farmer' as a 'located agent' whom he will link with an object 'farm' which he will already have defined (figure 4). It is then possible to study the influence specific definitions (definitions of interactions between agents for example) have on future developments.

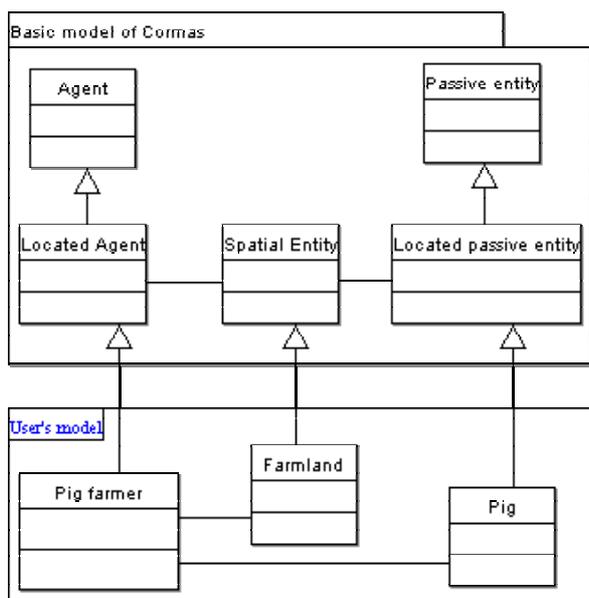


Figure 4: Diagram of a basic Cormas model

#### Complementarities of the two approaches

The advantage of combining the two approaches is the ability to represent, simulate and understand the roles of the different stakeholders and their strategies and, reciprocally, to see what influence the farmers' strategies have on the system as a function of the original situation and of the definitions of the rest of the system.

To this end, we used a multi-agent modelling approach (Cormas) to represent interactions, and a systemic representation of the holding (Olympe) to have a

suitable basis for the definition of individual strategies (figure 5).

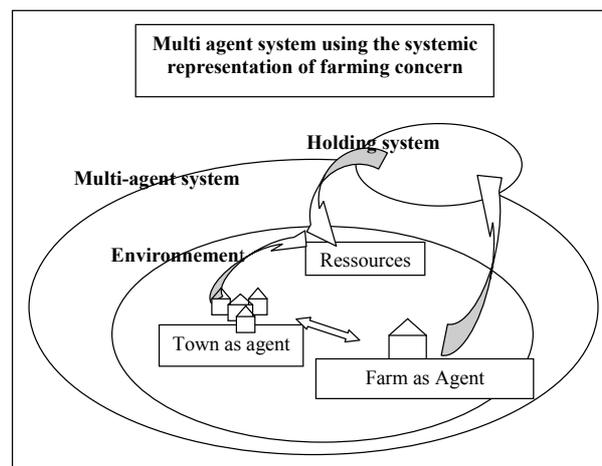


Figure 5: Coupling the models Farm as Agent should be Farm as agent

One of the main benefits in this work is to bring together people with different vocations to work on the creation of the same model. Agent-based models are often used to represent and analyse social problems at the regional scale, for example the impact pollution management or forest depletion will have on a limited resource, such as water or land. Those who create and use these models are often specialists in the problem concerned. In agriculture for example, using a systemic representation with a reasonably well defined unit enables an economic assessment that is relatively easy to process thereby enabling agro-economists to study the decision-making process in detail. Prospective analysis using Olympe provides scenarios that enable exploration of potentialities in partnership with farmers or with any other stakeholders involved in agriculture. Thus, knowledge accumulated over several years enables validation of hypotheses without losing touch with reality.

As Olympe is mono-agent with no interactions between agents, analysis of interactions, particularly with other agents, is consequently impossible. Linking Olympe with a MAS platform thus has two objectives: to obtain a better definition of the 'farmer' entity in the MAS (obtaining the economic information on farmers' behaviours from the Olympe data base) while MAS allows analysis of interactions firstly between farmers and secondly between farmers and other agents (Government, traders, policy makers, developers, project managers, etc.). In other words, Olympe feeds the SMA with information about one particular but crucial, agent: the farmer.

#### METHODOLOGY USED TO COUPLE THE TWO SOFTWARE SYSTEMS

##### Objectives and technical choices

After having defined the entities involved (i.e. the level of analysis, local/farming systems and regional/MAS) and the philosophy behind the approach and use of the

two software systems, we define the main functions we would like to integrate to ensure an effective and useful link. The objective is to assure the correct transfer of data (and the relevant data) from Olympe to Cormas to calibrate farms and activities related to agriculture with either primary data collected in original surveys, or ‘typical values’ that are representative of ‘standard’ farming systems. Olympe is a very efficient tool to characterize from an economic viewpoint a range of different farms from smallholdings to large estates with varying degrees of precision depending on the primary objective of the user. The number of Olympe users is currently expanding and thus providing feedback from a very interesting and pragmatic ‘network of users’ who share their data sets and experience. This should result in an impressive data base of farming systems all over the world. The main advantage is that comparisons are possible because users can share the same tools, the same simple but efficient economic indicators and consequently the same language and set of definitions. Such data can be used to calibrate a multi-agent model. Finally, it would be useful to create a link to feed data back to Olympe after it has been used in Cormas; this would give Olympe users the opportunity to react. Although a dynamic link of this type would be very useful it requires compatibility between the two software systems (written in different languages) and may thus be difficult to implement.

So far our objective has been to create a modelling framework with data from Olympe feeding the ‘farmer’ agent in Cormas, which allows users to transfer all management indicators of farming systems under Olympe to a multi-agent model through the definition of a proper Olympe data import protocol in the multi-agent model.

We chose to use the Cormas platform for our simulations. Specific classes were defined to respect Olympe’s representation of farming systems and to accept data originally formatted in Olympe. An XML file with the required structure is generated by Olympe as an output and loaded by Cormas at the beginning of the simulation. In other words, the Olympe conceptual model is introduced into Cormas’ world.

### Olympe’s conceptual model in Cormas’ world

A more detailed technical description of the link is needed here in order to understand the issues involved and consequently the limits of this work. Olympe will be used here only to model and not to simulate. Data are only generated by Olympe concerning the ‘farmer’ agent while Cormas generates the other agents in its own format. The simulation is implemented under Cormas. The data obtained and transferred with Olympe are the following:

- Quantity of products (inputs/outputs) and costs/benefits

- The structure of the cropping and livestock systems with 4 different formats for crops: annual crops, multi-annual crop (from 1 to 5 years), perennial crops (more than 5 years) and ‘animals’ (this is a potentially very complex livestock management tool to characterise all types of livestock systems). A cropping system is defined by the economic component of the ‘technical pathway’ (which is not technically very detailed) including lifespan, costs/benefits and labour requirements.

- Characteristics of the global farming system (structural costs, financial costs and other sorts of benefits and expenses).

This type of conceptual organisation is illustrated in the diagram of Olympe UML classes (figure 6).

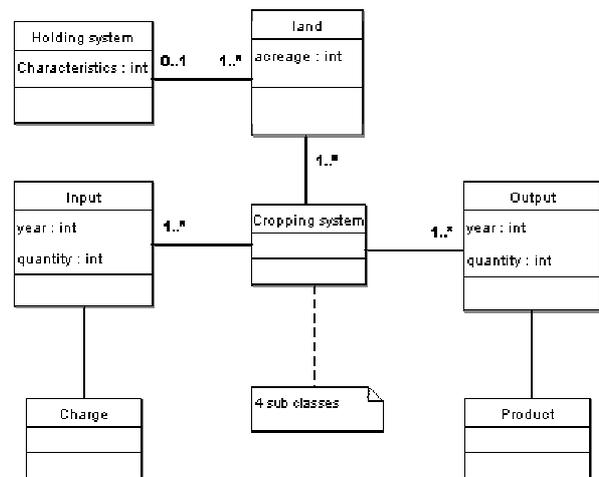


Figure 6: diagram of Olympe classes

It is important to mention that in such a framework, Olympe will only provide the economic results and the technical functioning of each sub-system at the farm level.

As the centre of decision-making in the systemic representation is defined at farm level, we chose to identify the ‘farmer’ agent and his family (livelihood) with the Farming system which would be a sub-class of ‘Located agent’ in MSA. The farmer owns some land which is cropped according to a ‘Kind of activities’ class. ‘Land’ is a sub-class of spatial entity and ‘Kind of activities’ a sub-class of passive entity (see figure 7). This ‘kind of activity’ will have all the characteristics of a cropping or livestock system defined using Olympe. Costs (Charges and Products) and kinds of production (kinds of activities) are defined using Olympe before beginning the simulation under Cormas and are not modified after transfer to Cormas; in other words, up to now, this is a one-way process. Agents (stakeholders) can only choose to farm their land by choosing the kind of activity they prefer from the list of available activities (‘kind of activities’) as a function of whatever available parameter the user wants.

## EXAMPLE OF A SIMULATION CARRIED OUT WITH OUR PLATFORM

### Description of the example

This example was developed for educational purposes to show the potential advantages of such a combination and as a method to implement it rather than to build a real case-study. MAS is used to represent interactions between farmers with behaviours defined according to the Olympe economic situation. The advantage lies in the definition of possible interactions and the impact of the results of the simulation on individual strategies.

The example is based on a real situation observed in the province of West Kalimantan (Borneo) in Indonesia, where rubber farmers have diversified with oil palm and other activities and also integrated new cropping methods and improved agroforestry practices (Leconte 2000). The situation described is typical of post-pioneer areas where diversification is used as a strategy to limit potential risks following a period of monoculture introduced during the first phase in pioneer zones when farmers relied on only one crop (in our case the jungle rubber system, an extensive agroforestry rubber based system), (Penot 2001). The same type of example could be developed for other typical pioneer areas (Amazon, Cambodia, West Africa, etc.). The characterisation of farming systems, and monitoring between 1997 and 2002 was carried out with emphasis on identifying farmers' strategies.

In our example, interactions correspond to knowledge transfer, and individual strategies are limited to the choice of which crops to rotate. (Table 1)

Table 1: interactions and strategies

#### ***Different sorts of "knowledge transfer": two types of transfer were selected for this example:***

- "Theoretical learning": farmers know how to implement technological recommendations or adopted recommendations two years after having been introduced to the new system by relatives. Once a farmer has acquired the know-how, he begins to teach it to his relatives.

- "Observation as learning process": a farmer acquires the know-how because one of his relatives has already implemented the new cropping system on his own farm.

#### ***Different individual strategies:***

- The "careful strategy": a farmer implements a new cropping system only if he can afford to pay for the plantation and to cover the costs of establishment.

- The "innovative strategy": a farmer systemically implements a new cropping system as soon as he thinks that it would be more profitable than other crops, particularly annual crops, even if he will need credit. In other words, he takes at least some financial risks.

Different situations are simulated. Based on his own financial situation, which is well defined under Olympe, each farmer can choose his level of investment in a new cropping system. Multi-agent modelling allows representation of the transfer of knowledge, its evolution, and the geographical aspect of the problem (diffusion) in the form of a simulated map.

Depending on the initial situation, the rules that govern interactions and consequently farmers' strategies can influence the simulation and its results. The purpose of this example is only to illustrate that the two different representations of the farmer as an agent in Olympe and in a MAS can be combined and are useful to study farmers' behaviour from different viewpoints, as well as to show how simulations leading to scenarios can be built. This example is not intended to be representative since neither the data nor the rules have been calibrated or tested with respect to the sensitivity of the initial parameters.

### Description of a simulation example

We simulated a situation in which we combined "observation as a learning process" and "Careful strategy". In this simulation there only are 3 different cropping systems:

- i) 'Upland rice', which is not very profitable but ensures food security;
- ii) "Jungle Rubber" (a classical extensive rubber agroforestry system) which is more profitable but only 15 years after it was planted (long immature period); and
- iii) RAS (improved Rubber Agroforestry Systems, developed by the SRAP project in partnership with farmers through on-farm experimentation) which are highly profitable with upland rice intercropping during the first years but require capital for implementation.

#### *Initialisation of the simulation*

20 farming systems were generated and placed on the topological support. Of these 20 farms, 4 were chosen as having the know-how to set up the new rubber cropping system (RAS). Others can only grow the classical upland rice crop and Jungle Rubber. Each farm received initial funds and a random set of the two cropping systems on 9 plots. In this example, jungle rubber crops were considered to be mature at the beginning of the simulation. Each farm is linked to two others (the nearest smallholding not too far away on the map) who are considered as "relatives".

#### *Description of each step of the simulation*

Each farmer undertakes the following activities:

- Crop his plots according to the chosen cropping systems

- Chooses the best cropping system he can afford based on his financial assets and the productivity of the system

- Learns how to set up the new cropping system if one of his relatives has already set it up.

*Observations*

Figure 8 is a simulated map of the situation at a key point in the simulation. Details of this particular simulation are presented in table 2.

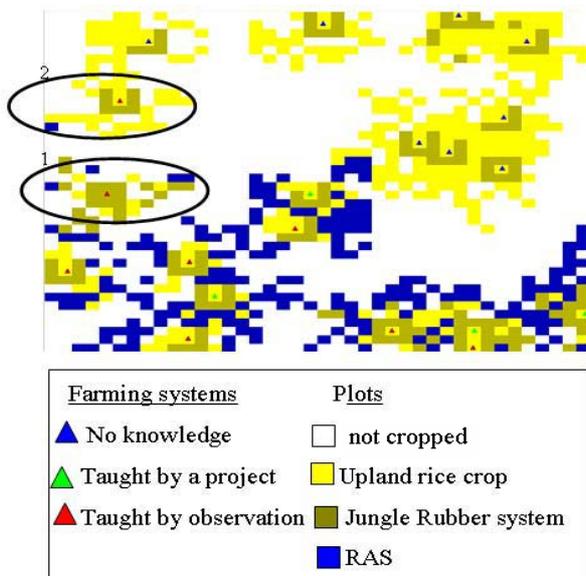


Figure 8: situation after 27 years of simulation



Figure 9: Situation after 56 years

The map in figure 8 shows the situation at the 27<sup>th</sup> step of the simulation. After about 30 years of simulation, it will be noted that the farmers who developed RAS are all located in the southern part of the map, due to the fact that the 4 smallholdings which had the knowledge at the beginning of the simulation were all located in that area and that the transfer of knowledge is linked to geographical proximity.

At the very beginning of the simulation, almost every farmer in the south had already implemented the new cropping system. However, the farmer in circle n° 1 on the map had only upland rice crops at the time of initialisation. At the beginning of the simulation, he began to plant “Jungle rubber” but he soon lacked funds to invest in other cropping systems. So by the time he had learned how to implement RAS, he could not afford to and now has to wait for his ‘Jungle rubber’ to be productive.

Due to their geographical location, all the farmers in the north except farmer n° 2 are all linked as relatives. Farmer n°2 must wait for farmer n° 1 to establish his first RAS to learn how to set it up himself. He has to wait 30 years in this simulation. Only then will northern farmers be able to access the information they need.

If we had chosen the “innovative strategy”, farmer n° 1 would have planted RAS as soon as he had learned how to proceed and the whole northern area would have had access to information much earlier. Similarly, “theoretical learning” would have allowed the northern area to establish RAS before farmer n° 1.

Figure 9 shows the situation after 56 years.

Of course this example is very simple but nevertheless shows how Olympe data can be used to feed the “farmer” agent in a MAS.

Table 2: details of the simulation

**LIMITS AND FUTURE OUTLOOK**

**Sensitivity studies**

An important point concerning simulations done with this new platform (Cormas & Olympe) is that there are a lot of parameters due to the precision of the information concerning the cropping systems generated by Olympe. It is consequently necessary to check the stability of the model and the validity of the values obtained by the MAS.

**Multidisciplinary studies**

Based on the structure of the data used by this type of platform, transfer of data between Cormas and Olympe (feedback) is possible but has not yet been implemented due to lack of time. Such a link would enable us to look at any farming systems at any time during the simulation and to check if the choices made by agents are rational. Rules governing farmers’

decisions can be defined very precisely since the economic information provided by Olympe can be very precise. But validation should be carefully implemented as the degree of complexity increases with an increase in the number of links to other agents.

## CONCLUSION

The objective of the study was to couple accurate economic information from a farming system modelling tool (Olympe) into a MAS in order to obtain a better definition of the 'farmer' agent as well as to profit from a data base on different types of farming systems created under Olympe. However the two software systems are written in different languages with different formats. We developed a link for educational purposes to demonstrate the advantages of coupling the two tools to better explain complex systems and to understand the potential impact of technical change in situations where the farmer is a major stakeholder and the economic impact of change needs to be defined with precision. MSA allows analysis of interactions that is not possible with Olympe. The two software systems appear to be very complementary despite their different objectives.

This initial study paves the way for further work in coupling the two tools to respect the degree of complexity required by a given agrarian situation.

## REFERENCES

- Le Moigne J.L. 1990. « La modélisation des systèmes complexes », Edition BORDAS, Paris
- Ferber J. 1995. « Les systèmes multi-agents », InterEditions, Paris
- Leconte J. 2000. « Modélisation des exploitations agricoles de Kalimantan », mémoire de stage CIRAD-TERA (contact E. Penot)
- Jouve P. 1992. « Le diagnostic du milieu rural à la parcelle » ou « Approche systémique des modes d'exploitation du milieu ». CNEARC Montpellier. 40 p.
- Courbet P. « Systèmes de production hévicoles et innovations techniques ». Le cas des agroforêts à hévéa de Kalimantan Ouest, Indonésie. Mémoire de fin d'étude en agronomie tropicale, ENGREF/ CNEARC, Montpellier, 86p.
- Penot E. 2001. « Stratégies paysannes et évolution des savoirs : l'hévéaculture agro-forestière indonésienne ». Thèse présentée pour obtenir le grade de Docteur de l'Université Montpellier I, 364 p.
- Ropella G.E., Railsback F.S. and Jackson S.K. 2002. "Software engineering considerations for individual-based models" *Natural resource modelling* Volume 15, Number 1, spring 2002

## DETAILS ABOUT THE AUTHORS

BRUNO BONTE was born in Paris, France and studied production management and computer science at the "Ecole des Mines de Douai". He took part in this study as a trainee for CIRAD-TERA. He is now doing a new internship with CIRAD until October 2005. E-mail address: [b.bonte@alem.tid.asso.fr](mailto:b.bonte@alem.tid.asso.fr).

Eric Penot is a senior agro-economist who has considerable experience on perennial crops in humid tropical regions, especially on rubber-based agroforestry systems in Indonesia. As a specialist of farming systems economics, he has been contributing to the development of the software "Olympe" in the INRA/CIRAD/IAMM team since 1999.

Jean Francois Tourrand is a livestock specialist whose main experience is in West Africa, and for the last 10 years, in the Amazonian Basin. He is currently leading a team of Brazilian and French scientists studying the evolution of pioneer areas in Amazonia, and is currently developing a MAS.

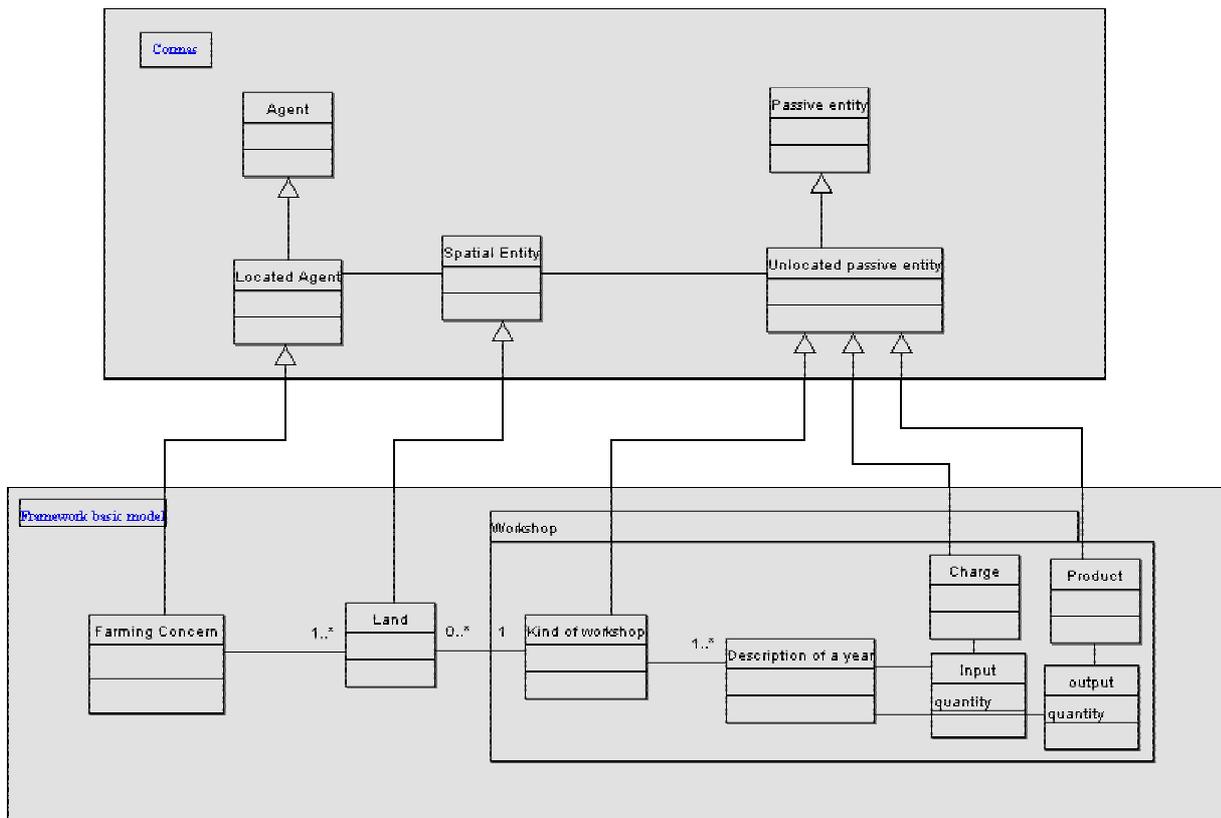


Figure 7: diagram of our platform

# MULTI AGENT SIMULATION IN INFERENCE EVALUATION OF STEAM BOILER EMISSION

Mincho Hadjiski, Kosta Boshnakov, Nikolinka Christova, Apostol Terziev  
Department of Automation in Industry  
University of Chemical Technology and Metallurgy-Sofia  
Kliment Ohridski Blvd. 8, 1756 Sofia, Bulgaria  
E-mails: hadjiski@uctm.edu, kb@uctm.edu, nchrist@uctm.edu

## KEYWORDS

Multi Agent System, Predictive Emission Monitoring System, multi models, aggregation, simulation, optimization.

## ABSTRACT

A multi agent based prediction monitoring system has been developed. The proposed multi agent system combines two types of autonomous agents – primitive and intelligent agents. Multi modeling techniques to predict the immeasurable gas components and following aggregation, simulation and successive optimization are used. The presented approach is a combination of Predictive Emission Monitoring System (PEMS) and Continuous Emission Monitoring System (CEMS) in order to improve the reliability of the developed monitoring system by generating replacement data in a case of sensor failure.

## INTRODUCTION

The problem of clean air and waste gas emissions are now an object of governmental regulation in most of the countries. Ecological monitoring became a wide market and field of extensive R&D. The investment load is particularly heavy for the East European countries, where the scale of environmental protection until a relatively short time ago was out of the national proprieties. Thus the strategy of emission monitoring still represents an actual challenge for investment and research (Gimenez et al.; Hadjiski and Boshnakov 2004).

The Continuous Emission Monitoring Systems (CEMS) is well known conventional design (Eberhard 1995). It can monitor flue gas emissions continuously on the base of direct measurements. However it has the disadvantages of the high installation and maintenance cost. There is a strategic need to develop alternative systems with an acceptable and competitive cost (Eberhard 1995; Ghien et al. 2003; Gimenez et al.; Hadjiski and Boshnakov 2004). The concept of a Predictive Emission Monitoring System (PEMS) (Eberhard 1995; Gimenez et al.) is to use the operating parameters of the stationary emission source in order to predict emissions through mathematical model. The main difference between CEMS and PEMS is that the PEMS does not need actual pollutant monitoring analyzers. Before

commercial operation several auditing tests must be conducted on both CEMS and PEMS to be certified. Recently several states in the USA have accepted the PEMS as a primary or alternative monitoring system. In the near future the same kind of regulations should be adopted in EC (Gimenez et al.). The main problem for PEMS is to build adequate model in order to pass the accuracy (RATA) test.

The goal of this study is to set up a PEMS in a coal – fired steam boiler by using Multi Agent System (MAS) approach (Wooldridge 2004). Our MAS realization combines two types of autonomous agents – (i) primitive agents, which can only communicate each other and perform direct algorithmic calculations or routine data performance and (ii) intelligent agents, which in addition are able to learn, to make decision, to be adaptive, to do simulation and optimization. The next peculiarity of our approach is that in contrast to the established unimodel realizations (Ghien et al. 2003; Gimenez et al.) we use multi model predicting of immeasurable gas components and following aggregation, simulation and successive optimization. The third feature is the combination of PEMS and CEMS in order to improve the reliability of the monitoring system by generating replacement data in a case of sensor failure.

The selected platform for multi agent architecture realization is JADE ([www.jadeworld.com](http://www.jadeworld.com)). The agent's software is written in Java. This ensures accordance with all FIPA specifications ([www.fipa.org](http://www.fipa.org)).

## COST ORIENTED DESIGN OF EMISSION MONITORING SYSTEM

According to the recent world experience (Ghien et al. 2003) installation and maintenance cost of PEMS are at least two times less than the corresponding expense for CEMS. In the case under consideration in this paper multiple sources must be covered by integrated monitoring system. Following the best practices in the USA and Europe it is preferable to use PEMS not only for emission monitoring but in the future re-engineering as a base for a tool to optimize combustion process (Gimenez et al.; Hadjiski and Boshnakov 2004). Thus the adopted configuration consists of minimal number of gas analyzers with multiplexing of

compulsory measurement points. As a result in admissible in conformity with regulations a sample time problem appears. Mathematical models must generate the insufficient report data. In this approach a small part of the data is received from direct measurements by CEMS, and the rest – through PEMS. The cost analysis shows a big advantage of the proposed combined system mainly due to the reduction of analyzers and operational savings.

### **SIMULATION BASED DATA RECONCILIATION**

The inference in PEMS based on the secondary measurements is specific in any particular realization. In our case the preliminary investigations show strong variability depending on: the type of approximation model, the set of input variables for each model, the aggregation approach and parameterization. In the present paper this mixed combinatorial problem has been solved on the base of successful simulation following the next scheme for each inferred gas component:

- a) Deriving a cluster of realizations for a given model from the data, varying input sets of secondary measurements.
- b) Deriving a set of clusters for three types of data driven models: interpolation-based, regression-based and neural networks-based.
- c) Simulation of the gas concentration behavior in 30 minutes intervals based on each already derived model in a form of discrete time series.
- d) Aggregation of different models.
- e) Selection of the best combination of weighted individual models by optimization.
- f) If it's necessary adaptation of the selected hybrid model.

After the computer simulation the adopted multi – model based PEMS must pass the Relative Accuracy Test Audit (RATA) in order to be certified by the authority.

### **MULTI AGENT ARCHITECTURE**

The main reasons to accept multi-agent architecture could be summarized in the following way:

- Variety of sources of information: Decentralized Control Systems (DCS), SCADA, CEMS.
- Different type of information – technological data, status signals.
- Asynchronous data flow.
- Multi-model approach for data reconciliation.
- Computer simulation as a permanent procedure.
- Necessity of intelligent operations – learning, adaptation, reconfiguration.
- Irregular internal data exchange.

The architecture of the Multi-Agent System (MAS) (Figure 1) is structured in four layers: data processing, model-based time series generation, aggregation and data reconciliation, standard processing and reporting. Each layer consists of several autonomous agents. These layers are functional, not hierarchical ones. Two types of agent are accepted:

- Primitive Agents (PA), which accomplish algorithmic calculation and data base formation.
- Intelligent Agents (IA), which undertake intelligent functions like learning, adaptation, decision making.

All autonomous agents correspond to the FIPA specifications ([www.fipa.org](http://www.fipa.org)). The multi agent architecture allows a distributed intelligent system to be realized independently of the space properties of the Thermal Power Plants (TPP).

### **AGENT FUNCTIONS**

MAS consists of thirteen autonomous agent altogether. The agents can communicate bilaterally.

The agents of the data processing layer ( $A_{11}$  - Raw data collection agent;  $A_{12}$  – Raw data preprocessing agent;  $A_{13}$  – Agent for preliminary data base) receive, synchronize, and arrange data from the sensors, make intelligent filtration and verification of the data, analyze the status signals from the preliminary data base.

Agent  $A_{12}$  analyses the condition of the whole measurement system – measurement equipment, analyzing equipment, emission source, if the waste treatment station works or not. It forms a message for reliability of the data according to the requirements of the regulations. This agent compares the data of the technological parameters and the status signals and it communicates to agent  $A_{13}$  to transfer the processed data.

Agent  $A_{13}$  communicates to agents  $A_{12}$ ,  $A_{41}$ ,  $A_{42}$  for data receiving/sending. It evaluates the completeness of the data, which are necessary for agents  $A_{21}$ ,  $A_{22}$ , and  $A_{23}$  performance. Agent  $A_{13}$  communicates to agents  $A_{24}$  and  $A_{31}$  to limit the alternative aggregations in the cases when some of agents  $A_{21}$ ,  $A_{22}$ , or  $A_{23}$  cannot generate data because of leak of measured information.

The agents from the data generation layer ( $A_{21}$ ,  $A_{22}$ ,  $A_{23}$ ,  $A_{24}$ ) realize three different type of models – interpolation ( $A_{21}$ ), statistical ( $A_{22}$ ) and neural network ( $A_{23}$ ) for each predicted gas component –  $\text{NO}_x$ ,  $\text{SO}_2$ ,  $\text{CO}$ ,  $\text{O}_2$ . Through simulation each model originates a cluster of time series by varying the sets of input variables and model structures.

These agents are intelligent ones – they could learn from current data, adapt/change model structure and parameters, make some decisions.

The generated data for each 30 min test intervals are collected into an intermediate database, which is accomplished by primitive agent A<sub>24</sub>.

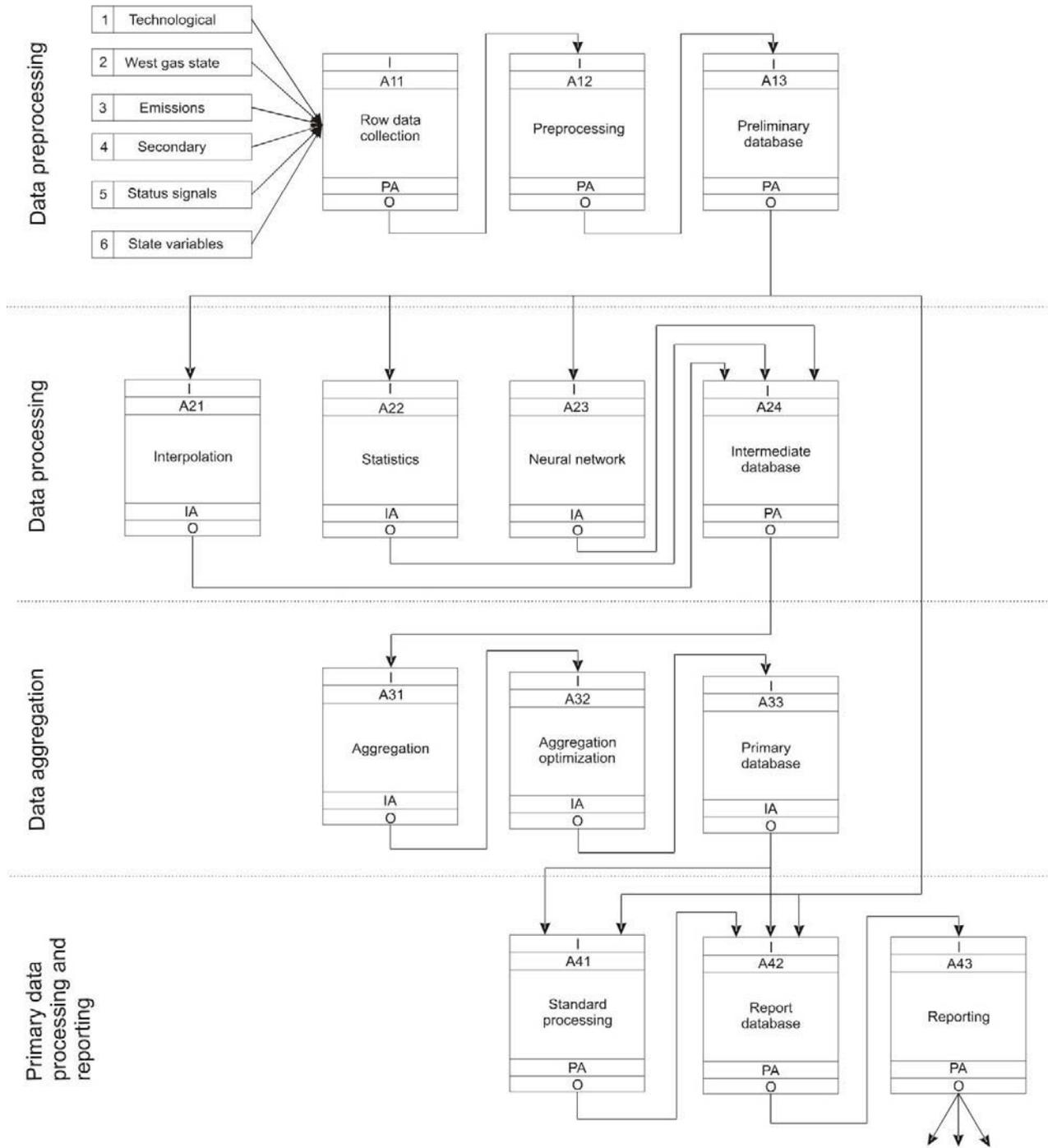


Figure 1: Structure of the multi agent based Predictive Emission Monitoring System

Agent A<sub>21</sub> receives data from agent A<sub>13</sub>. The mathematical model uses data from DSC and CEMS. This agent reconstructs the values of the analyzed gas emission parameters during sampling time 2 minutes on

the base of interpolation model using the analyses data from CEMS, performed during bigger sampling time. The reconstruction of the missed values of the gas emissions is carried out using combined approach,

including interpolation and mathematical model prediction. The implemented approach possesses ability to evaluate the necessary values of the state parameters and the emissions of the combustion process. During the interpolation function  $y = f(x)$  (equation 1) is defined. The interpolation is accomplished using Lagrange polynomial.

$$y = L_0(x)y_0 + L_1(x)y_1 + \dots + L_n(x)y_n, \quad (1)$$

where  $(x_0, y_0), (x_1, y_1) \dots (x_n, y_n)$  are the coordinates of points  $A_i$  - the direct measurements from CEMS.

The data reconstruction of the concentrations is performed as follows:

$$C_i(t_j) = C_i^{int}(t_j) + \Delta C(t_j) \quad (2)$$

where

$C_i(t_j)$  is the concentration of the  $i$ -component for the desired time  $t_j$ ;

$C_i^{int}(t_j)$  is the evaluated value of the concentration;

$\Delta C_i(t_j)$  is the correction value calculated by the mathematical model using indirect sources of information.

The general mathematical model is defined as

$$\Delta C_i(t_j) = f(B(t_j), V(t_j), O_2(t_j), \dots) \quad (3)$$

where the more important sources of indirect information are:  $B(t_j)$  - the fuel flow rate;  $V(t_j)$  - the air flow rate;  $O_2(t_j)$  - the oxygen content in the combustion gases.

A comparison of the experimental values of the nitrogen oxides  $NO_x$  concentrations (\*) and the corresponding calculated values  $NO_x$  obtained from the agent is presented at Figure 2. The relative mean square error is 3.8 %.

Agent  $A_{21}$  tunes the mathematical model during decreasing of the prediction accuracy. As well as it sends the calculated parameters values to agent  $A_{24}$ , realizing the intermediate data base.

Agent  $A_{22}$  realizes statistical model and receives data from agent  $A_{13}$ . The model uses data from the DCS to evaluate the gas emissions. Under decreasing the prediction accuracy agent  $A_{22}$  tunes the regressive mathematical model. Agent  $A_{22}$  sends the calculated parameter values to the agent  $A_{24}$ , realizing the intermediate data base.

Agent  $A_{23}$  realizing neural network model uses data from the unit DCS. The developed neural network models predict the concentrations of  $O_2$ ,  $NO_x$  and  $SO_2$  in each of the gas outlet ducts A and B.

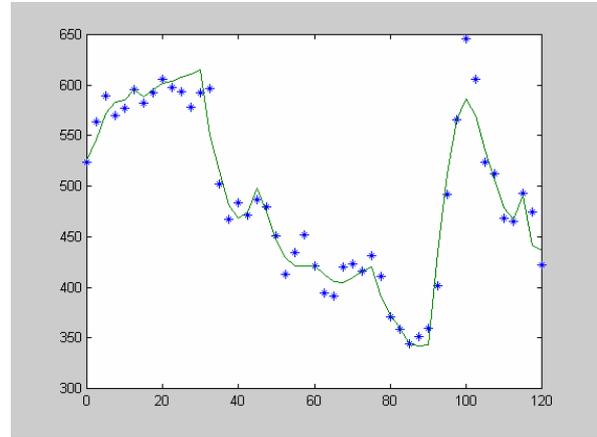


Figure 2: Experimental and calculated values of the  $NO_x$  concentrations

The highest prediction accuracy is achieved by using two layers neural networks of Cascade-Forward Backdrop type with sigmoid transfer function in the hidden layer and with linear one in the output layer. The input variables  $X_1, X_2, \dots, X_N$  are analogous to these ones, used by the statistical model (agent  $A_{22}$ ). One of the developed neural networks, namely for prediction of the emissions of nitrogen oxides  $NO_x$  in gas outlet duct A ( $NO_x^A$ ) is given at Figure 3.

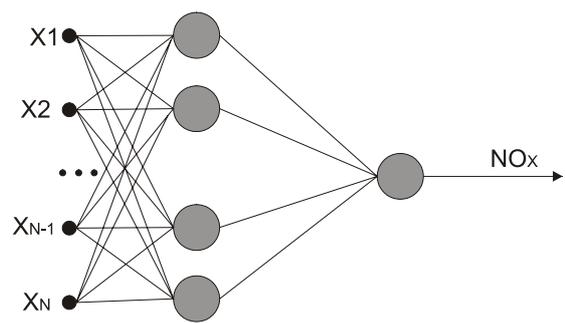


Figure 3: Neural network in Agent  $A_{23}$  for each prediction of  $NO_x$

The received simulation results are summarized in Table 1. The mean square errors of the predicted values of the gas concentrations given by agent  $A_{23}$  compared to the real experimental data are presented.

Table 1: Results from agent  $A_{23}$  performance

Gas	$O_2^A$	$O_2^B$	$NO_x^A$	$NO_x^B$	$SO_2^A$	$SO_2^B$
Mean Square Error, %	2.71	2.94	6.22	6.99	2.77	3.07

The prediction ability of agent  $A_{23}$  is illustrated at Figure 4.

When the prediction accuracy decreases agent  $A_{23}$  tunes the neural network model. Agent  $A_{23}$  sends the calculated parameters value to agent  $A_{24}$ , realizing the intermediate data base.

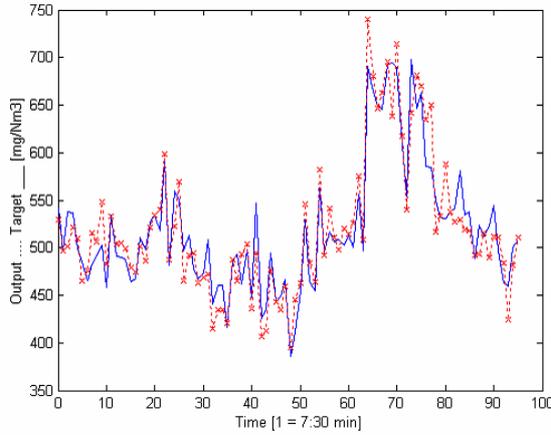


Figure 4: Results from Agent  $A_{23}$  performance

Agent  $A_{24}$ , realizing the intermediate data base communicates to agents  $A_{21}$ ,  $A_{22}$  and  $A_{23}$  to receive data and to agent  $A_{31}$  to send the required data. It forms rules for possible aggregation in agent  $A_{31}$ .

The aggregation layer consists of three agents: two intelligent ( $A_{31}$ ,  $A_{32}$ ) and one primitive ( $A_{33}$ ). The agent  $A_{31}$  aggregates different time series by weighted blending with weights, given from the optimizing agent  $A_{32}$ .

Agent  $A_{31}$  defines the aggregation method and the initial settings of the aggregation procedure. It communicates to agents  $A_{24}$  and  $A_{32}$ . Agent  $A_{31}$  forms time series at given optimization stage.

Agent  $A_{32}$  is carrying out the aggregation optimization. It calculates the difference between the predicted values by the autonomous agents  $A_{21}$ ,  $A_{22}$  and  $A_{23}$  and the current aggregated predictive model  $M(\kappa)$ . Agent  $A_{32}$  defines the mean square errors of the prediction and the parameter varying in the aggregation procedure in the next step ( $\kappa+1$ ). It stores the necessary data for evaluation of the permanent parameters, which do not vary at each step. Agent  $A_{32}$  stops the optimization process and forms the optimal prediction set, which contains primary data for the given 30 minutes interval. This agent communicates to agents  $A_{31}$  and  $A_{33}$ .

The best resulting time series are collected as reconciliated data in the agent for primary database  $A_{33}$ .

The agents from fourth layer make standard processing on primary data ( $A_{41}$ ) according to the regulations; form the complete report database ( $A_{42}$ ) and provide all the requested reports ( $A_{43}$ ). All of them are primitive agents.

## MULTI AGENT SYSTEM SOFTWARE PLATFORM

After comparative analysis of variety of MAS platforms like JADE, ZEUS, Grasshopper etc., Jade (Java Agent DEvelopment framework) was accepted for first realization of MAS based environmental monitoring system. JADE is open source software to develop agent-based application in compliance with the FIPA (Foundation for Intelligent Physical Agents) specifications that provide the normative framework within developed agents can exist, operate and communicate. In JADE agents are implemented as one thread per agent. JADE is based on the Java language and supports scheduling of cooperative behaviors and structuring complex tasks as aggregation of simpler ones. JADE allows remote management, monitoring and controlling the status of agents.

## APPLICATION

Described above Multi Agent System (MAS) was developed for the purposes of ecological monitoring of waste flue gases from coal-fired steam boilers in Thermal Power Plants. A cost oriented system was commissioned as a combination of Continuous Emission Monitoring System (CEMS) and Predictive Emission Monitoring System (PEMS). The carried out simulation tests with NO<sub>x</sub> and SO<sub>2</sub> concentration have shown high accuracy of the Predictive Emission Monitoring System (PEMS), with a standard deviation according a special direct measurements lying in the interval 3 – 7 %.

## CONCLUSIONS

The computer simulation is suitable approach for building a Predictive Emission Monitoring Systems (PEMS) for coal-fired steam boiler. The combination of direct and secondary measurements provides acceptable initial and maintenance costs. The multi agent part of the project is now in development phase based on Jade concept, mentioned above and it will be incorporated in the existing DCS before the end of the year.

## ACKNOWLEDGMENTS

This paper is supported in part by the contract No. TH1408/04 of the Bulgarian National Science Fund.

## REFERENCES

- Eberhard, W.H. 1995. "Design of an Alternative Emissions Monitoring System for Clean Air Act Compliance". *Proceedings of the Power-Gen'95*. Anaheim, CA.
- Ghien, T.W.; H. Chu; W. C. Hsu; T. K. Tseng; C. H. Hsu; and K. Y. Chen. 2003. "A Feasibility Study on the Predictive Emission Monitoring System, Applied to the Asinta Power plant of Taiwan Power Company". *Journal of the Air&Waste Management Association*. Vol. 53.
- Gimenez, A.; M. Estela; B. Benoit; F. Peledan; T. Abbas; and J. L. P. Nieto. "Advance Predictive Tool to Optimize Combustion and Emission Performance". *Cordis RTD-Projects*. <http://dbs.cordis.lu>.
- Hadjiski, M. and K. Boshnakov. 2004. "Ecological monitoring in thermal power stations based on mathematical models". *Proceedings of the International symposium "Control of the Thermoenergetic plants and systems"*. 5-6 November 2004, Bobov dol, Bulgaria, pp.7-12.
- Wooldridge, M. 2004. *An Introduction to Multiagent System*. John Wiley&Sons.
- Java Agent Development Framework (Jade), [www.jadeworld.com](http://www.jadeworld.com).
- Foundation for Intelligent Physical Agents (FIPA), [www.fipa.org](http://www.fipa.org).

# MULTI AGENT SYSTEM FOR THE SIMULATION OF AN AIRCRAFT STRUCTURE DESIGN PROCESS

Jean-Baptiste Welcomme  
Romaric Redon  
EADS CRC - Corporate Research Center  
Information Technologies Department  
Centreda 1, Avenue Didier Daurat  
F-31700 Blagnac, France  
E-mail: jean-baptiste.welcomme@eads.net  
E-mail: romaric.redon@airbus.com

## KEYWORDS

Multi-agent application, coalition formation, engineering, agent-based simulation.

## ABSTRACT

This paper investigates an approach based on a Multi Agents System (MAS) for aircraft structure design such as fuselage frame part. More precisely, we propose a simulation of an engineering process through the design of a local and distributed decision model. To design the system, first we use the MESSAGE (Caire et al. 2001) methodology to formalize the problem through a description of the role organisation. Then we describe the identified agent and the chosen resolution strategy. An application using fuzzy logic and the Multi-Agent framework (JADE) was successfully developed. We are now considering the possibility to extend the results to other structural parts of the aircraft and also to investigate other areas, where the key issue is to define/simulate appropriate technical or human organisations.

## INTRODUCTION

The design process of aeronautical structures implies that a consensus between various actors and disciplinary (design, manufacturing, costing, stress analysis...) is reached. To help the conception, the concurrent engineering proposes to achieve a maximum of tasks in parallel, and to discuss the opinions and the expectations between actors as soon as possible (Bernard and Taillandier 1998). This discussion allows to anticipate the problems and to earlier investigate some alternative solutions. On the whole we can say that it favours the collective decision-takings. However, in order to reach these requirements, it is necessary to provide solution for facilitating information sharing and coordination/cooperation between actors. Today, it seems that Multi-Agent Systems (MAS) offer new simulation and management capabilities to address these issues. In this context, we studied how to use physical agents to simulate the design of an aircraft structure such as a fuselage frame part. As in concurrent

engineering, our physical agents will have to take collective decision and to exchange information.

Through this study, we expected to evaluate the capacities of MAS

- to propose a local representation of the part to design,
- to use interaction / negotiation to simulate a design phase and thus to propose a global configuration of the part.

## MAS AND ENGINEERING

A number of MAS applications that provide assistance to engineers are already proposed with various types of architectures (Shen et al. 2001). These architectures mainly differ by the level of autonomy given to the agents. In particular, differences rest on agents capabilities to communicate straightforwardly with each other (with/without mediator), on their autonomy to take decisions... (Choi et al. 2003)

For example, the DIDE system (Shen and Barthès, 1996) showed that architectures privileging the autonomy of the agents provide dynamic and flexible systems. However, this study also showed the limits of this type of architecture when agents are too heterogeneous and numerous. In that case, it becomes necessary to structure the system through hierarchical organisations as in the Metamorph system (Shen et al. 2001). Later works such as MetaMorph II showed the advantages of combining the two approaches of autonomous agents and hierarchical organisation of agents.

We consider that the results sensitivity to the chosen agent organisation is a very interesting property of the agent technology. This property enables to simulate different organisations and to conclude about their relative efficiency.

## USE CASE DESCRIPTION

The frame function is to stiffen the aircraft fuselage. The main objective of the frame design process is to minimise its mass while supporting the applied stresses.

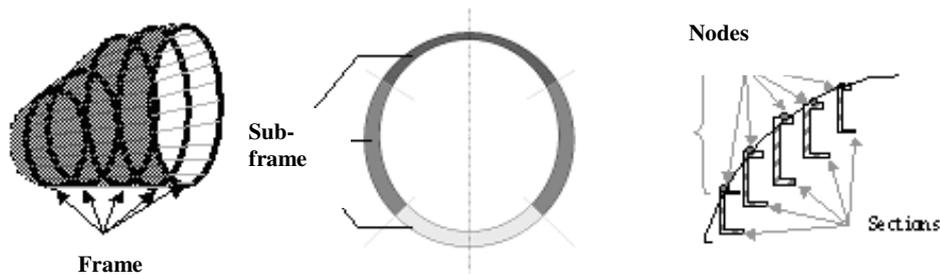


Figure 1: Frame description

A frame is not a single manufactured part, it is composed of several sub components that we will call sub-frame. The current process for designing sub-frames is to determine the frame section characteristics (shape and dimensions), at given points, (nodes) where the stress values are computed using finite element analysis methods. Although the process implies to design locally the sub-frame (section by section), the designers should also take into account global constraints. For example, he/she has to choose compatible shapes and dimensions all along the sub-frame. To resume, the challenge for designing a sub-frame is to find an effective configuration for all the sub-frame sections.

For the purpose of our study we start off with some simplified hypothesis. First, we consider only one representative dimension for the section – the height - and only four different shapes (E,C,I,J). Then, we formulated a simplified relation for each shape giving the height that supports a given stress. We call these 4 laws “height-stress laws” (figure 2).

Finally, in order to reflect the global constraints, we define a rule of shape compatibility and a rule of margin size:

- The shape compatibility rule specifies that a frame part can contain a mix of either E and C shapes or I and J shapes.
- The margin size rule defines an acceptable difference of 30 mm between two consecutives section heights.

In this article we propose to use a MAS to define an effective configuration of the sub-frame. The idea is to give some autonomy to the nodes in order to enable them to determine locally their best section characteristics. Then they use their coordination /negotiation capabilities to iteratively form coalitions (groups of nodes with common interest) until an effective configuration for the entire sub-frame is found.

## SYSTEM DESIGN

To design the system, we used the GAIA (Wooldridge et al. 2000) and MESSAGE (Caire et al. 2001) methodologies to formalize the problem and to define an adapted resolution strategy.

## Analyse

During this phase, we identified a lot of potential roles to play in the system and several possible organizations. For each role, we associated objectives (constraints to be satisfied), advantages to play it, and responsibilities like in (Ferber et al. 2003). Each agent can play one or several roles, which defines its individual capabilities. The MESSAGE methodology enticed us to use different points of view resulting in different diagrams (roles, organisation, workflow, collaboration...). By this way, we obtained a good global view of the study case, and identified several feasible organisations.

For our implementation, we choose to adopt a hybrid architecture, where the autonomy of the agents is partially controlled by a hierarchical manager, and to limit the implementation to the configuration of one sub-frame. However as we will see chapter 6, we foresee testing the other identified organisations.

The next paragraph explains the implemented strategy. It consists in creating coalitions between sections/nodes, which have similar configurations (shape, size). During this process, we make the agents negotiate and degrade their local optimal configurations to merge in the direction of a more global configuration acceptable for several other agents. The final objective is to find a single compatible configuration for the sub-frame.

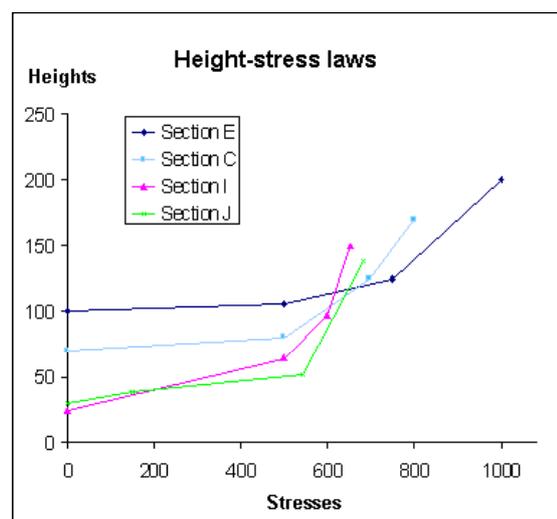


Figure 2: Height-stress laws

## Agent description

Three different types of agents are implemented to solve the problem: “the configurationService agent”, “the sub-frame agent” and “the node agent”.

The “*configurationService agent*” is a service agent that answers the demands of the “node agents”. Taking the applied stresses into account, the configurationService agent uses the height-stress laws to compute the minimal required height for each section shape. This capability is implemented through a service agent in order to enable the future use of external software and to manage the complexity by reusing simulation models, as in a Web-based modelling (Reed et al. 2000).

The “*sub-frame agent*” plays the role of a team manager for the “node agents”. Its objective is to find a set of compatible shapes for the entire sub-frame that it represents. To achieve this objective, it actually plays several roles: it attributes roles to the “node agents”; it analyses their returns of experience; it informs the user of the configuration evolution of the sub-frame... But most important, it leads and organises the negotiation between the “node agents” in order to proceed with the chosen resolution strategy, which will be described later in this article.

The “*node agents*” are the core agents of the system. Each of them has personal objectives (section shape, sizes), local knowledge (applied stresses...) communication capacities and decision capacities, based on a fuzzy logic model. Thanks to these capacities node agents are able to negotiate one with another, to evaluate the different configurations, to compare their shapes and to take decision in order to form coalitions.

## The negotiation strategy

The agents solve the problem by interacting and negotiating. Consequently, a lack of coordination may entail endless and voluminous processes, and it is

essential that the communication activities are coordinated to ensure the coherence and convergence of the system. Several solutions exist and must be generally used in parallel. According to (Mathieu and Wedge 1999), negotiations are performed at two levels: a macroscopic level (the society of agents) and a microscopic level (the resolution of conflicts). With respect to these principles, the negotiation strategy is organised as described in the figure 3.

The *macroscopic level* allows organizing the negotiation process by defining rules and exchange protocols, which define the social model and structure the dialogues. This level corresponds to the step 2 of the figure 3, when the “sub-frame agent” allocates the roles.

The *microscopic level* deals with the conflict situations and allows taking local decisions. In our model, it corresponds to the step 3. When an agent is confronted to a conflict, it uses his local decision model to find a solution.

### The coalition formation

As previously described, the system negotiates and converges by forming coalition of “node agents”. As represented in figure 3, the negotiation is an iterative process, because coalitions are formed progressively at the initiative of the “sub-frame agent” (step 1) until a single coalition is found. During the negotiation process, each “node agent” plays its attributed role (step 2). Thus when an agent has to form a coalition (former role), it proposes to the others adopting its shape. Consequently when an agent has to join (joiner role), it chooses the coalition that offers the most interesting proposition (step 3). To compare the propositions, the “node agents” use a set of fuzzy spaces (figure 4) to calculate an adequacy for each proposed configuration. (This process will be detailed bellow, see “the negotiation spaces”).

When a joiner agent finally joins a coalition, it starts playing the new role of coalition member. When a former agent obtains member(s), it plays both roles of coalition member and coalition manager, which consists in managing the coalition for the future negotiation. For

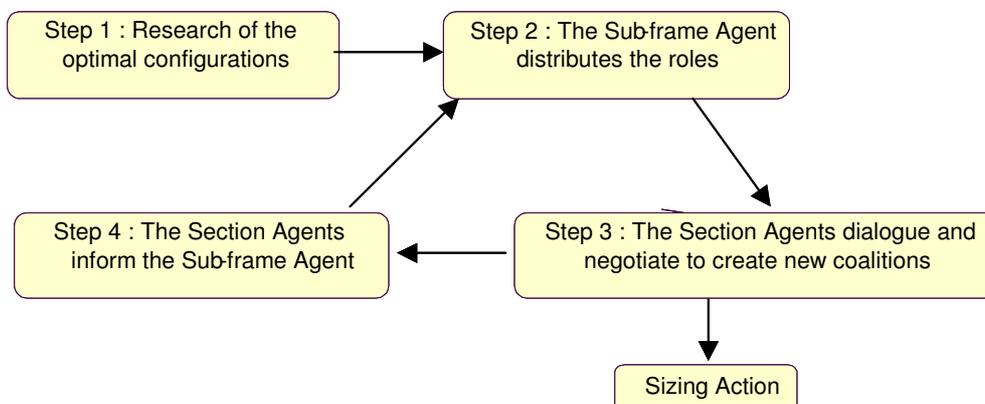


Figure 3: Negotiation Strategy

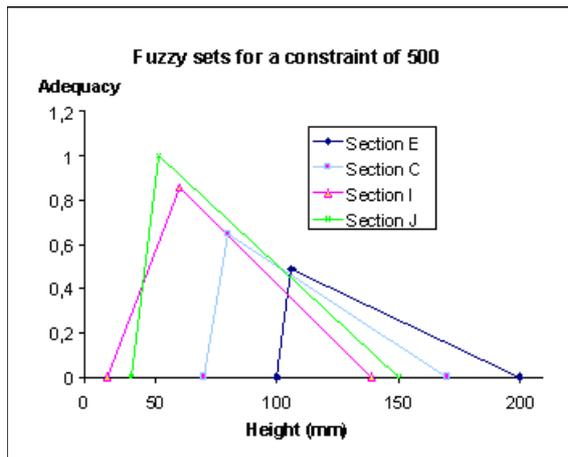


Figure 4: Fuzzy sets of a « node agent »

example, while two coalitions negotiate, the coalition manager receives/sends the propositions and takes the best decisions for all its members. Particularly, it can demand the members to compute their adequacy, to change shape for joining a coalition, to change size with respect to the margins...

In this way, the “sub-frame agent” controls a part of the strategy; attributing the roles (former, joiner) and enabling the system to converge (macroscopic level). But another part of the strategy (microscopic level) is achieved through the negotiation and the decision of the “node agent” as in the sizing action (figure 3 step 3)

#### Sizing action

As described, when a new coalition is created, a compatible shape is chosen by the set of agents. When the margin size between two consecutive sections is not respected a sizing phase is required. This phase allows each agent to choose a size, which respects the margin. The sizing step is an emergent process, because each “node agent” is going to adjust its dimensions with its neighbours without any central process or control. If its size is too small, the agent increases it and informs again its neighbours, who verify their compatibility with their new environment. So, the modifications are made locally and propagated across the sub-frame part only if necessary.

#### The negotiation spaces and adequacy computation

To proceed with the negotiation between agents we measure a node local adequacy, which is computed using fuzzy logic models. This adequacy is equal to 1 when the node characteristics are locally optimal. Then for a given shape, the adequacy decreases linearly according to the height. Each “node agent” builds its 4 fuzzy logic models (I, J, C and E) as follow:

1. The “configurationService agent” provides the best heights for each shape.
2. Each node agent constructs triangular spaces according to the configurationService heights (tops

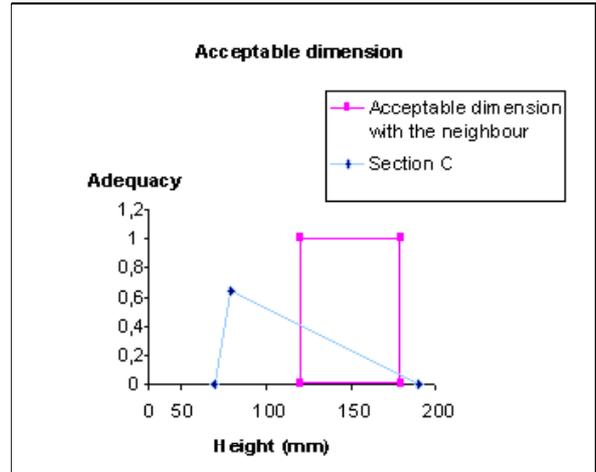


Figure 5: Adequacy determination

of the triangle) and to the absolute maximum/minimum height for each section (bases of the triangle) (figure4)

Finally, the fuzzy sets are constructed with respect to the global objective of weight minimisation (a penalty function is applied to heavier sections), and are used to estimate the adequacy with a proposition of a neighbour node. For example, if an agent receives a proposition in a C shape with a dimension of 150 mm (figure 5), it determines that it can take a height between 120 and 180 mm (if the margin size between neighbours is 30mm). With an intersection between its fuzzy set (C shape) and a second fuzzy set that represents the compatible sizes with the proposition, it can determine the best configuration to take and calculate its adequacy (figure 5).

To sum up, the fuzzy sets enable:

- To determine the satisfaction of the agents, by calculating the adequacy of a configuration
- To have a simple and local decision process and to compare the propositions.
- To have a simple model that can be easily customised in order to better fit with the reality

## IMPLEMENTATION AND RESULTS

An application was developed according to the previously described concepts. This application uses the Multi-Agent framework (JADE) and the fuzzy logic (FuzzyJ) to define the negotiation spaces.

During the simulation, the 18 agents find an acceptable configuration after 5 processes of negotiation. Every line of the figure 6 corresponds to a negotiation cycle. The histograms in the left part give the configurations chosen by the agents (height, shape, adequacy). The right part of the figure represents the formed coalitions. For each coalition, we also give the chosen shape and the coalition adequacy, which is computed as the average of the coalition member adequacies. With these

two representations, we can observe the evolution of each “node agent” during the negotiations (shape, height, adequacy), as well as the formation of coalitions. To summarize:

- At first according to the constraints, the agents choose the local optimal configuration (shape and height). This configuration corresponds to the local optimal solution.
- After a first negotiation cycle, the agents are gathered together around very similar configurations (local configurations are a little degraded).
- But during the following negotiations, the agents make more important concessions and finally find a global and acceptable configuration for the entire sub-frame.
- In our example, the agents that made the most of concessions are situated at the extremities of the sub-frame.
- Within each formed coalition, sections sizes are adapted in order to respect the margin. Consequently, the histograms progressively adopt a smoothed evolution.

### FUTURE WORK

The results obtained are interesting, because they show that the MAS enable to find a solution to the problem with a local and distributed decision process. Nevertheless, several improvements are to be brought that can be ordered according to three directions:

1. Improve the Multi-Agent System and test further organisations of agents, because we would like to

evaluate the influence of the organisation on the final results;

- A first experimentation will consist in increasing the competences of the “sub-frame agent”. Having a more complex strategy of role distribution, could influence the final results, and explore a more delimited solution space. The construction of this strategy could be automatic and based on the experience of the “sub-frame agent”.
  - A second experimentation will consist in reviewing the negotiation process and giving more local degree of freedom to the “node agents”. In such a case, the “sub-frame agent” would just choose the initiator of the negotiation, but wouldn’t define directly the agent roles.
  - A last alternative will be to use a completely local logic. In that case, we get closer with a purely emergent solution. This approach is also interesting and allows to investigate the Distributed Constraint Satisfaction Problems (DisCSP) (Modi et al. 2003; Mesh and Lesser 2004). However this type of approach was presented for simple problems and it is not sure that it is well adapted to the configuration of an entire frame, which remains the final goal of our study.
2. Extend the use case and take into account more complex hypothesis.
    - First of all, it is necessary to get closer to the reality. For example, we would like to take into account several constraint situations (take off, landing...) and several sizing dimensions for the

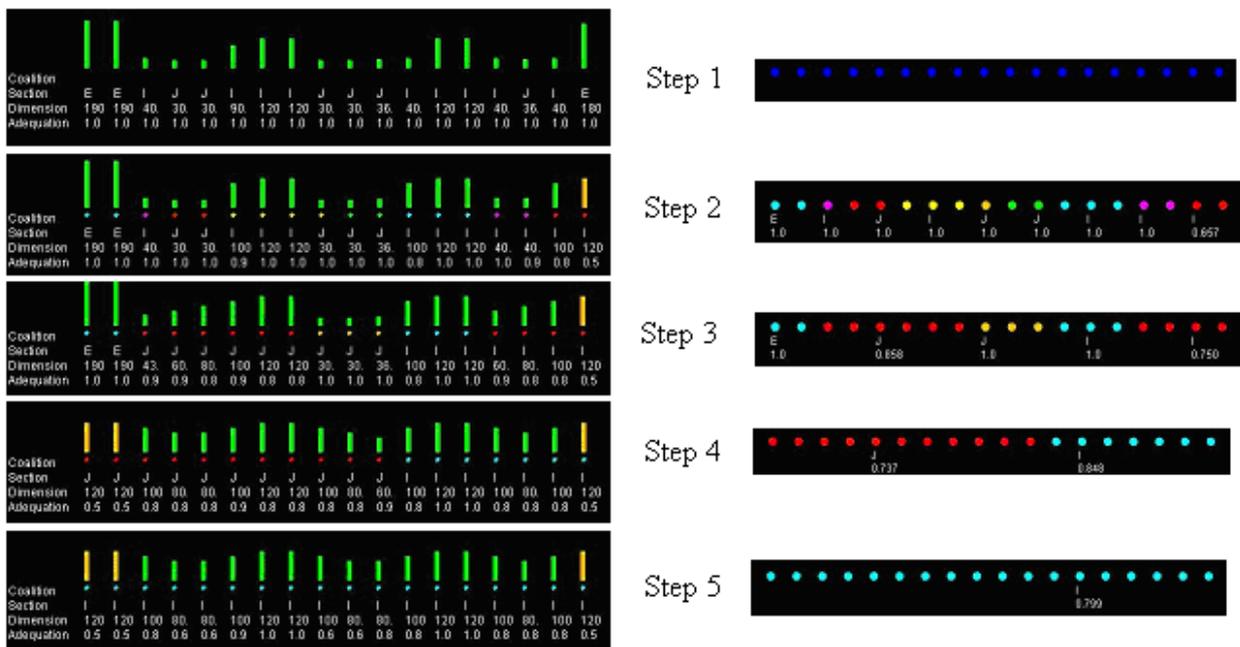


Figure 6: Results of a simulation

node (height, width, length).

- To complete the use case, we would have first to consider the sizing of the entire frame. Then in a second step, it could be very interesting to consider the sizing issues of several frames.
3. Introduce the possibility of human interactions with the system. This last point is very important to investigate, because a real system must be understandable and verifiable by the user.

## CONCLUSION

In this article, we presented a MAS for the simulation of an aircraft structure design process. The design of the system showed that methods such as MESSAGE (Cairo et al. 2001) enable the efficient identification of the roles and agents. The implementation phase achieved with the framework (JADE), allowed us to validate the design choices and demonstrated that MAS are well adapted to the simulation of complex systems. This study participated in the evaluation of MAS for EADS, and allowed investigating agents negotiation processes using local knowledge and distributed decisions.

To conclude, in our view the simulation with MAS offers very good perspectives, and first results encourage us to consider the possibility to simulate/aid the design of other technical parts of the aircraft. More generally, we think the agent-based simulations could be applied for EADS in many other areas, especially, when it is justified, in simulating/testing organisation (technical/human).

## REFERENCES

- Bernard, A. and M. G. Taillandier. 1998. “*Le prototypage rapide*”. Editions Hermès, Paris.
- Caire, G.; F. Leal; P. Chainho; R. Evans; F. Garijo; J. Gomez; J. Pavon; P. Kearney; J. Stark; and P. Massonet. “Agent oriented analysis using MESSAGE/UML”. 2001. In *Proceeding of the Second International Workshop on Agent Oriented Software Engineering*. Montreal Canada, 101-108.
- Choi, H. R.; H. S. Kim; J. Y. Park; B. J. Park; and Y. S. Park. “Multi-Agent based negotiation support systems for order based manufacturers”. 2003. In *Proceedings of the 5th international conference on Electronic commerce*, 479 – 487.
- Ferber, J.; O. Gutknecht; and F. Michel. “From Agents to Organizations: An Organizational View of Multi-agent Systems”. 2003. In *Proceedings of Second International Joint Conference on Autonomous Agents and Multiagent Systems*. ACM Press, 214-230
- FuzzyJ Toolkit, NRC-National Research Council of Canada's JADE (Java Agent DEvelopment Framework)
- Mailler, R. and V. Lesser. “Solving Distributed Constraint Optimization Problems Using Cooperative Mediation”. 2004. In *Proceedings of Third International Joint Conference on Autonomous Agents and Multiagent Systems*. IEEE Computer Society, 438-445.
- Mathieu, P. and A. Taquet. “Une forme de négociation pour systèmes multi-agents”. 1999. In *Proceeding Journées Francophones pour l'Intelligence Artificielle Distribuée et les Systèmes Multi-Agents (JFIADSMA'99)*. Edition Hermès, Paris.
- Modi, P-J; W-M. Shen; M. Tambe; and M. Yokoo. “An asynchronous complete method for distributed constraint optimisation”. 2003. In *Proceedings of Second International Joint Conference on Autonomous Agents and Multiagent Systems*. ACM Press, 161 – 168.
- Reed, J.A.; G.J. Follen; and A.A. Afjeh. “Improving the aircraft design process using Web-based modeling and simulation”. 2000. *ACM Transactions on Modeling and Computer Simulation*. 58-83.
- Shen, W.; D. H. Norrie; and J.P. Barthès. 2001. “*Multi-Agent Systems for Concurrent Intelligent Design and Manufacturing*”. Taylor and Francis.
- Shen, W. and J.P. Barthès. “An Experimental Multi-Agent Environment for Engineering Design”. 1996. *International Journal of Cooperative Information Systems*, Vol. 5, Nr. 2-3, 131-151.
- Wooldridge, M.; N. R. Jennings; and D. Kinny. “The Gaia Methodology for AgentOriented Analysis and Design”. 2000. *Autonomous Agents and Multiagent Systems*. Kluwer Academic Publishers, Vol. 3, 285-312.

## AUTHORS BIBLIOGRAPHIES



**JEAN-BAPTISTE WELCOMME** is a PhD Student at the EADS Corporate Research Centre in the Information Technologies Department. He is researching on MAS for simulating the design and the integration of complex systems. He received a M. S. in Computer Science from the Technology University of Belfort in 2003 and a M. S. in Artificial Intelligence from the University Pierre and Mary Curie of Paris in 2004. His e-mail address is: [welcomme@gmail.com](mailto:welcomme@gmail.com)



**ROMARIC REDON** went to the National Polytechnic Institute of Grenoble, FRANCE, where he obtained his degree of mechanical engineering in 1999. Since then, he worked in the fields of Knowledge Based Engineering (KBE). First as a consultant, for the company Knowledge Technologies International he developed several KBE application to assist and/or automate aerospace part and tool design processes. Then he joined the EADS common research center, where he worked on an integrated environment for supporting KBE lifecycle. He also started an activity on MAS and their application to the simulation of complex aerospace systems. His e-mail address is: [romaric.redon@airbus.com](mailto:romaric.redon@airbus.com)



**Computational Modelling  
and Simulation  
in  
Science and Engineering**



# COMPUTATIONAL MODELING AND SIMULATION OF RECONFIGURABLE RESPONSIVE EMBEDDED COMPUTING SYSTEMS

Dietmar P. F. Moeller  
University of Hamburg, Department Computer Science  
Chair Computer Engineering, AB TIS  
Vogt Kölln Str. 30, D-22527 Hamburg, Germany  
dietmar.moeller@informatik.uni-hamburg.de

## INTRODUCTION

Many computer applications required as solution provider inside larger, in general, complex systems. Hence these computer systems are embedded within the real complex systems, which themselves may consist of mechanical, electrical and/or electronical components etc. This led to the well-known terms of embedded systems, and often emdedded processors or embedded control. Embedded systems, as well as embedded control, with high reliability and hard realtime constraints, are the top end of embedded systems design, called embedded responsive systems.

With the growing demands in this field, solutions are requested integrating operating systems, realtime capabilities as well as fault tolerance within a overall system approach. Due to those facts the term responisble system was introduced for such type of complex and configurable architectures. The most important features of responsive systems deal with reliability and realtime capability.

Reliability as well as hard realtime capability demand both for intelligent scheduling algorithms. Scheduling means, that even in the sequentialized world of the processor concurrent requests may be served. Normally this is performed by static or dynamic scheduling algorithms with well-known advantages and disadvantages and checked by simulation methods.

But intelligent scheduling is nothing else as scheduling per se because looking for the best priority schedule of the tasks in progress. Hence the hardcore boundary for responsive systems cannot be solved by the scheduling schemes known up to now. The other aspect is that the embedded controller as well as the embedded processor architecture inside shows no difference at all in compa-rison to 'normal' architectures.

This paper introduces a block oriented structure for em-bedded controller as well as embedded processors in the area of *responsive systems*. This architecture will include computing capabilities for high performance tasks as well as hardware support for scheduling, and this yields to the need of simulation for system design and evaluation. Even fault tolerance – partial as well as global – is possible done based on simulation within the embedded controller architecture. This support should be introduced as hard-ware/software partitioning based on an extended block-oriented

microarchitecture model for the embedded con-troller design in *responsive systems*.

The demands for the simulation method change from separated hardware, operating system and application software simulation to integrated system co-simulation methods due to the mutual dependance of all system parts.

Using system modelling languages like VHDL for the block oriented FPLD architecture it could be shown that some of the simulation results are obtainable by abstraction but some not. Field Programmable Logic De-vices (FPLDs) like Field Programmable Gate Arrays (FPGAs) have emerged as an ultimate solution to the time-to-market and risk problems in embedded controller design because they provide instant manufacturing and low-cost prototyping e.g. of embedded processor kernels, using simulation as a powerful design tool. An FPLD is a device in which the final logic structure can be directly configured by the end user, without the use of an inte-grated circuit fabrication facility, because the user only needs some simple electrical equipment for programming purposes.

First type of programmable logic device introduced into the market was the Programmable Read Only Memory (PROM), a one-time programmable device that consists of an array of read-only cells. Later these cells were as-sembled in EPROM or EEPROM technology enabling the re-programmability. A specific logic circuit can be imple-mented by using the PROM's address lines as the circuit's inputs, and the circuit's outputs are then defined by the stored binary information. Based on this strategy, any binary function

$$F := B^n \rightarrow B^k$$

can be realized

## RESPONSIVENESS OF EMBEDDED INFORMATION SYSTEMS

In embedded information systems it is important to perform with the correctness of computations in a timely manner, meaning responsiveness of the embedded computing system, that requires correctness of the computation stringent timing constraints due to

- Reliability
- Predictability

- Adaptability
- Timely
- Criticalness

Responsiveness can be achieved with real-time embedded information systems. Real-time embedded information systems are fundamentally composed of two or more concurrent processes that execute with stringent timing requirements and cooperate with each other in order to accomplish a common goal. Typically, a real-time embedded information systems have timeliness requirements, typically in the form of deadlines that can't be missed. They consists of a control system and a controlled system. For example, in an automated production line, the controlled system is the production floor with its robots, assembling stations, and the assembled parts, while the control system is the real-time embedded information systems and the human computer interface that manage and coordinate the activities on the factory floor. Henceforth, the controlled system can be viewed as the environment with that the embedded information systems interacts, based on the information available, that requires periodic monitoring of the environment as well as timely processing of the sensed information.

Responsive or real-time embedded information systems differ from conventional information systems while having deadlines or explicit timing constraints, that are attached to tasks, to that the responsive or real-time embedded information systems has to react on-time, otherwise timing faults may cause catastrophic consequences. In addition to timing constraints, a task may also posses the following types of constraints and requirements:

- Resource constraints, means that a task may require access to certain resources other than the CPU such as I/O devices, data structures, files, etc.
- Precedence relationship, means that a complex task may require access to many resources, that can be sufficiently handled break it into several subtasks related by precedence constraints and each requiring a subset of that resources
- Concurrent constraints, means that tasks should be allowed concurrent access to resources providing the consistency of the resources is not violated
- Communication or networking requirements, that are mostly timing requirements
- Placement constraints, means that instances of a task are executed – for fault-tolerant reasons – on different processors
- Criticalness, means that meeting the deadline of one task may be considered more critical than another.

The characteristics of the various application specific tasks are usually known a priori and can be scheduled statically or dynamically. In static scheduling, boosted by parallel code optimization, the compiler of the embedded information system has to

detect and to resolve data, control and resource dependencies during code generation. Moreover, the compiler also performs parallel code optimization. The static specification of schedules is typically in use for periodic tasks, the opposite is true for use in a-periodic tasks. Let an embedded system being static. The characteristics of the controlled system, are assumed to be known a priori, and, hence, the activities and the sequences in which these activities take place can be determined off-line, means before the system is running. Embedded information systems of such type are quite inflexible even though they may incur lower run-time overheads. Whereas a large proportion of currently implemented traditional real-time systems are static, by necessity, responsive embedded computing systems has to adopt solutions that are more dynamic and flexible. Hence, responsiveness in embedded systems requires for system characteristics, such as:

- Fast
- Predictive
- Reliable
- Adaptive

Therefore, a responsive embedded information system has to be fast and predictable. Predictability means that a task that is activated should be possible to determine its completion time with certainty. This can be done taking into account the state of the system and the task resource needs.

Reliability is another prerequisite on a responsive embedded information system, meaning that real-time constraints cannot be achieved if embedded information system components are not reliable. The degree of reliability has to be specifiable and predictability, and will involve determining system performance under different levels of reliability.

Adaptability is of particularly importance for a responsive embedded information system, because if a task deadline can be met only under a restricted system state or configuration, reliability and performance may be compromised.

There are three popular strategies on static scheduling:

- Register distribution, that is mainly used to avoid structural hazards
- Loop reorganization, that prevents both structural and data hazards
- Code promotion, that is used to compensate for losses caused by data hazards

Often code is written such that its behavior is clear to other programmers. Since code is optimized for the programmer its typically not optimized for the architecture. Programmers will often dedicate general-purpose registers to the same task throughout their code. This will make code easier to understand but it can place unnecessary over use some registers and under use others. This creates competition for the use of the over used registers and this causes structural hazards. This is unfortunate since there are usually other registers in the architecture that could be used to avoid the structural hazard. Register redistribution techniques can be applied to a code after the software

designer has written it to evenly distribute the register usage. In this way the software designer can create the code from the designers perspective and the static scheduler can optimize it for the architecture.

Just as the programmer specifies register utilization for readability loops are written to be readable. For this reason the loop index calculations are often calculated at inefficient times. A good example is a loop written to traverse the pixels in an image. The following program code excerpts are written in C and demonstrate loop index inefficiency.

Example 1:

```
char* pImage1;
char* pImage2;

int nRows = 480;
int nCols = 640;
int row, col;

for( row = 0; row < nRows; ++row )
{
    for( col = 0; col < nCols; ++col
    )
    {
        pImage1[(row*nCols) + col] =
        pImage2[(row*nCols) + col];
    }
}
```

Example 2:

```
char* pImage1;
char* pImage2;

int nTotalPixels = 640 * 480;
int pixelIndex;

for( pixelIndex = 0; pixelIndex <
nTotalPixels; ++pixelIndex )
{
    pImage1[ pixelIndex ] =
    pImage2[ pixelIndex ];
}
```

In both examples the data in *Image1* is copied to *Image2*. The difference is the logical technique used for traversing the images and the system for computing the image indexes. In Example 1, two separate loop indices are used, requiring two registers. In Example 2, only a single index is used. Not only are more registers used for the same task but also they both must be compared to detect loop completion. In Example 1 the index of an individual pixel is computed using an integer multiply and an integer add. Additionally this computation is executed twice even though the two computations will have the same result. The total number of computations executed in Example 1 is 614400 multiplies and 922080 adds. In Example 2 the pixel index for both images is the same integer and its computed using a single integer add. Additionally the

integer used as the pixel index is the loop index. In Example 2 the total number of computations is 0 Multiplies and 307200 adds. Example 2 has less than one third the number of adds and expert two and has no multiplies. The relative cost reduction of Example 2 is clear from these figures. The importance of loop reorganization is shown by the fact that Example 2 is easier to understand from a programming perspective. Its very structure suggests that it is traversing through the rows and columns of an image and copying the data across. For this reason the example given above will be written more commonly by software designers. A loop reorganization strategy would identify the dependencies in Example 1 and factor out the redundant actions. After this process Example 2, which was clear to the programmer would be synthesized as Example 1.

Avoiding data hazards is often achieved by simply stalling the issue of instructions that may be affected by a data hazard. When this is done the entire processing stream is held up. A technique for avoiding this stall is to look further back in the processing stream for independent instructions. If instructions that may be executed out of order without creating additional hazards can be found then their execution may be promoted to fill the stalled execution cycles. In this way the data hazard causing instructions are delayed until the hazard is cleared but the progress of the execution stream is not stalled.

Dynamic scheduling involves the addition of specialized hardware to an embedded processor that detects and mitigates hazards. A dynamic scheduler can mitigate hazards using the same techniques applied to static scheduling. These techniques include code promotion, loop reorganization, and resource redistribution. Dynamic scheduling requires a hardware-based system to detect active functional units and data. This management unit then only issues instructions that do not depend on any active data or functional units, and is often called a dispatch unit. Additionally a functional unit must be included which tracks when instructions finish and relinquish resources. This unit is often called a retirement unit. All of the specialized hardware involved in dynamic scheduling takes up precious silicon resources that could be applied to further parallelism. Additionally it requires that the complex techniques involved in safely rescheduling instruction flow must be implemented in hardware and cannot be updated in the field. Since the technical challenges of dynamic scheduling are great it became common practice to boost the performance of dynamic scheduling by using static parallel code optimization, that is performed either by a separate post-pass code optimization that follows a traditional compiler, or by enhancing the traditional compiler with parallel code optimization.

Traditional compilers speed up sequential execution and reduce the required memory space mainly by eliminating redundant operations. In case that the code scheduler usually follows the traditional sequential optimizer in the back-end-part before register allocation and subsequent code generation this type of

code scheduling is called pre-pass-scheduling. The other approach is to use a traditional optimizing compiler and carry out code afterwards, that is called post-pass scheduling.

Code scheduling can be performed on three different levels:

- Basic block scheduling
- Loop scheduling
- Global scheduling

where basic block scheduling is the simplest but least effective code scheduling technique. Here, only instructions within a basic block are eligible for recording. As a consequence, the achievable speed-up is limited by both data and control dependencies.

The next level of code scheduling is loop-level scheduling, where instructions belonging to consecutive iterations of a loop can usually be overlapped, resulting in considerable speed-up. However, recurrences may impede speed-up.

The most effective way to schedule is at the highest possible level, called global code scheduling, where parallelism is sought and extracted beyond basic blocks and simple loops, in constructs called compound program, involving loops as well as conditional control constructs. In case of a chunk of code larger than a basic block, there are many possibilities for code scheduling.

In general, responsiveness of embedded computing systems can be introduced as some kind of process scheduling, that is based on three basic concepts:

- Process states
- State transition diagram
- Scheduling policy

As far as process states are concerned, there are three basic states connected with scheduling:

- Ready-to-run state, meaning processes are able to run when a processor is allocated for them
- Running state, meaning execution on the allocated processor
- Wait or blocked state, meaning processes are suspended or blocked waiting for the occurrence of some event before getting ready to run again

Possible state transitions and their conditions are stated in the state transition diagram, shown in Fig. 5.19. When the scheduler selects a process for execution, its state changed from ready-to-run to running. The process remains in this state until one of the following three events occur:

- Depending on the scheduling policy the scheduler decides to cease execution of the process and puts it into the ready-to-run queue again, changing its state accordingly
- The process in execution may issue an instruction that causes this process to wait until an event takes place, meaning the process state is changed to the wait state
- If the process reaches the end of execution, it terminates

Finally, the scheduling policy component specifies rules for managing multiple competing processes in that way selecting the next thread to run, based on scheduling algorithms, such as run-to-completion, shortest-job-first, earliest-deadline-first, etc. The term thread is another name for a task. This term is more common in operating systems that support processes, while a task is simply a thread in a single-process system.

## RECONFIGURABLE EMBEDDED COMPUTING SYSTEMS PLATFORMS

Dynamically reprogrammable field programmable gate arrays or other techniques to implement hardware that varies in form and function over time representing the platform on which reconfigurable embedded computing systems can be build up. In a sense, FPGAs can be thought of as a hardware computing system on which one set of computing primitives might execute for a while, then another set, etc. Henceforth, an array of programmable logic devices, such as FPGAs, can be used as an execution platform for one or more hardware objects, e.g. a reconfigurable processing unit. The novelty component of the FPGA is its large internal configuration memory, and the two possible modes of operation, the download mode and the configuration mode.

For the implementation of reconfigurable embedded computing systems, based on FPGAs, one simply connects together in a regular mesh,  $n \times m$  identical programmable logic blocks, that are cell based devices with many simple logic elements, as fundamental basis for the user dependent configuration, to build up complex logic functionality based on the concept of modularity. The interconnections possible with a FPGA are huge but the realized circuits speed depends on the used place and route algorithms. Therefore, the time dependent behavior is not predictable. A FPGA is a device with which the final logical structure can directly be configured by the user, without the use of an integrated circuit fabrication facility, because the user only needs some simple electrical equipment. The fast innovation cycles in the development of FPGAs has gained opportunities which are not possible with other technologies. Because the programming of the FPGA can be changed very fast, without rewiring or refabrication, its only reconfiguration.

The conceptual structure of a typical FPGA consists of a two-dimensional array of configurable logic blocks (CLB) that can be connected by general interconnection resources. The interconnection comprises segments of wire, where the segments may be of various lengths. At present interconnections are programmable switches that serve to connect the CLB's to the wire segments, or one wire segment to another. Logic circuits are implemented in the FPGA by partitioning the logic into individual logic blocks and then interconnecting the blocks as required via the switches.

The structure and the content of a logic block represents the logic block architecture. Logic block architecture can be designed in several ways like static RAM cells (SRAM), anti-fuse technology, EPROM and EEPROM technology. Some FPGA logic blocks are as simple as NAND Gates, other blocks have a more complex structure, such as multiplexers or lookup tables (LUT). Consider the basic elements of the CLB are small SRAM cells (LUT) that can be used implementing any logical function, one can distinguish between three different configuration types:

- Configuration type 1, that consists of an array of discrete LUTs, each of which depending on a set of input variables  $x$

$$f = f_i(x_i); \text{ where } 1 < i < k$$

- Configuration type 2, that can be introduced being a two level LUT structure. The first level is identical to the configuration type 1, but, the outputs of the first level are combined with the second level. The second level logic can be arbitrary or restricted to some specific Boolean functions. Assuming the first level outputs as  $g_i = g_i(x_i)$ , the second level output can be defined as follows:

$$f(x_1, x_2, \dots, x_k) = g_1(x_1) \circ g_2(x_2) \circ \dots \circ g_3(x_3);$$

$$\text{where } \circ \in \{\Delta, \nabla, \oplus, \equiv, \dots\}$$

- Configuration type 3, that can be introduced as a structure with two LUT's sharing input variables. Additionally to an arbitrary number of common variables  $x_2$  each LUT might depend on a set of further disjunctive inputs  $x_1$  and  $x_3$ . Therefore, the output can be written as follows:

$$f_{pair}(x) = [f_a(x_1 * x_2); f_b(x_2 * x_3)];$$

$$\text{where } x = x_1 \cup x_2 \cup x_3 \text{ and } x_2 \cap x_3$$

Over the last few years, a number of different types of FPGAs have been launched into the market, each of which has unique features, that can be classified into the following four categories:

- Symmetrical array FPGA
- Row based FPGA
- Hierarchical FPGA
- Sea-of-gates FPGA

## EMBEDDED PROCESSOR KERNEL DESIGN FLOW

The goal of this part is to present a method for implementing a processor kernel on an FPGA. We restrict ourselves on the design of the ALU as one kernel part of the CPU to show how well-known or specialized operations may be performed on an FPGA.

The description of a processor kernel circuit can be entered using a schematic capture program with a simulation tool box like OrCAD, View-Logic, Synario etc. This involves using a graphical interface to interconnect circuit blocks. The available building blocks are taken from a component library.

An alternative way specifying an embedded processor kernel circuits is to use Boolean expression or State Machine language. With this methods, no graphical interface is needed. After the processor kernel circuit has been fully designed and merged into a circuit, it is translated into a special format that is understood by the used CAD tool, e.g. as netlist format. The partitioner partitions the circuit into logic cells of the selected FPGA which means a technology mapping which converts the processor kernel circuit, which is a netlist of basic logic gates, into a netlist of specific FPGA logic cells.

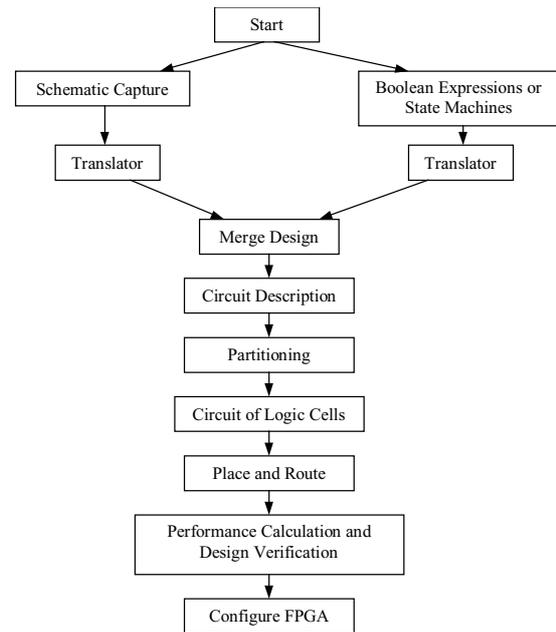


Figure 1: Design Flow for FPGAs

Placement means, each logic cell generated during the partitioning step is assigned to a specific location inside the IC, This is automatically done by CAD tools, or manually by the user in order to optimize the fitting. location in the FPGA. Automated placement is done using simulated annealing algorithms. After placement, the required interconnections among the logic cells must be realized by selecting wire segments and routing switches within the FPGAs interconnection resources.

Once the circuit is routed, the physical paths of all signals within the FPGA are known. Hence it is possible checking the performance of the implementation, which can be done either by downloading the configuration bits into the FPGA and checking the part within its circuit board, which will be a non effective method when designing complex kernels, or, which is the better way, by using a simulation tool box with timing analysis. If the performance or functionality of the circuit is not acceptable, shown by simulation, it will be necessary to modify the design at some point in the design flow in order to optimize timing and functionality by simulation runs with changed parameter sets.

All the descriptions above show how to implement a well-defined processor into any kind of FPL, but using FPLs it would be possible to change the processor kernel itself for obtaining best performance using this system design. This leads to Application-Specific Processor Design (ASP), and it should be clearly understood that inside this reported area of Hardware/Software Co-Design there will be always several level of simulation with mutual dependance. Fig. 2 outlines this situation.

Level 1 describes the simulation on the circuit level using a field-programmable device. This simulation may be performed before or after routing phase and will be responsible for correct functionality during real runtime. Level 1 may be omitted when using CPLDs with guaranteed timing behaviour through the whole device and will then be substituted by using computed running times and clock rates.

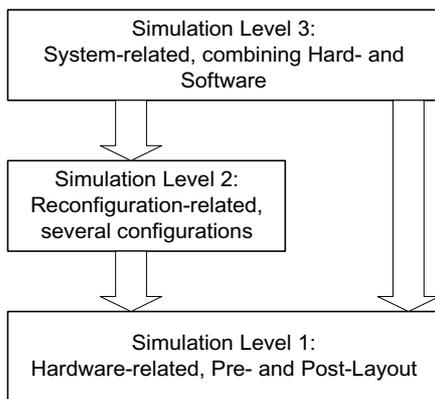


Figure 2: Levels of Simulation for a Mixed Processor/Software Design

On the other side level 3 is a system simulation level. The device, e.g. the processor kernel, will be modelled on a behavioural level for several reasons:

- Simulation on behavioural level saves (simulation) runtime
- Modelling on behavioural level offers a first well-defined device definition and will be often the first existing description.

As a result the runtime behaviour of the whole system will be simulated (and defined) inside this level.

Therefore the system designer has to describe the system knowing many details of level 1.

Situation is getting very complex when level 2 is no longer empty. This level integrates runtime reconfigurability and has to take in account that reconfiguration will cost time. It will be very difficult to obtain any analytical cost function which will compute the complete system runtime including hardware runtime, reconfiguration times (for several possibilities) and number of instructions needed to solve a given problem – and this under all runtime conditions. System simulation with levels 2 and 3 and at least knowledge of the behaviour in level 1 will be the only possible way of obtaining quantitative results for the system behaviour.

The design of a reconfigurable FPGA based CPU kernel as part of an embedded computing system design is assumed being based on the following assumptions.

- CPU kernel, as shown in Fig. 3
- Logical Operations such as
  - AND ACC, D
  - XOR ACC, D
  - NOT ACC
  - NAND ACC, D
  - INC ACC
  - OR
  - DEC ACC
  - NOR ACC, D
- Aruthmetical Operations
- VHDL description, as shown in Figure 4.

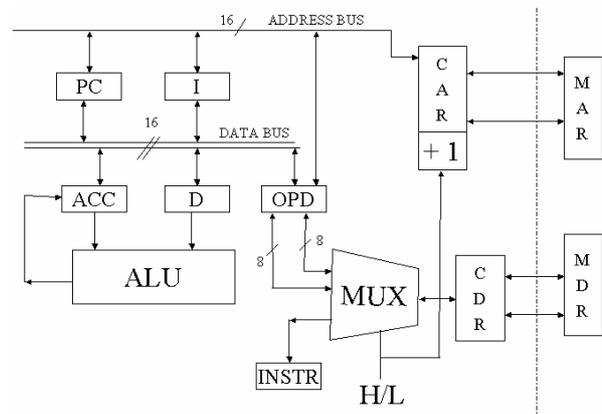


Figure 3: Block diagram of the processor kernel

The implementation of the FPGA based processor kernel may be restricted due to pin limitations by the IOB count or by complexity due to the CLB count of the FPGA chosen. In these cases another FPGA series should be chosen or the respective module has to be partitioned into smaller parts, each of them fitting the given constraints. Partitioning optimize the logical structure of the CLA. It could be seen that the number of internal signals between the different levels is twice the number of outputs. Hence, horizontal cut lines are not suitable for partitioning since in most cases pin limitation is more restrictive than complexity.

```

Library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.std_logic_unsigned.all;

Entity CSMP02 is
-- THIS SECTION OF THE VHDL CODE DESCRIBES
-- THE EXTERNAL PINS OF THE MICROPROCESSOR
End CSMP02;

Architecture CSMP02_main of CSMP02 is
-- DESCRIBES THE BEHAVIOR OF THE MICROPROCESSOR
-- DECLARATION OF CONTROL UNIT STATES
-- DECLARATION OF MEMORY UNIT STATES
-- DECLARATION OF INTERNAL SIGNALS AND REGISTERS
-- INITIALIZATION OF SIGNALS AND REGISTERS
-- CONTROL UNIT STATE MACHINE
-- RESET, FETCH, DECODE AND EXECUTE.
-- MEMORY UNIT STATE MACHINE.
-- INITIALIZATION OF I/O'S
End CSMP02_main ;

```

Figure 4: VHDL Diagram of the Processor Kernel

One of the most exciting novelties of FPGAs move beyond the implementation of embedded computing systems and instead harness large numbers of FPGAs as general-purpose computation medium.

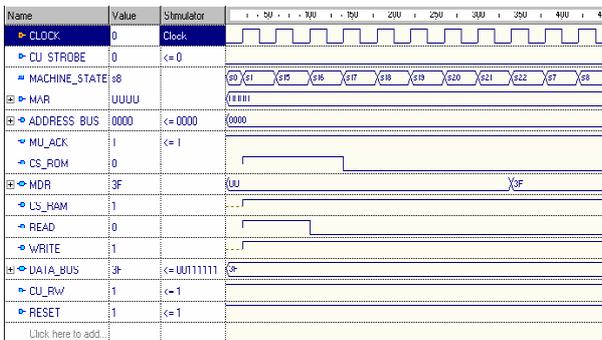


Figure 6: Microprocessor timing diagram

The embedded computing system architecture mapped into the FPGA needs not only be described based on standard hardware equations but can even be operated from general computational algorithms. While these FPGA based customized computing machines may not challenge the performance of embedded computing systems for all application areas, for computations of the right form, FPGA based embedded computing systems can offer extremely high performance, surpassing any other programmable solution. Although a customized hardware approach beats the power of any other generic programmable system, and thus there must always be a faster solution.

An FPGA based embedded computing system, that can be reprogrammed like a standard PC, offers the highest realizable performance for many different applications. In a sense, it is a hardware supercomputer, surpassing traditional machine architectures for certain applications. Because of their reprogrammability and cellular arrangement, FPGA's can process large data streams fast and simultaneous, providing a huge amount of parallel computing capability, a need if not a must in real time embedded computing systems applications.

## REFERENCES

Berger, A. S. 2002. "Embedded Systems Design" CMP Books, Berkely, 2002.

Brown, S.D., Francis, R.J., Rose, J., Vranesic, Z.G. "Field-Programmable Gate Arrays". Kluwer Academic Publishers, Boston, 1992

Hartenstein, R.W., Glesner, M. (Eds.) "Field-Programmable Logic and Applications". *Lecture Notes in Computer Science Vol. 1142*, Springer Verlag, Berlin, 1996

Marwedel, P. 2003. "Embedded System Design". Kluwer Academic Publ., Boston, 2003.

Möller, D. P. F. 2003. "Embedded Systems Processor Kernel Design". In: 5<sup>th</sup> Internat. Conf. Computer Simulation and AI, pp. 11-18, Ed.: S. Raczynski, Universidad Panamericana Publ. Mexiko, 2000

Möller, D.P.F., Siemers C. "Simulation of an Embedded Processor Kernel Design on SRAM based FPGA" In: *Proceed. 31<sup>st</sup> Summer Computer Simulation Conference*, pp 633-638, Eds.: M. S. Obaidat, A. Nisanci, B. Sadoun, SCS Press San Diego, 1999

Möller, D. P. F., Siemers, C., Roth S. 1998. "A Concept for New Architecture Paradigm for Hardware/Software Co-Design". In: *Computer Architecture 98*, pp. 179-180. Ed.: J. Morris, Springer Publ. Singapore, 1998

Schmitz, M. T., Al-Hashimi, B., M., Eles, P. 2004. "System-Level Design Techniques for Energy-Efficient Embedded Systems". Kluwer Academic Publ. Boston, 2004.

## AUTHORS



**DIETMAR P. F. MÖLLER** was born in Preetz, Germany. He enrolled at the Universities of Lübeck, Bremen Mainz and Bonn, where he studied Electrical Engineering and Human Medicine. He obtained his doctoral degree in 1980. From 1985 to 1991 Dr. Möller lead the [anaesthesia](#) division of Dräger AG in Lübeck, Germany. 1991 he has been elected as Full Professor for Computer Engineering (TI) at the Technical University of Clausthal. In 1998 he has been elected as Full Professor for Computer Engineering Systems (AB TIS) at the University of Hamburg where he holds the chair for Computer Engineering. Since 1998 he also is Adjunct Professor at the California State University Chico. His E-Mail address is [dmoeller@informatik.uni-hamburg.de](mailto:dmoeller@informatik.uni-hamburg.de) and his Web-page can be found at <http://www.informatik.uni-hamburg.de/TIS/>

# UNDERSTANDING AND PREDICTING THE ELECTRONIC AND DYNAMIC BEHAVIOR OF NANOSCALE MAGNETIC RANDOM ACCESS MEMORY (MRAM) CELLS USING MICROMAGNETIC MODELLING AND SIMULATION

Markus-A. B. W. Bolte  
Dr. Guido D. Meier  
Institut für Angewandte Physik und Zentrum für  
Mikrostrukturforschung, Universität Hamburg,  
Jungiusstrasse 11, D-20355 Hamburg, Germany  
E-Mail: [mbolte@physik.uni-hamburg.de](mailto:mbolte@physik.uni-hamburg.de)

Prof. Dr.-Ing. Dietmar P.F. Möller  
Fachbereich Informatik,  
Arbeitsbereich Technische Informatiksysteme  
Universität Hamburg,  
Vogt-Kölln-Strasse 30, D-22527 Hamburg, Germany

## KEYWORDS

Simulation of phenomena in solid state physics, Nanomagnetism, Micromagnetic Model, MRAM

## ABSTRACT

In recent years, micromagnetic simulation has evolved into a vital tool for understanding and optimizing the electronic and dynamic properties of magnetic structures on the nanometer scale. The main focus of current research has moved from establishment of correct models and simulation tools to understanding the results furnished by such tools, with the exception of two major fields that are currently given much attention. The first is the study of the interaction between the magnetization or spin of a ferromagnetic material with electric currents, and secondly the study of the dynamic properties in such nanomagnets when switching between different magnetization states, e.g., spin wave modes, damping, etc. We have adapted an existing simulation code by optimizing its performance for dynamic simulations, integrated a feature that computes one type of current-spin interaction, and computed the Fourier transformation into frequency space thus revealing the spin wave modes of typical nanomagnetic structures. In this paper, we present the fundamental equations and methods implemented in the simulation of current-spin interactions and in the calculation of spin wave modes, as well as results from these simulations and computations.

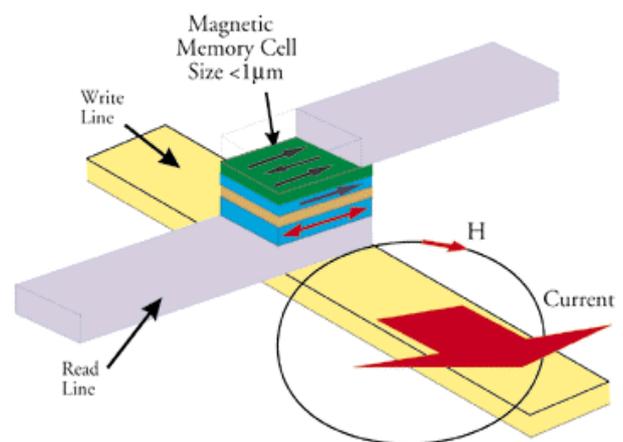
## INTRODUCTION

With the current progress in processing power and storage density, several challenges arise from fundamental physical limitations. Fundamentally new storage devices will be needed when the current semiconductor technology will have become exhausted. One of the most promising advances for fast, high-

density storage is the use of magnetic random access memory (MRAM) cells. Storage of information in MRAM cells has several key advantages:

- It is non-volatile, because its information is stored by magnetic rather than electric states so that the information remains even when a device is switched off.
- It consumes less power than DRAMs or SRAMs, because it doesn't need bit refreshment.
- It is potentially faster (d'Aquino et al. 2004; Ounadjela 2004) than DRAM or SRAM.

Figure 1 illustrates how an MRAM cells works in general:



**Figure 1:** How an MRAM cell works: The cell is made up out of several layers, a soft magnetic layer (green) that can alter its magnetization direction in the presence of external magnetic fields, and hard magnetic layers for reference. A current running perpendicularly through the layers (via the purple read line) detects the resistance due to the relative alignments of the spins in the different layers; a current through a write line (yellow) creates an external field strong enough to flip the magnetization in the soft magnetic layer.

Due to these and other features, i.e., the easy inclusion into existing CMOS fabrication processes (Ounadjela 2004), MRAMs can revolutionize the way computers architectures are designed. With MRAMs, one can easily include larger inexpensive permanent data storage elements very close to the processing units, thus eliminating the need for complicated multi-level CPU-architectures. Ultra-fast and large memory units onchip with CMOS-processing units would also allow for innovative multiprocessor architectures, which would more adequately meet current computing power demands.

To understand and produce reliable non-volatile memory cells, the several requirements must be guaranteed. The information that is stored in these cells must be stable for several years, and the writing to and reading from the cells has to be reliable for every one out of several hundred thousand memory cells in a device in order to make the technology feasible for mass production.

Among the questions to be answered are:

- The minimal distance between neighboring MRAM cells, i.e., the distance in which nearest-neighbor-interference from stray-field interaction between the cells becomes minimal;
- Understanding and control of the switching behavior in minimal time (Savtchenko et al. 2003);
- An investigation of new methods to accelerate the switching processes, for example, by heating the storage cells with an electric current (heat assisted magnetic switching – HAMS) or by using the interaction of conduction electrons and the electrons within the ferromagnet responsible for the magnetic moments (Current-induced switching – CIS) (Weller 2004).

In this paper, we present recent results of our simulations that have helped to receive answers to the questions mentioned above. We first outline the underlying model, discuss some advancements on our simulation program that have helped to speed up the computation by two orders of magnitude, and show some results, especially to answer the question concerning the interaction of neighboring nanometer-sized multilayer ferromagnets. We also show some simulation results of current-spin interaction via the anisotropic magnetoresistance effect.

## THE MODEL

The micromagnetic model is described in great details in standard works (Aharoni 1996; Brown 1963, Hubert and Schäfer 1998) and has been laid out at a previous conference (Bolte et al. 2004).

### Acceleration of the simulation

The Runge-Kutta integration scheme is chosen as a first step toward greater computing speed because it is well established and very robust and reliable. The authors of “Numerical Recipes” (Press et al. 2002) suggest it as a

good start to numerical integration, especially when details of the problems, like their smoothness or stiffness, are unknown. They call it “their workhorse”, and present a rather competitive implementation of it.

Their version, found by Cash and Karp, a fourth order Runge-Kutta integration with a fifth order error estimate and adaptive time steps is given in (Press et al. 2002), but the algorithm had to be applied to the specific problem domain and fitted into the object oriented micromagnetic framework. With OOMMF’s (Donahue and Porter 1999) flexible module structure the former Euler-integrator could then easily be substituted for a Cash-Karp Runge-Kutta evolver. Extensive tests and performance evaluations confirmed the correctness of the implementation.

### Numerical Stability

The error estimation for the Runge-Kutta implementation follows the algorithm of Cash-Karp as given by

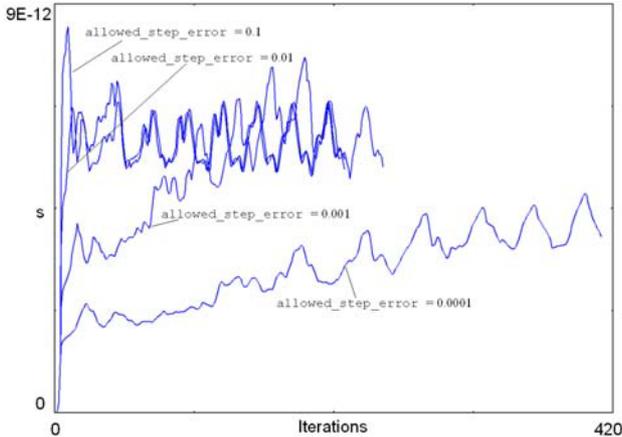
$$\Delta \equiv y_{n+1} - y_{n+1}^* = \sum_{i=1}^6 (c_i - c_i^*) k_i . \quad (1)$$

$$h_0 = h_1 \left| \frac{\Delta_0}{\Delta_1} \right|^{1/5} , \quad (2)$$

where  $y$  is the function values of two approximations at time step  $n+1$ ,  $c$  the weights of derivatives  $k$ ,  $\Delta_0$  the allowed and  $\Delta_1$  the actual step errors, and  $h_0$  and  $h_1$  the next and current step size, respectively. If the allowed error per time interval is chosen too large, then the time step can be so wide, that the integration „jumps“ beyond the optimal path in the energy terrain. Also the numerical error can increase continually so that the boundary condition  $\delta G \leq 0$  can be violated. If on the other hand, the allowed error is chosen too small, then the step size will automatically decrease according to formula (2) until a stable average time step is reached. A smaller time step directly corresponds to longer computing time, so that an optimal value between the two extremes needs to be found. Figure 2 plots the results of an evaluation of several test runs with different error boundaries. The problem solved for this test run was a small  $100 \times 100 \times 10 \text{ nm}^3$  permalloy square with medium damping ( $\alpha=0.1$ ) relaxing from a uniform magnetization in x-direction with an external magnetic field of 14 mT that is aligned diagonally in positive x-y-direction switched on at time  $t=0$ . This field was applied to ensure that there would be one path into equilibrium, unlike the other cases where several energetic minima exist.

The graphs reveal that all the allowed errors between 0.1 and 0.0001 degrees per nanosecond lead to stable solutions, but that for very small error bounds the time step needs longer to converge to its optimal value of around  $6.5 \cdot 10^{-12} \text{ s}$ . For values higher than one the system becomes instable. The least iterations were needed for an error boundary of 0.01 degrees per nanosecond. An alteration of the damping factor  $\alpha$  revealed that the border of stability is shifted towards smaller values for the allowed error, because dynamic problems

naturally require a smaller time step to adequately represent the precession of the magnetization. For very small damping factors, errors beyond one degree per nanosecond led to unstable systems. The size of the problem didn't seem to have much of an influence on the numerical stability. In all following simulations, the allowed error value was set to a default of 0.01 degrees per nanosecond to achieve maximum speedup.



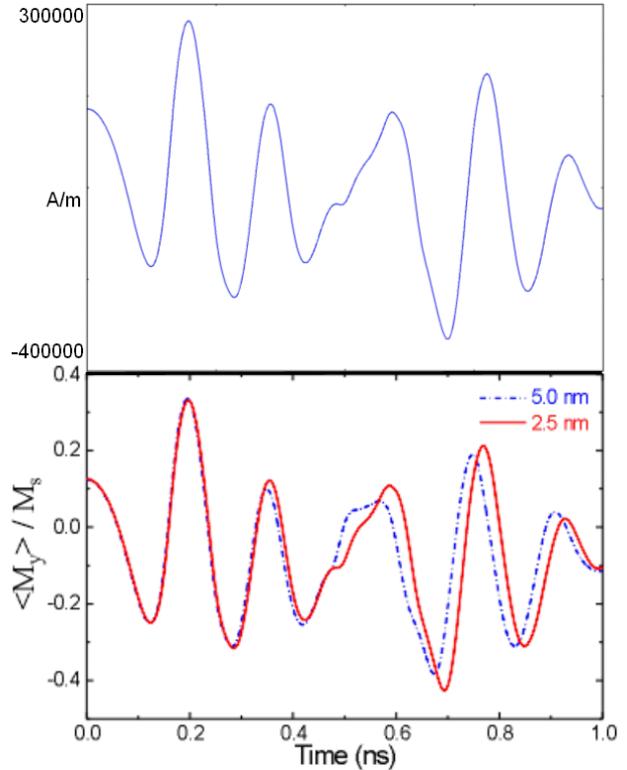
**Figure 2:** Stepsize of a Runge-Kutta-solver for different error boundaries.

### Correctness

One way to test complex numerical calculations for their correctness is to establish solutions through different algorithms for certain simple or standardized problems, ideally those, for which an analytical solution is known. Such standard problems have been designed by  $\mu$ MAG (Micromagnetic Modeling Activity Group) for micromagnetic simulation, located at the National Institute of Standards and Technology (NIST).

The Runge-Kutta implementation was tested with hundreds of trial runs, comparing not only the final magnetization configuration, but also the energy path the individual algorithms took towards equilibrium. For example, a systematic scan through the problem space was performed to derive a phase diagram to compare the performance of Euler versus Runge-Kutta as functions of the damping coefficient  $\alpha$  and the problem size  $N$  (see Figure 4). It was also tested on three of the four  $\mu$ MAG standard problems, namely problems #1, #2, and #4. Problems #1 and #2 deals with the correct computation of an energetic ground state by solving the basic micromagnetic equations (Bolte et al. 2004), while problem #4 is requiring all the simulation variables to agree with prescribed values for every simulation step, a much stricter demand than simply agreeing in the final result. As will be explained in detail later, the Runge-Kutta implementation didn't perform well for problems with large damping parameter, and so for the static standard problems #1 and #2, the Runge-Kutta was outperformed by the Euler method. But for standard problem #4, where an external magnetic field causes a switching in a sample's magnetization, Runge-Kutta was 9.1 times faster than the conventional Euler algo-

rithm. In all cases, Runge-Kutta yielded very similar results to those already given as solutions on the website of  $\mu$ MAG and similar to results by the Euler method. For example, there is basically perfect agreement in the values of the key variables (see Figure 3).



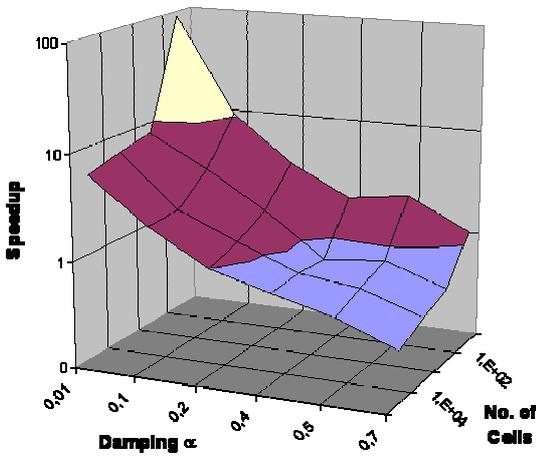
**Figure 3:** Second part of the  $\mu$ MAG standard problem #4: Component of the spatially averaged magnetization of the sample as a function of time. Top:  $M_y$ -components as derived by Runge-Kutta for a cell size of 1 nm. Bottom reference values from (Buda et al. 2001) for two cell sizes.

However, slightly different results between the two methods are to be expected since they don't have the same time steps and therefore calculate different derivatives  $dm/dt$  at each step, leading to different paths. Generally, the deviations in the final results are minimal. Results from the second approach will be compared to experimental techniques in the following section, including magnetic-force microscopy (MFM) (Barthelmeß et al. 2004, Bolte et al. 2005), magnetic transmission x-ray microscopy (MTXM) (Fischer 2002, Bolte et al. 2005) and measuring the anisotropic magnetoresistance (AMR) (Steiner et al. 2004, Steiner et al. 2005) effect.

### Performance

The implementation of the Runge-Kutta-evolver was examined upon its computational performance by solving problems with various damping coefficients and number of simulation cells. It was initially expected that a Runge-Kutta integration with adaptive step size would be one to two orders of magnitude faster than the "slow" Euler method (Press et al. 2002). What was found was a performance gain that strongly depends on (a) the size

of the problem and (b) the damping coefficient, and that for strong damping and large problems the Euler would actually outperform the Runge-Kutta implementation. The speedup factor found by simulating the micromagnetic problems at sample points in the problem space is shown in Figure 4. The problems investigated were square permalloy blocks with homogeneous magnetization along one of the long axis as starting configuration. Five different sizes were used for these blocks: 100, 200, 400, and twice 1000 nm in xy-direction having 10, 20, 40, 10, and 100 nm thickness, respectively. The cell size was taken 10 nm throughout the whole set of problems. The damping factor was altered for each of these sizes between 0.01 and 0.7 with a total of six different values. Each problem was run for both integration methods, most of them several times for statistical purposes.



**Figure 4:** Phase diagram of the performance of the Runge-Kutta implementation in relation to the Euler method as a function of damping factor  $\alpha$  and the problem size  $N$ .

The speedup was deduced as the quotient of the total computational times needed to relax a micromagnetic problem into equilibrium with the Runge-Kutta versus the Euler method (for small problems) or by computing a problem for a given simulated time and then comparing the computing time (for larger problems). It was ensured that other boundary conditions of the simulation, such as the number of outputs and input parameters were equal, because these factors influence the computation time. From the data retrieved, the average step size, average computing time per time step and the total computing time for each sample point were calculated. The phase diagram for the speedup factor as a function of damping coefficient and size of the problem (Figure 4, note the logarithmic scale for the speedup factor) shows that the performance of the Runge-Kutta in relation to Euler decreases with increasing size and damping; the damping has a greater effect on the performance than the size of the problem. There is a small area where the benefit of the Runge-Kutta algorithm is one to two orders of magnitude faster than Euler; this is for small damping (0.01 to 0.1). The Runge-Kutta does better than the Euler in a large region

where either the damping is less than 0.2 or the size of the problem is very small. For the remainder of the problems Euler was actually faster than the Runge-Kutta implementation.

Because this size and damping dependent behavior is not intuitive, two questions can be asked:

- Why does the Runge-Kutta implementation scale with the size of the problem?
- Why does it scale with damping  $\alpha$ ?

The answer to the first question stated above is simply this: Due to the higher time complexity of the demagnetization routine,  $O(N \log N)$  instead of  $O(N)$  as the other terms, the algorithm needs more and more time for increasingly large problems sizes  $N$  to compute the solution as compared to the Euler method, because the logarithm grows faster than any constant value. An increasing percentage  $p$  of the total computing time will be spent in the  $O(N \log N)$ -routine as opposed to the percentage of time  $q$  spent in the recurring  $O(N)$ -routines

$$t_{RK} = [5 \cdot (p+q) + (1-p-q)] t_{Eu} = [1+4(p+q)] t_{Eu} \quad (3)$$

until nearly all of it is spent in calculating the stray-field energy, converging into the constant speedup

$$s = \frac{t_{Eu}}{t_{RK}} \cdot \frac{\tau_{RK}}{\tau_{Eu}}; \lim_{p \rightarrow 0} s = \frac{\tau_{RK}}{\tau_{Eu}} \text{ and } \lim_{p \rightarrow 1} s = \frac{\tau_{RK}}{5 \cdot \tau_{Eu}}. \quad (4)$$

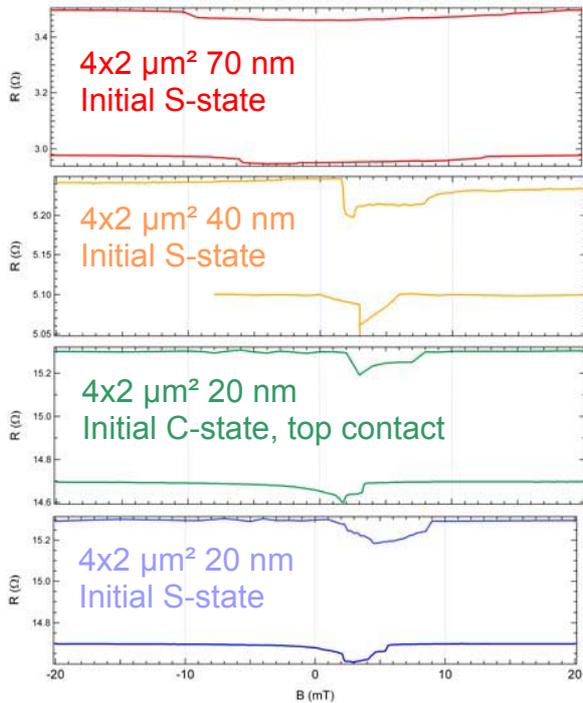
Here,  $\tau$  are the simulation time steps per iteration for the Runge-Kutta (RK) and the Euler (Eu) algorithm, and  $t$  the computing time needed to further the simulation by one step by the corresponding algorithm.

To answer the second question, one needs to look at the ratio of the simulation time steps  $\tau$ , the other part of the speedup in (4), a clear picture appears. The ratio of time steps decreases by one-and-a-half orders of magnitude between low and high damping values. A closer look reveals that the time step for the Runge-Kutta does not change as significantly as does the step size for the Euler method. The Euler algorithm is able to take more advantage of the accelerated simulation for higher damping than the Runge-Kutta. For very high damping, the Euler steps are almost as large as the Runge-Kutta steps, which makes it clear that in no way can the Runge-Kutta outperform the Euler method for static problems.

## EXTENSIONS TO THE STANDARD PACKAGE

The speedup achieved by the Runge-Kutta implementation greatly facilitates the use of micromagnetic simulation in the study of magnetic switching processes. Runge-Kutta is only the first of many possible numerical integration schemes (Fidler and Schrefl 2000). Several extensions to the OOMMF package were developed in order to mimic magnetoresistance measurement and to allow spectral analysis of dynamic switching effects, and to implement current-magnetization interaction currently not included in the

standard micromagnetic model. With these extensions, the study of dynamic processes in magnetic micro- and nanostructures is greatly facilitated.

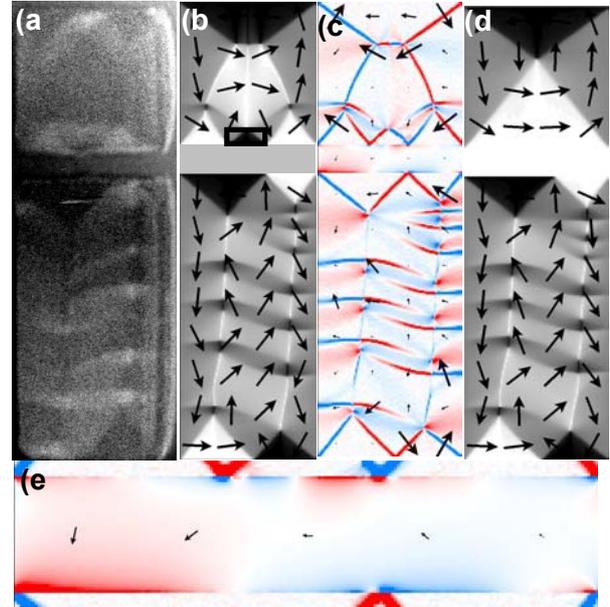


**Figure 5:** Measurement and simulation of the anisotropic magnetoresistance of a  $4 \times 2 \mu\text{m}^2$  permalloy rectangular ferromagnet with varying thicknesses. The graphs depict the absolute values of the potential difference between two gold contacts situated on top of the magnets. Note the good quantitative agreement between the simulation and the measurements. With the exception of the yellow curve, the bottom curve is from the measurement, and the top curve from simulation. See (Steiner et al. 2005) for details. Taken from (Steiner et al. 2005).

### SIMULATION OF THE STRAY-FIELD INTERACTION OF MRAM CELLS

For an introduction to the physics of magnetic-force microscopy, magnetic transmission X-ray microscopy, and magnetoresistance measurements, we refer the reader to (Hubert and Schäfer 1998; Fischer 2002; McGuire and Potter 1975). A brief outline to the results furnished by the combined use of micromagnetic simulation, MTXM and MFM is given in the following paragraphs, for more detail refer to (Meier et al. 2004; Bolte et al. 2005). Meier et al. in their publication show that the magnetic configurations of Fe/Ni-double layered microstructures as seen by MFM and MTXM can be compared by using micromagnetic simulation as a bridge between the measuring techniques. The simulation also allows a more detailed interpretation of experimental data, as shown in (Bolte et al. 2005). Here, the simulation yields information about the internal magnetic configuration and energy distribution which information is not accessible by experimental methods.

The difference in energy between two observed magnetic configurations was determined to be about 12 percent of the total energy. By reformulating the integrals of the stray-field energy calculation, the stray-field interaction of two neighboring magnetic microelements and even the local distribution of the stray-field can be determined. The results of this computation is shown in Figure 5.



**Figure 5:** (a) MTXM measurement of Fe/Ni double layered microelements at +8 mT. (b) Simulated magnetization with a black rectangle indicating the pinned region, (c) simulated stray field of magnetization shown in (b), colors represent field strength along the long axis (from red to blue). (d) alternative simulated magnetization when no pinning is assumed. (e) Enlargement of the stray-field distribution within the gap showing interaction between the domains. The arrows represent the averaged strength and direction of magnetization or stray-field of a cluster of simulation cells.

It provides evidence for the existence of pinned magnetization, i.e., impurities in the material that cause the magnetization to resist changes in an external magnetic field, in the investigated microelements. The stray-field energy outside of the elements amounts to 32% of the total stray-field energy of the double contact or roughly 11% of its total magnetic energy. Figure 5(c) shows the local stray-field strengths in direction of the long axis, with black representing 50 mT or higher. It reaffirms that most of the stray field is stored inside the domain walls. Figure (e) shows the distribution of stray fields inside the small gap between the elements, having local densities up to 50 mT, and average values across the gap of 0.6 mT (center) to 10 mT (left edge). Taken from (Bolte et al. 2005).

### CONCLUSION

Through implementation of a more efficient numerical integration algorithm, a micromagnetic simulation tool

experiences a speedup of one to two orders of magnitude for dynamic (lowly damped) problems. Further extensions enable the simulation of the anisotropic magnetoresistance, current-magnetization interaction and spectral analysis. Those extensions are important to gaining a deeper understanding of fast and current-driven switching processes that are necessary to the development of fast, non-volatile magnetic random access memory devices.

## REFERENCES

- Aharoni, A. 1996. *Introduction to the Theory of Ferromagnetism*. Clarendon, Oxford.
- Barthelmeß, M.; Thieme, A.; Eiselt, R.; and Meier, G. 2003. "Stray fields of iron electrodes for spin-polarized transport". *J. Appl. Phys.* **93**, 8400-8404.
- Barthelmeß, M.; Pels, C.; Thieme, A.; and Meier, G. 2004. "Stray fields of domains in permalloy microstructures - measurements and simulations". *J. Appl. Phys.* **95**, 5641
- Bolte, M.; D. P. F. Möller; and Meier, G. 2004. "Simulation of Micromagnetic Phenomena". *Proceedings of the 18th European Simulation Conference*. SCS Publishing House. 407-412.
- Bolte, M.; Eiselt, R.; Eimüller, Th.; Fischer, P.; and Meier, G. 2005. "Micromagnetic simulation as a bridge between magnetic-force and magnetic-transmission x-ray microscopy". *J. Magn. Magn. Mat.* **290-291**, 723-725.
- Brown, W. F. Jr. 1963. *Micromagnetics*. Interscience Publishers, New York.
- Buda, L. D.; Prejbeanu, I. L.; Demand, M.; Ebels, U.; and Ounadjela, K. 2001. The 8th Joint MMM-Intermag Conference 2001 Proceedings.
- d'Aquino, M.; Scholz, W.; Schrefl, Th.; Serpico, C.; and Fidler, J. 2004. "Numerical and analytical study of fast precessional switching". *J. Appl. Phys.* **95**, 7055.
- Donahue, M. J. and Porter, D. G. 1999. "OOMMF User's Guide, Version 1.0" Interagency Report NISTIR 6376, National Institute of Standards and Technology, Gaithersburg, MD (Sept 1999)
- Fidler, J. and Schrefl, Th. 2000. "Topical Review: Micromagnetic Modelling – the Current State of the Art". *J. Phys. D: Appl. Phys.* **33**, R135.
- Fischer, P.; Eimüller, Th.; Schütz, G.; Denbeaux, G.; Lucero, A.; Johnson, L.; Attwood, D.; Tsunashima, S.; Kumazawa, M.; Takagi, N. Köhler, M.; and Bayreuther, G. 2001. *Rev. Sci. Instr.* **72**, 2322.
- Hubert, A. and Schäfer, R. 1998. *Magnetic Domains: The Analysis of Magnetic Microstructures*. Springer.
- Meier, G.; Eiselt, R.; Bolte, M.; Barthelmeß, M.; Eimüller, Th.; and Fischer, P. 2004. "Comparative study of magnetization reversal in isolated and stray field coupled micro contacts". *Appl. Phys. Lett.* **85**, 1193.
- Ounadjela, K. 2004. "Industrial Applications for Nano-Magnetism". IFF Spring School "Magnetism goes Nano" lecture, Institute of Solid State Research, Jülich, Germany.
- Press, W.; Teukolsky, S.A.; Vetterling, W.T.; and Flannery, B.P. 2002. *Numerical Recipes in C, The Art of Scientific Computing, 2<sup>nd</sup> Ed.* Cambridge University Press, New York, NY
- Rüdiger, U.; Yu, J.; Thomas, L.; Parkin, S.S.P.; Kent, A.D. 1999. "Magnetoresistance, Micromagnetism, and Domain Wall Scattering in Epitaxial hcp Co Films" *Phys. Rev. B* **59**, 11914-11918.
- Savtchenko, L.; Korkin, A. A.; Engel, B. N.; Rizzo, N. D.; Deherrera, M. F.; and Janesky, J. A. 2003. US Patent No. US 6,545,906 **B1**. Apr. 8.
- Steiner, M.; Pels, C.; and Meier, G. 2004. "Reversible and irreversible magnetoresistance of quasi-single domain permalloy microstructures". *J. Appl. Phys.* **95**, 6759.
- Steiner, M.; Pels, C.; Barthelmeß, M.; Kruse, J.; Bolte, M.; Merkt, U.; Meier, G.; Holz, M.; and Pfannkuche, D. 2005. "Magnetotransport measurements and simulations of complex domain patterns in permalloy microstructures". Submitted.
- Weller, D. 2004. "Assault on storage density of 1 Terabit/sq-in and beyond". Spring conference lecture of the *Deutsche Physikalischen Gesellschaft*, Regensburg, Germany.

## AUTHOR BIOGRAPHIES



**MARKUS-A. B. W. BOLTE** is currently working on his Ph.D. in Physics at the University of Hamburg, Germany, on the topic of simulation of micromagnetic structures. Last year, he received his masters in computer science and physics on the same topic.



**GUIDO D. MEIER** is a division leader in the research group of Prof. U. Merkt at the Institute of Applied Physics of the University of Hamburg, Germany. His research activities include semiconductors, ferromagnets and hybrid devices of both materials on the micro- and nanometer scale.



**DIETMAR P. F. MÖLLER** is a professor of computer science and engineering and chair of computer science at the University of Hamburg, Hamburg, Germany. He is currently working on embedded computing systems, virtual and augmented reality, modeling and simulation, mobile autonomous systems, soft computing, medical informatics and nanostructures.

# MODELLING FOR BLUETOOTH PAN RELIABILITY

Xiao Xiong  
John Pollard  
University College London  
Department of Electronic and Electrical Engineering  
Torrington Place, London, WC1E7JE, UK  
Email: jp@ee.ucl.ac.uk

## KEYWORDS

Bluetooth, Personal Area Networks, communication reliability, redundant links, sensors.

## ABSTRACT

Bluetooth-enabled sensors may be used for monitoring personal health but there is considerable interference in the received signal due to multi-path reflections. It is important that communication reliability should be improved in safety-critical data-acquisition applications.

A new topology of the Bluetooth radio system is presented using *cloned* transmitting devices to enhance signals in particular directions. The multi-path interference at the receiver was reduced and this enabled improved signal quality (lower Bit Error Rates) to be achieved. Computer modelling was conducted to evaluate the magnitude of improvement by using *Space/Time/Frequency* redundancy in indoor wireless communications.

## INTRODUCTION

Communication reliability is very important for sets of wireless-enabled sensors that monitor the health of people in their home. Personal safety may be at stake.

A suitable technology is a Bluetooth Personal Area Network (PAN). This is a set of low power, inexpensive, short-range (10-100m) radio transceivers that operate at 2.45 GHz (Bluetooth SIG 2003). A single Master controls no more than seven Slaves in a Piconet and a PAN comprises several communicating Piconets.

Packets of information are transmitted over radio channels that suffer co-channel interference. This is due to multiple paths between source and sink because of the many irregular scatterers (walls or furniture) in an indoor environment. Time-dependent errors appear at the receiver because the communicating devices hop from frequency to frequency with time. Frequency transitions are chosen to change multi-path interference in a non-coherent manner.

The aim of the present work is to reduce the communication errors by adding redundancy in connectivity for the channel. Bluetooth *clones* allow two links between Master (+ Master-Clone) and Slave.

Computer modelling has been used to confirm the observed performance of a single Bluetooth radio link (Pollard and Kontakos 2001). This work is extended to examine the cloned communication system and to determine its improvement in terms of Bit Error Rate.

## CLONE TOPOLOGY

The topology of a cloned Bluetooth system is presented in Figure 1. Master1 and Master2 (Master + Clone) are synchronized by a common clock and communicate with a Slave. Phase shifters are used in order to set different delays of the transmitted signals at the Master end (Brabant 2003).

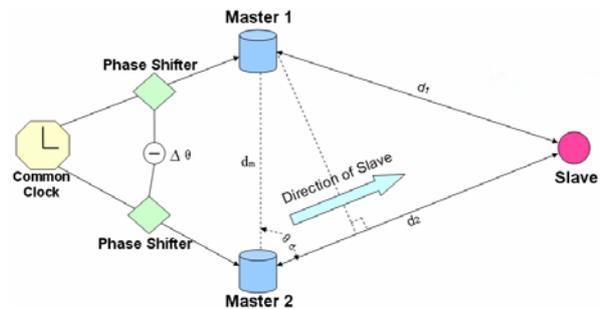


Figure 1: Schematic of Cloned Bluetooth System

The idea of this design is to reduce the effect of reflection in fading channel. In this topology, the original signal is transmitted by two identical Masters and reaches the Slave with varied phase shift as the frequency hops. As the two frequency-hopping signals meet with different reflections while passing through the indoor channels, they do not suffer constant destructive interference. After being received and added at the Slave, the quality of the time-averaged received signal improves.

## MODELLING PROCESS

### Overview

For verifying above proposal, the characteristics of baseband signal throughout the entire transmission process are examined by using computer modelling. A general model is illustrated in Figure 2.

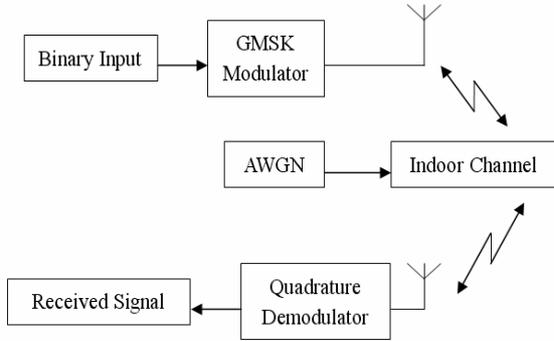


Figure 2: General Communication Model

Maximal length shift-register binary data are up-converted by a Gaussian Minimum-Shift-Key (GMSK) modulator at a prescribed carrier frequency. The pass-band signal then passes through the multi-path indoor channel with Added White Gaussian Noise (AWGN). At the receiver end, a quadrature demodulator is adopted to recover the original baseband signal.

### Modulation

Figure 3 shows a simulation model of GMSK modulator. A Gaussian filter shapes sampled binary data. Samples are fed to Voltage Controlled Oscillator (VCO) where they are integrated and up converted to high frequency band.

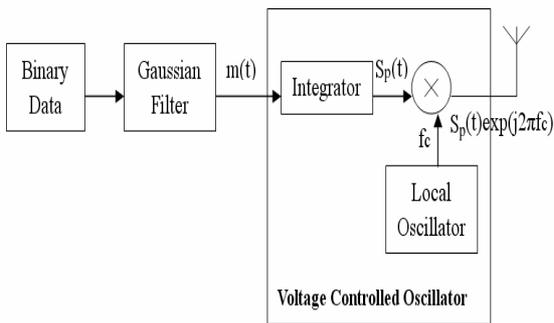


Figure 3: GMSK Modulator

The two signals  $s_1(t)$  and  $s_2(t)$  transmitted from two Masters that are subject to different phase delay can be expressed as  $\text{Re}[\hat{S}(t)\exp(j2\pi f_c t + \phi_1)]$  and  $\text{Re}[\hat{S}(t)\exp(j2\pi f_c t + \phi_2)]$ , respectively.

Continuous-Phase Frequency-Shift Keying (CPFSK) is used as the modulation scheme. The pass-band CPFSK signal,  $S_p(t)$  of a constant envelope modulated signal of carrier frequency,  $f_c$  (2.45 GHz) is:

$$S_p(t) = \text{Re}[S_b(t)\exp(j2\pi f_c t)] \quad (1)$$

$\text{Re}[\ ]$  is the real part of the complex number, and

$S_b(t)$  is the baseband modulated signal:

$$S_b(t) = \cos[\theta(t)] + j\sin[\theta(t)] \quad (2)$$

Here,  $\theta(t)$  is the phase of the signal:

$$\theta(t) = \int m(t)dt \quad (3)$$

where  $m(t)$  is the Gaussian-shaped sampled binary data signal to be transmitted over the channel (Haykin 2000).

### Indoor Channel

Figure 4 shows a statistical model for indoor multi-path propagation (Saleh and Valenzuela 1987). There are three clusters of waves which follow different reflection paths (*multi-path*) illustrated in this figure. These clusters are delayed with respect to the Line-of-Sight (LOS) wave by several nanoseconds.

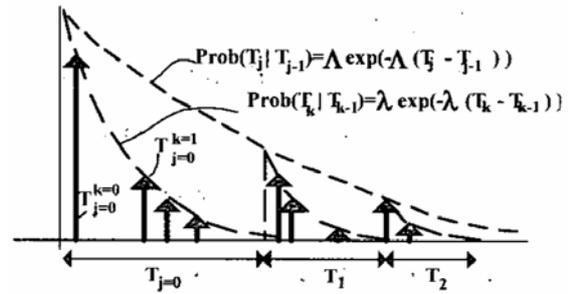


Figure 4: Double-Poisson Arrival Channel Model

The simulation is of the propagation of two signals  $S_{p1}(t)$  and  $S_{p2}(t)$  through two different indoor channels which are based on the model shown in Figure 4. Additionally, AWGN samples are generated using the Box-Muller method (Press et al. 1988).

Figures 5 and 6 are the impulse responses of the two example indoor channels by using an ideal input pulse with unit amplitude. The figures show that the clusters of reflected waves in channel 1 are delayed by a different amount from those of channel 2. This indicates that a large interference does not occur at the same time in the two channels. The channels are decorrelated.

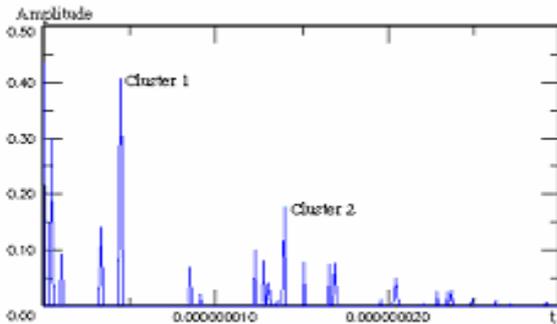


Figure 5: Channel 1 Impulse Response

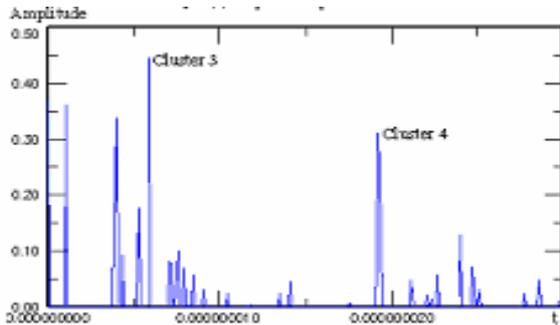


Figure 6: Channel 2 Impulse Response

### Demodulation

Figure 7 is a demodulator that consists of an adder, a quadrature demodulator and a differentiator. The recovered baseband signal  $m(t)$  can be used to examine the transmission characteristics.

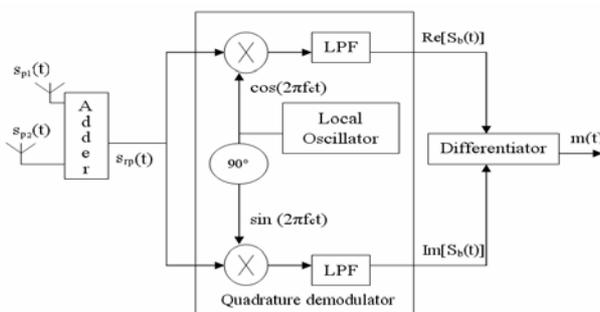


Figure 7: Demodulator Model

Quadrature demodulation is employed to down-convert the pass-band signal to baseband. The baseband signal may be recovered by multiplying the carrier signal and removing the high frequency component. A differentiation with respect to time is then required for

recovering the Gaussian-shaped sampled binary signal,  $m(t)$  (see Equation 3).

### MODELLING RESULTS

Figure 8 is input phase data filtered by Gaussian filter:

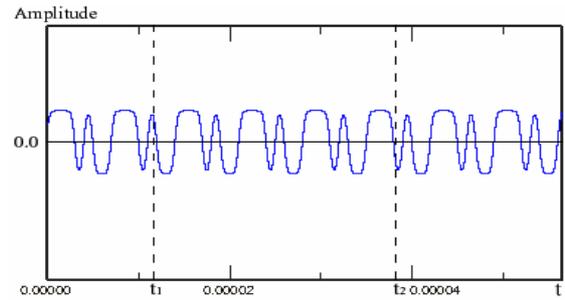


Figure 8: Original Data after Gaussian Filter

After the above simulation procedure, the integrated phase output was obtained that had a similar shape to the input. This is shown in Figures 9 and 10:

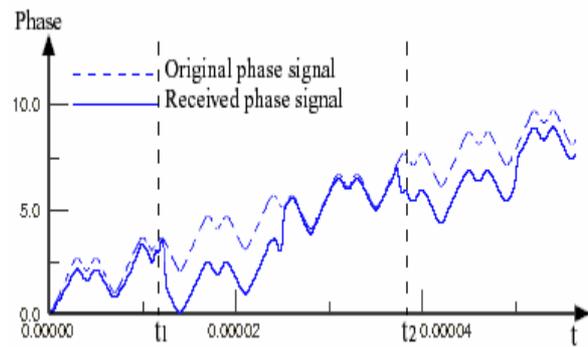


Figure 9: Phase vs. Time for Master Only Data

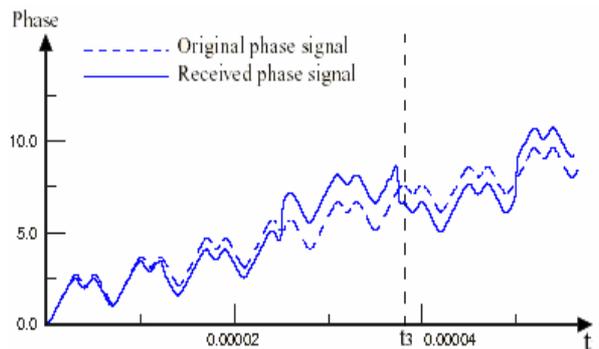


Figure 10: Phase vs. Time for Master+Clone Data

The mean (broken-line) phase signal increases with time as shown and represents the integrated phase signal expected with no effects of the channel. The continuous line is the integrated phase signal after being passed through the indoor channel in each case. Distortion can be seen. This is due to the multi-path reflected waves and causes errors in received data.

The wave form of the single Master signal in Figure 9 has severe distortion points at time  $t_1$  and  $t_2$ . In comparison, the (Master + Clone) signal in Figure 10 is more smooth and more nearly approximates the expected phase signal except for time  $t_3$ .

Figure 11 and Figure 12 show the recovered Gaussian filtered sampled signal after differentiating the waveforms shown above.

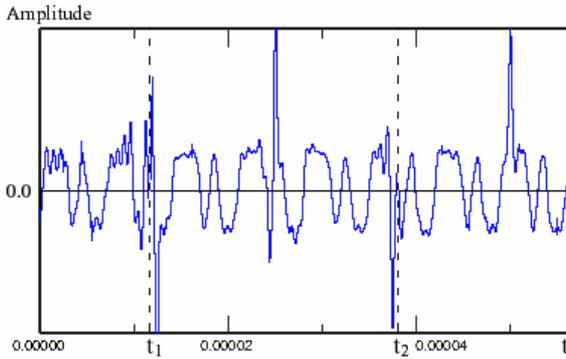


Figure 11: Recovered Signal for Master Only Data

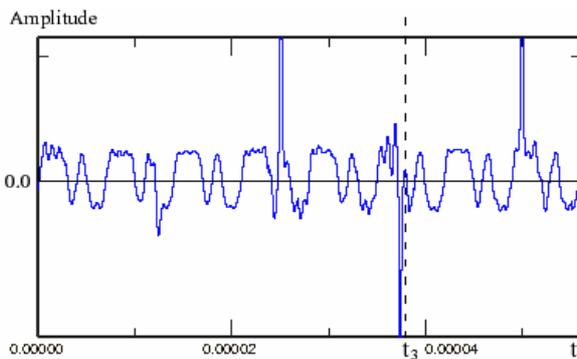


Figure 12: Recovered Signal for Master+Clone Data

The broken line indicates the times ( $t_1$ ,  $t_2$ ,  $t_3$ ) where an error in the output was observed in comparison to the original input data.

The time frame for simulation of the results allowed a very limited snapshot of the phase vs. time. Nevertheless, it can be seen in Figure 12 that only one error is produced in detection at time  $t_3$  within the same measurement period as in Figure 11. It can be inferred that lower Bit Error Rate, BER can be achieved under the (Master + Clone) topology.

The results below further demonstrate the above statement. Figure 13 shows the simulated Bit Error Rate (BER) performance of the single Master topology as the carrier frequency was changed for a constant channel model. It can be seen that the multi-path interference that caused errors was critical in the determination of channel reliability and this was a

strong function of frequency. Between 2.413 and 2.417 GHz (a frequency change of 4 MHz), the BER decreased by nearly an order of magnitude. The reliability of communication had  $BER > 10^{-2}$  for the whole frequency-hopping spectrum for this physical channel. This is satisfactory for speech but barely satisfactory for data transfer for wireless health monitoring.

It can be seen that at carrier frequencies near 2.417 GHz and 2.439 GHz, BERs are as high as  $10^{-1}$ . This means that these frequencies are not available for use in reliable communication systems and an Adaptive Frequency Hop table would be parameterised to preclude them.

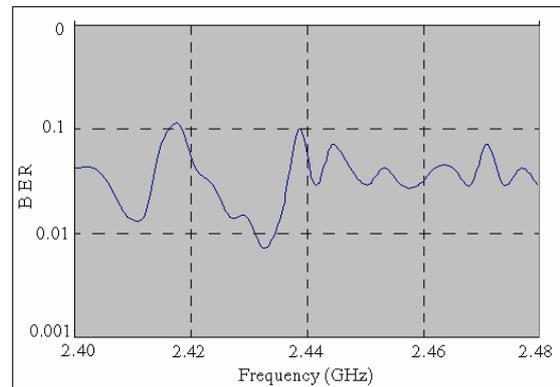


Figure 13: BER vs. Frequency for Single Master

The measured Received Signal Strength Intensity (RSSI) vs. frequency is shown for a single Master topology in Figure 14. There is an inverse relation between RSSI and BER and the measured results bear out the sensitivity of BER simulated results vs. frequency.

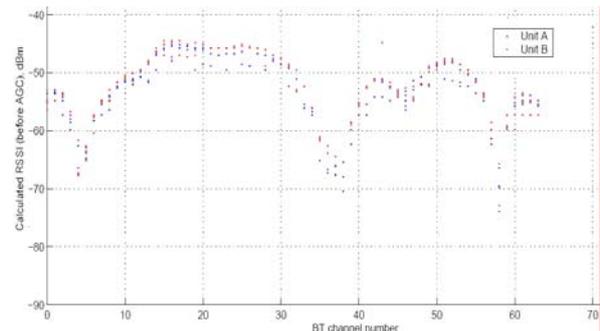


Figure 14: Received Signal Strength vs. Frequency [CSR]

The results from the Bit Error Rate simulation of Single and (Master + Clone) topology are tabulated in Table 1.

Table 1: Comparison of BER Performance in Single and Master + Clone Topology

Hopping Frequency (GHz)	BER			
	Single Master Topology	Cloned Masters Topology		
		Phase Shift		
		0°	90°	180°
2.400	4.29 X10 <sup>-2</sup>	4.29 X10 <sup>-2</sup>	1.43 X10 <sup>-2</sup>	4.29 X10 <sup>-2</sup>
2.418	1.14X10 <sup>-1</sup>	1.43 X10 <sup>-2</sup>	4.29 X10 <sup>-2</sup>	2.86 X10 <sup>-2</sup>
2.439	1X10 <sup>-1</sup>	5.71 X10 <sup>-2</sup>	7.14 X10 <sup>-2</sup>	5.71 X10 <sup>-2</sup>

It can be seen that the BER at frequency 2.400 GHz has been improved from 4.3x10<sup>-2</sup> for a Single Master topology to 1.4x10<sup>-2</sup> for a phase difference of 90° between Master and Clone. In addition, the error rates at frequencies: 2.418 GHz and 2.439 GHz are significantly improved for all phase differences and these two frequencies become usable for speech communication.

Table 1 therefore indicates that more hopping frequencies are available and the communication reliability could be enhanced by using the Master + Clone topology in Bluetooth radio systems.

### SUMMARY

A (Master + Clone) Bluetooth topology was proposed and this has been simulated using a realistic model of the Bluetooth radio system. The characteristics of base-band signal were examined. The initial theory of this topology was verified to be feasible and useful. Improved signal quality (reduced error rate) was capable of being achieved.

### REFERENCES

Bluetooth SIG. 2003. *Specifications of the Bluetooth System. V1.2*, <http://www.bluetooth.com>.

Brabant, G. 2003. *Bluetooth in Space*, MSc. in Space Technology. & Satellite Communications, UCL.

Haykin, S. 2000. *Communication Systems*, John Wiley.

Pollard, J.K. and N.P. Kontakos. 2001. *Bluetooth RF layer performance evaluation*, London Communications Symposium, UCL, 155-158.

Press, W.H. et al.. 1988. *Numerical Recipes in C*, Cambridge University Press.

Saleh, A.A.M. and R.A. Valenzuela. 1987 *A Statistical Model for Indoor Multipath Propagation*, IEEE Journal of Selected Areas of Communications, **SAC-5(2)**, 128-137.

### AUTHOR BIOGRAPHIES



**XIAO XIONG** was born in Wuhan, Hubei province in China. He graduated in Communication Engineering from Huazhong University of Science and Technology, which is one of the best-regarded universities in the country.

He completed a Masters Degree by Research in Telecommunication Engineering at UCL in 2004.



**JOHN K. POLLARD** is in the Department of Electronics and Electrical Engineering at UCL. His background is in the design of Integrated Circuits, communication systems and software systems.

In recent years, he has been interested in the use of the World-Wide Web as an enabling technology for teaching and for distributed software simulation. An integrated combination of hardware and software is necessary to connect a distributed system of computers and mobile input/output electronics (such as mobile telephones and Bluetooth-enabled devices) with databases and real, physical apparatus.

# SIWAPRO DSS: A TOOL FOR COMPUTER AIDED FORECASTS OF LEACHATE CONCENTRATIONS

Oliver Kemmesies

KP Ingenieurgesellschaft für Wasser und Boden mbH  
 Bahnhofstr. 37, D-91710 Gunzenhausen, Germany  
 E-Mail: Oliver.Kemmesies@ibwabo.de

René Blankenburg

Department of waste management and contaminated sites  
 Technical University of Dresden  
 Pratzschwitzer Str. 15, D-01796 Pirna, Germany  
 E-Mail: Rene.Blankenburg@mailbox.tu-dresden.de

## KEYWORDS

Model, leachate concentration, unsaturated zone.

## ABSTRACT

Leachate forecasts are claimed to evaluate the hazard to the groundwater caused by contaminations in the subsurface. It is advisable to employ a numerical model to simulate the complex nature of hydraulics and solute transport in the unsaturated zone. However many parameters are needed to formulate an appropriate simulation model. To minimize the efforts and the costs of exploration, knowledge of the main processes is necessary. Scenario and sensitivity analyses were done to evaluate the influence of each parameter on the results. The analyses showed, that there is no need for transient simulation if the solute is not degradable. On the contrary, if there are organic compounds, the solute concentration is very high subjected to time and also to degradation.

## INTRODUCTION

Leachate forecasts are claimed to evaluate the hazard to the groundwater caused by contaminations in the subsurface. The source is described by a concentration either in the leachate or in the soil vs. time. The hazardous material is subjected to retardation and in some cases also to degradation or decay during its transport as solute through the unsaturated zone to the saturated or groundwater zone.

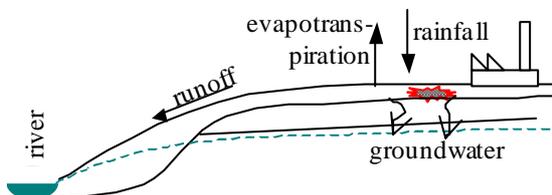


Figure 1: Illustration Figure

Because of the complex nature of soils and their transport properties it is recommended to use computer models for the leachate forecast. But quite a lot of parameters are needed to describe the properties of the source, of the transport through a heterogeneous unsaturated zone and also of the climatical boundary conditions and

the fluctuating groundwater table. The assessment of these parameters is expensive and prone to errors, the uncertainty of the values is very high.

Therefore we need to know, which parameters are really important to the result and need to be assessed most accurately and which parameters do not have such an influence on the calculated concentration in the leachate. In a first step, we took a real case for our scenario-/sensitivity analyses of boundary conditions and parameters. At the chosen site a chromium plating factory was established. The factory started at the end of the sixties and worked until the beginning of the nineties. A massive groundwater contamination with chromium and chromate was detected during the investigations in the nineties. The arising questions were

1. what will be the highest chromate concentration in the leachate
2. how long does it take until the max. concentration reaches the groundwater table
3. which chromium mass will enter the groundwater per year

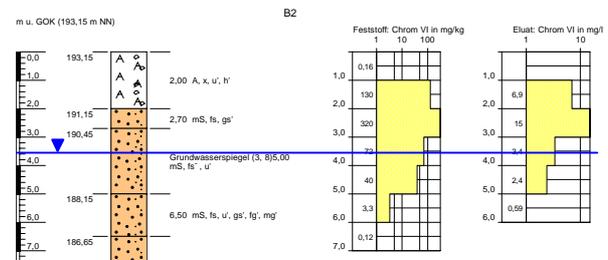


Figure 2: Soil and Concentration Profile at Location B2

## MODELS

The scenario and sensitivity analysis were done using the computer program SiWaPro DSS. The name is the german synonym for Sickerwasserprognose (leachate forecast) Decision Support System. The program is based on the commonly used simulation code SWMS\_2D (Šimunek et al. 1994).

## Flow Model

The flow model describing unsaturated one dimensional vertical water flow in the unsaturated zone is given by RICHARD's-equation (1)

$$\frac{\partial}{\partial z} \left( k(\theta) \cdot \left( \frac{\partial h_p}{\partial z} + 1 \right) \right) = \frac{\partial \theta}{\partial t} - w_0 \quad (1a)$$

and

$$\frac{\partial \theta}{\partial t} = C(h_c) \cdot \frac{\partial h_p}{\partial t} \quad (1b)$$

where the independent variables are time  $t$  and spatial coordinate  $z$ . The dependent variables of equation (1) are the water pressure head  $h_p = p_w / \rho_w \cdot g$  ( $h_c = -h_p$ ) and the water content  $\theta$ .  $w_0$  is the sink/source term. The capillary capacity function  $C(h_c)$  is the first derivative of the hysteretic soil water retention curve drawn in Figure 3. The unsaturated hydraulic conductivity  $k(\theta)$  depends on the water content in the soil.

The hysteretic parametric model of soil water retention curve is given after van Genuchten (1980) and Luckner et al. (1989) by:

$$\theta = A + \frac{\phi - A - B}{\left[ 1 + (\alpha \cdot h_c)^n \right]^{\frac{1}{n}}} \quad (2)$$

The parameters of equation (2) are the porosity  $\phi$ , the residual water content  $\theta_{W,r}$ , the residual air content  $\theta_{A,r}$ , the scaling factor  $\alpha$  and the slope parameter  $n$ . Figure 3 shows a typical curve set for this hysteretic function, where

- |   |     |                         |
|---|-----|-------------------------|
| ① | PDC | Primary Drainage Curve  |
| ② | SWC | Scanning Wetting Curve  |
| ③ | SDC | Scanning Drainage Curve |
| ④ | MWC | Main Wetting Curve      |
| ⑤ | MDC | Main Drainage Curve     |

The function of unsaturated hydraulic conductivity was modeled by Mualem (1976) and Luckner et al. (1989) with

$$k(\theta) = k_0 \cdot \left( \frac{\bar{S}}{\bar{S}_0} \right)^\lambda \cdot \left[ \frac{1 - \left( 1 - \bar{S}^{\frac{1}{m}} \right)^m}{1 - \left( 1 - \bar{S}_0^{\frac{1}{m}} \right)^m} \right]^2 \quad (3)$$

The parameters of equation (3) are the hydraulic conductivity  $k_0(\theta_0)$  at a known degree of water mobility  $\bar{S}_0 = (\theta_0 - \theta_{W,r}) / (\phi - \theta_{W,r})$ , the parameter  $\lambda$  and the transformation parameter  $m$ . The function of unsaturated hydraulic conductivity is shown in Figure 4.

The parameters  $\phi$ ,  $k_0$  and  $\theta_0$  must be estimated in advance using lab and/or field tests. The parameter  $\lambda$  in our model may range between  $0 < \lambda < 1$ , but it is kept fixed at  $\lambda = 0.5$ .

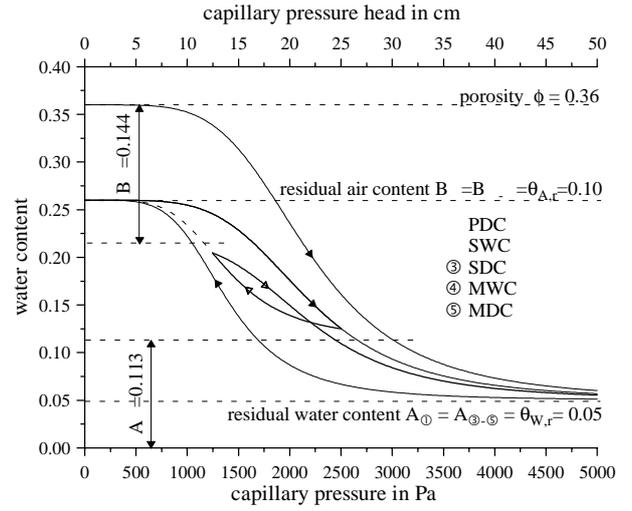


Figure 3: Hysteretic Soil Water Retention Curve

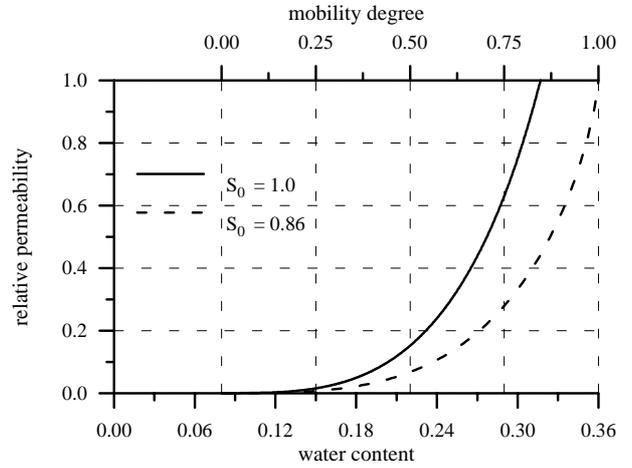


Figure 4: Relative Hydraulic Conductivity Function

### Transport model

The well known convection-dispersion-equation (4) is used to describe the transport processes in the unsaturated zone.

$$\underbrace{\frac{\partial}{\partial r} \left( D \cdot \frac{\partial s_{fl,m}}{\partial r} \right)}_{\text{dispersion}} - \underbrace{\frac{\partial (u \cdot s_{fl,m})}{\partial r}}_{\text{convection}} = \underbrace{\frac{\partial s_m}{\partial t}}_{\text{mass storage changes}} + \underbrace{\mu_m \cdot s_m + \gamma_m \cdot \theta}_{\text{degradation terms}} - \underbrace{\frac{q_m}{V}}_{\text{sinks/sources}} \quad (4)$$

where is

$r$	$m_R$	spatial coordinate
$t$	$s$	time
$\theta$	$m^3 / m^3$	water content
$D$	$m^2 / s$	dispersion coefficient ( $D = \delta \cdot  v $ )
$\delta$	$m$	dispersivity
$s_m$	$kg / m^3$	total specific mass ( $s_m = s_{fl,m} + s_{s,m}$ )

$s_{fl,m}$	$\text{kg} / \text{m}_R^3$	spezific mass in the liquid phase
$s_{s,m}$	$\text{kg} / \text{m}_R^3$	spezific mass in the solid phase
$u$	$\text{m}_R / \text{s}$	mean flux
$\gamma_m$	$\text{kg} / (\text{m}_{fl}^3 \cdot \text{s})$	0 order degradation coefficient
$\mu_m$	$\text{s}^{-1}$	1. order degradation coefficient
$q_m$	$\text{kg} / (\text{m}_R^3 \cdot \text{s})$	sinks/sources

R,fl,B indexes the space, the liquid and and the soil

The convection term describes the solute transport with the water flux in the unsaturated zone. The dispersion term in equation (4) is the sum of the molecular diffusion and the hydrodynamic dispersion. Both processes are caused by concentrations gradient. The hydrodynamical dispersion is always bound to convection, but molecular diffusion is independent from it and may appear without any convection.

The dispersivity is an empirical parameter, it is a measure of heterogeneity of the soil and therefore depends on the scale. The reasons for dispersion are

- different velocities in the pore channels
- different pore sizes and therefore different velocities
- different flow times because of different flow paths
- transversal spreading of particles

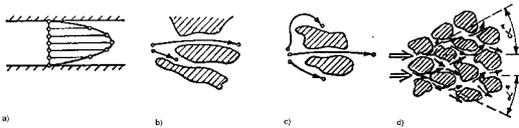


Figure 5: reasons for dispersivity in pore scale (figure taken from Luckner & Shestakov 1991)

Figure 6 shows an example for the results of dispersion in a column experiment. A short input (DIRAC impulse) was adapted on the upper boundary condition.

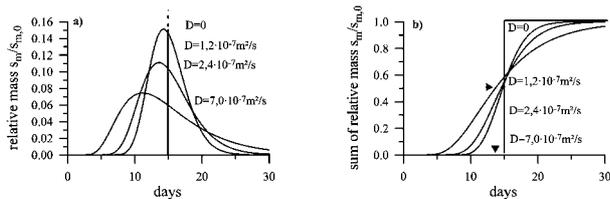


Figure 6: influence of dispersivity on breakthrough curves

The reason for retardation is sorption. The parameter describing the linear distribution function between the solute in the liquid and at the solid phase is the distribution or HENRY-coefficient  $K_D$ . The distribution and the retardation coefficient  $R$  are related to each other through equation (5). figure 7 shows the influence of retardation of the breakthrough curve at the bottom of a column experiment.

$$K_d = (R - 1) \cdot \frac{\theta}{\rho_b} \quad (5)$$

where  $\rho_b$  is the bulk density of the soil.

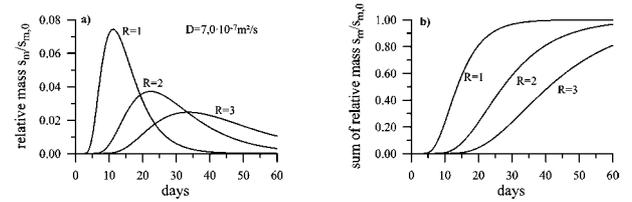


figure 7: influence of retardation/sorption on an impulse on top of a column

Internal reactions (decay and/or degradation) may be described as zero or first order process as shown in figure 8.

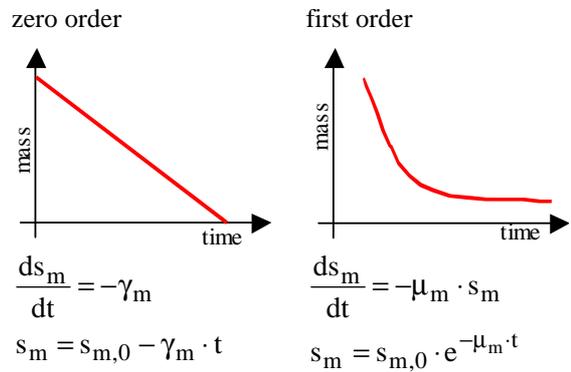


figure 8: order and description of degradation processes

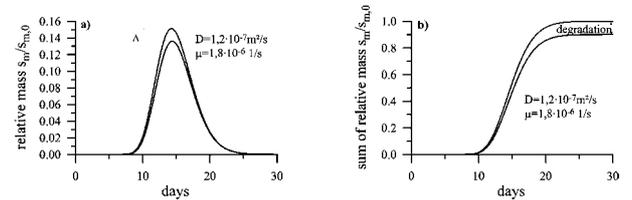


Figure 9: influence of degradation on breakthrough curves

## BOUNDARY CONDITIONS AND PARAMETERS

### Flow Model

The column has a height of 3.80 m, which corresponds to the groundwater table taken from Figure 2. The groundwater is the lower boundary condition, it is a first kind boundary condition of eq. (1). The flux difference between precipitation and evapotranspiration is applied as second type boundary condition at the top of the column. This is actually a transient boundary condition as shown in Figure 10.

Using transient flow conditions results in a very large the computig time. Therefore it is a common simplification to assume steady-state flow conditions with a mean recharge rate. But a proof is needed, if there are no differences in the results.

The space discetization of the column was held constant with  $dz=0.01$  m. Each of the soil layers needs to be de-

scribed in its hydraulics with 5 parameters. Their values were taken from the field soil description and available soil data bases, like UNSODA (Schaap et al. 1999). The chosen parameter values are listed in Table 1.

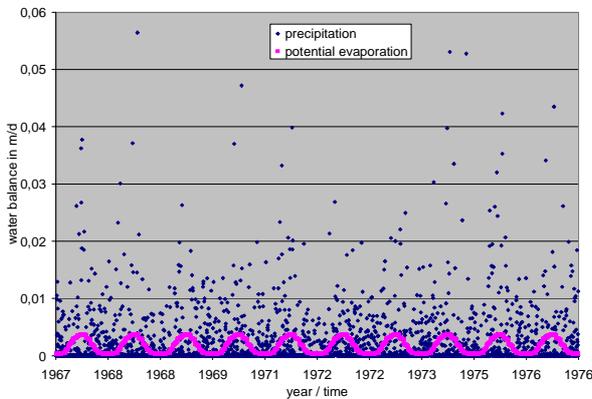


Figure 10: Precipitation and Potential Evaporation vs. Time

Table 1: Soil Hydraulic Parameters

parameter	symbol	dimen- sion	infilling	coarse middle sand	silty middle sand
porosity	$\phi$		0.36	0.35	0.38
residual wa- ter conetent	$\theta_{w,r}$		0.09	0.07	0.1
scale factor	$\alpha$	1/m	0.8	1.0	1.0
slope factor	$n$		2.0	2.0	2.0
hydraulic conductivity	$k_f$	m/s	$1 \cdot 10^{-4}$	$1 \cdot 10^{-3}$	$5 \cdot 10^{-4}$
bulk density	$\rho$	kg/m <sup>3</sup>	1.700	1.700	1.700
dispersivity	$\delta$	m	0.5·GWFA		

legend: GWFA – groundwater level below surface

### Transport Model

The source chromate concentration at the top of column was chosen to 30 mg/l, starting from  $t=0$  to  $t=20$  a. The remaining parameter is the distribution coefficient to describe the sorption and retardation of chromate in the unsaturated zone. More than 42 drillings were done at the site. From these 42 drilling more than 100 samples were taken and analyzed for their chromium and chromate content in the soil and in the leachate. So it should not be a problem to establish a relation between the chromate content in the soil and in the liquid. But there is still a lack of methods and understanding. Right now it is known, that the measured chromate content in the liquid is not to compare with an actual leachate concentration in the field scale. So we had to do, what we always do in such cases, we searched in the literature for similar cases and found some hints for chromate retardation coefficients. From there we calculated the distribution coefficients using eq. (5) to  $K_D=1,3 \cdot 10^{-3} \text{ m}^3/\text{kg}$ ,  $R \approx 6$ .

### SCENARIOS

The defined cases distinguish between simplification of basic equations and parameter models and sensitivity calculations for parameter values.

Case 1 is the basic case. It is defined as a transient simulation of water flow and solute transport in the unsaturated zone for our contaminated site.

Commonly used simplifications of equation (1) are

case 2: the neglectation of storage terms, which results in steady-state conditions with a mean groundwater recharge rate as given flux instead of a transient flux

case 3: gravity driven flow processes, this case is actually not physically based and will not be regarded any further

A further commonly used simplification of the model of the subsurface is

case 4: a homogeneous unsaturated zone

The cases for sensitivity analysis are defined as

case 5: variation of the leachate rate (=groundwater recharge rate) in the range  $v_N=178 \text{ mm/a} \pm 20\%$

case 6: variation of the saturated hydraulic conductivity in the range  $5 \cdot 10^{-4} \leq 1 \cdot 10^{-3} \leq 5 \cdot 10^{-3} \text{ m/s}$

case 7: variation of the scale factor  $\alpha$  from eq. (2) in the range  $0.5 \leq 1.0 \leq 2.0 \text{ 1/m}$

case 8: variation of the slope factor  $n$  from eq. (2) in the range  $1.5 \leq 2.0 \leq 3.0$

case 9: variation of the residual water content  $\theta_r$  from eq. (2) and (3) in the range  $0 \leq 0.07 \leq 0.1$

case 10: variation of the porosity  $\phi$  from eq. (2), (3) and (5) in the range  $0.35 \pm 0.05$

case 11: variation of the sorption coefficient  $K_D$  from eq. (4) and (5) in the range  $5 \cdot 10^{-4} \leq 1 \cdot 10^{-3} \leq 5 \cdot 10^{-3} \text{ m}^3/\text{kg}$

case 12: variation of the source concentration  $c_0$  in the range  $25 \leq 30 \leq 50 \text{ mg/l}$

case 13: variation of the dispersivity  $\delta$  from eq. (4) in the range  $0.5 \leq 1.0 \leq 2.0 \text{ m}$

The case 14 “variation of degradation rate  $\mu_n$  from eq. (4)” could not be analyzed, because chromium/chromate is not degradable.

Please note there are no cases defined to investigate simplifications of the convection-dispersion-equation (2). There must not be any simplification, because all of incorporated processes must be taken into account to get an appropriate concentration forecast.

Case 2 is the most commonly used simplification for leachate forecasts. The results of all other cases will be compared with the results of case 2.

The flow problem is solved if the recharge rate at the top of column is given (=case 2). Therefore one may not expect, that the results of cases 6-10 will differ significantly to the result of case 2. The mean recharge rate is given with  $0.178 \text{ m/a} = 5.6 \cdot 10^{-9} \text{ m/s}$ . This value is as the functions in Figure 11 show much more smaller than the

saturated and also the unsaturated hydraulic conductivity of used materials. The water content of these materials is always greater than the residual water content, therefore the given recharge rate can be flow through the unsaturated zone without any resistance. Please note, that we have much more higher recharge rates under transient conditions. They may be greater than the saturated hydraulic conductivity. In this cases runoff occurs at the surface, the soil is saturated and not able to receive more water. The storage capacity (represented with the parameters  $\phi$ ,  $\theta_{w,r}$ ,  $\alpha$  and  $n$ , cases 7 - 10), is neglected anyway. But there might be a small influence of the parameters  $\alpha$  and  $n$  because of the incorporation of the water content in eq. (4). This may influence the migration velocity too. This statement also meets the parameters  $\phi$ ,  $\theta_{w,r}$ , which have an influence to eq. (2) via the calculation of the water content from eq. (1b).

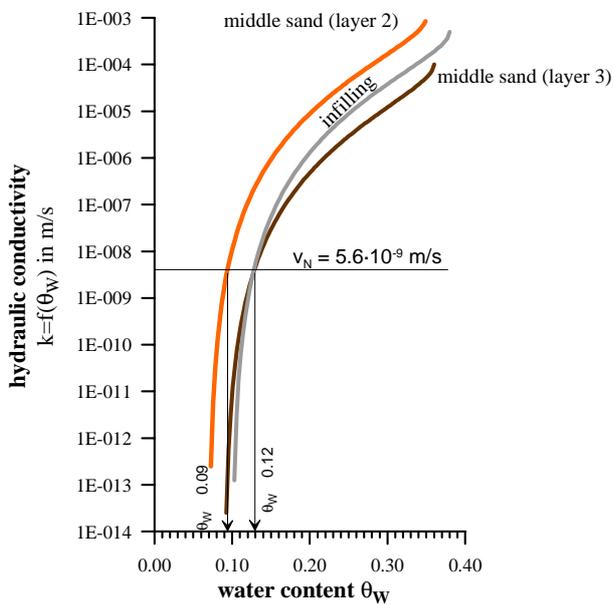


Figure 11: hydraulic conductivity function of used materials

## RESULTS

The computational results of the basic case 1 are shown in Figure 12. The highest chromate concentration in the solute will be about 25 mg/l at  $t=24$  a. This results meets the observations, which were done in the 90-ies. The concentration descends after the peak went through, because the source was removed in the beginning of the 90-ies. Within the regarded 100 years about 110 g/m<sup>2</sup> chromate will reach the groundwater.

The highest yield occurs after 22 years with 105 g/a. This amount is caused by a high but shortterm flux rate. The results for all other cases are listed in Table 2. The columns contain

1. the highest chromate-concentration  $c_{max}$ ,
2. the time of its occurrence,
3. the amount of highest yield  $Y_{max}$ ,

4. the time of its occurrence and
5. the total chromate mass within 100a, reaching the groundwater per m<sup>2</sup>.

The comparisoin of results from case 1 to case 2, shows that there is no need for transient flow conditions, the calculated concentration, yield, mass and times of occurences are always close together or identical.

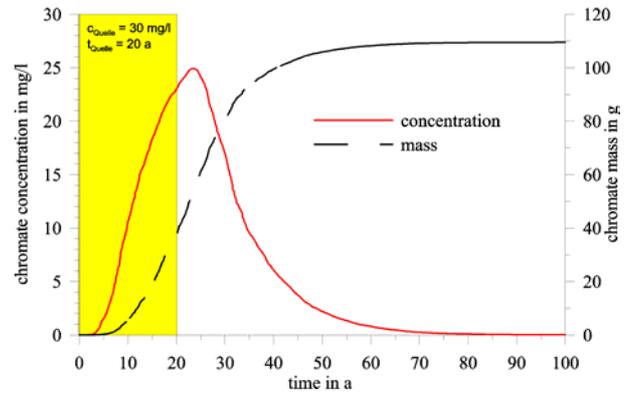


Figure 12: calculated chromate concentration and sum of chromate mass vs. at the bottom of the column

Table 2: Calculated Results

case	concentration		yield		sum of mass
	$c_{max}$	year	$Y_{max}$	year	m
	mg/l	a	mg/a	a	g
1	24.9	24	105000	22	110
2	24.5	24	4242	24	105
3	1.5	100	269	100	7
4a	24.5	24	4238	24	105
4b	24.6	24	4263	24	105
4c	24.5	24	4235	24	105
5a	26.4	23	5481	23	125
5b	22.1	25	3053	25	85
6a	24.7	24	4263	24	105
6b	24.7	24	4263	24	105
7a	25.0	24	4302	24	105
7b	24.4	24	4225	24	105
8a	24.9	24	4291	24	105
8b	24.5	24	4238	24	105
9a	24.6	24	4253	24	105
9b	24.8	24	4288	24	105
10a	24.5	24	4235	24	105
10b	24.8	24	4291	24	105
11a	17.0	30	2935	30	> 105
11b	28.8	21	4982	21	105
12a	32.9	24	5684	24	140
12b	16.4	24	2842	24	70
13a	26.5	22	4581	22	105
13b	24.7	27	4262	27	105

The case 3 is as mentioned above an exception. It is not acceptatable to reduced the flow conditions to a gravity

driven flow by a given first flux boundary condition at the bottom.

The cases 4a, b and c are defined by a homogeneous soil column. The results show, that there is also no significant difference to the results of case 2. The soil hydraulic parameters  $k_f$  (case 6), the van-Genuchten-Parameters  $\alpha$ ,  $n$  (case 7, 8), the porosity  $\phi$  and residual water content  $\theta_{w,r}$  (case 9, 10) also do not have too much influence on the results. Significant differences are calculated with slight changes in the mean groundwater recharge rate and all transport parameters.

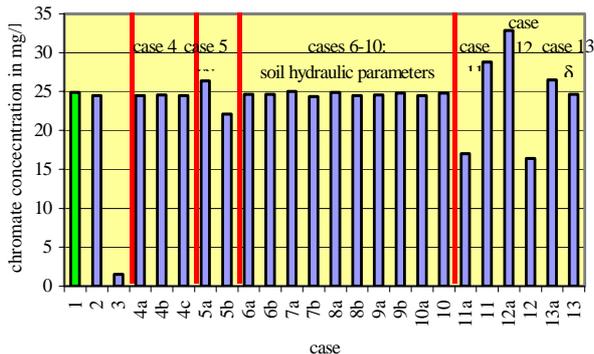


Figure 13: calculated concentrations at the bottom of the column

## CONCLUSIONS

The results of the scenario and sensitivity analysis showed the thesis, that

- ✓ it is acceptable to assume steady state flux conditions over a long term period instead of using a transient precipitation boundary condition
- ✓ the soil hydraulic parameters do not influence the calculated concentration at the bottom of a column, if their order of magnitude was quite real estimated

During parameter identification the

- ✓ groundwater recharge rate
- ✓ source concentration and
- ✓ distribution coefficient

must be determined as accurately as possible.

But these statements are only valid, if the hazard is not degradable. Further investigations will be done on organic contaminations like a fuel depot. First calculations have shown, that the calculated concentration is highly related to the given transient flux conditions due to the degradation. A systematization scheme must be developed.

## REFERENCES

- Luckner, L. and W.M. Shestakov. (1991). Migration Processes in the Soil and Groundwater Zone. Lewis Publishers, Inc. 485 p.
- Luckner, L., M. Th. van Genuchten, D. R. Nielsen. (1989). "A Consistent Set of Parametric Models for the Two-Phase

Flow of Immiscible Fluids in the Subsurface", *Water Resour. Res.* 25, 2187-2193.

Mualem, Y (1976). "A New Model for Predicting the Hydraulic Conductivity of Unsaturated Porous Media". *Water Resour. Res.* 12, 513-522.

Schaap, M. G., F. J. Leij and M. Th. Van Genuchten. (1999) „Bootstrap and Neuronal Network Model“, In: M. Th. Van Genuchten, F. J. Leij and L. Wu (eds.): Proceedings of the International Workshop on Characterization and Measurement of the Hydraulic Properties of Unsaturated Porous Media, Part 2, University of California, Riverside, 1999 p. 1237-1250.

Šimunek J., T. Vogel and M. Th. van Genuchten. (1994). "The SWMS\_2D code for simulating water flow and solute transport in two-dimensional variably saturated media", Version 1.1. , Research Report No.132, U. S. Salinity Laboratory, USDA, ARS, Riverside, CA.

van Genuchten, M.Th. (1980). „A Closed-form Equation for Predicting the Hydraulic Conductivity of Unsaturated Soils“, *Soil Sci. Soc. Am. J.* 44. 892-898.

## ACKNOWLEDGEMENTS

This research work was granted by the German Department of Education and Science (BMBF-Bundesministerium für Bildung und Forschung) contract # 02WP 0192, 0242, 0502 and 0503.



**OLIVER KEMMESIES** was born in Leipzig, Germany and went to the Technical University of Dresden, where he studied civil engineering and obtained his degree in 1991. He did his Ph.D. studies with the Dresden Groundwater Research Centre and obtained his doctor degree in 1995 from the Technical University Mining Academy Freiberg. He worked for a couple of years in a larger consulting company as head of the department "Numerics in water management", before he founded his own consulting company for water and soil exploration in 1999. His work is mainly subjected to modeling of unsaturated and groundwater flow as well as fate transport in the underground. His e-mail address is: Oliver.Kemmesies@ibwabo.de and his Web-page can be found at <http://www.ibwabo.de>.



**RENÉ BLANKENBURG** was born in Soemmerda, Germany. For his studies in geodesy he went to the Technical University Dresden where he obtained the degree 'Dipl.-Ing.' in 2004. Since August 2004 René Blankenburg is employed at the institute of Waste Management and Contaminated Site Treatment at the TU Dresden. His task is the further development of the existing computer software SiWaPro-DSS, a decision support system for seepage prognosis. His e-mail address is: rene.blankenburg@mailbox.tu-dresden.de and his Web-page can be found at <http://www.tu-dresden.de/fghh1aa/>.

# FROM STEADY-STATE AND DYNAMIC ANALYSIS TO ADAPTIVE CONTROL OF THE CSTR REACTOR

Jiri Vojtesek and Petr Dostal  
Department of Process Control  
Institute of Process Control and Applied Informatics  
Tomas Bata University in Zlin  
Nam. T.G.M. 275, 762 72 Zlin, Czech Republic  
E-mail: vojtesek@ft.utb.cz, dostalp@ft.utb.cz

## KEYWORDS

Steady-state analysis, dynamic analysis, CSTR, adaptive control, polynomial synthesis, pole-placement method

## ABSTRACT

Simulation is the technical discipline which shows the behavior and reactions of any system on its model. Most processes in nature have nonlinear properties and simulation is one way to examine behaviour of these systems. The behaviour is then obtained by steady-state and dynamic analysis of the model which is usually represented by a set of differential equations. The next step after dynamic analysis is choosing a suitable control strategy and finally designing a controller. The paper presents the progress from modeling and simulation of a nonlinear process represented by a continuous stirred tank reactor (CSTR) to adaptive control of the system based on polynomial synthesis with pole-placement method.

## INTRODUCTION

Computer simulation is very often used at present as it has advantages over an experiment on a real system, which is sometimes not feasible and can be dangerous, or time and money demanding. A large amount of processes in chemical industry is nonlinear and computer simulation can help us to give us an insight into the system behaviour. The first step is usually introduction of a mathematical model (Ingham et al. 2000, Luyben 1989). Mathematical model can be described by a set of linear, nonlinear or differential equations.

A Continuous Stirred Tank Reactor (CSTR) is widely used for control because input flow of the reactant or cooling liquid can be easily controlled. From the system engineering point of view, CSTR belongs to the class of nonlinear systems with lumped parameters. Mathematical models of these reactors are described by a set of nonlinear ordinary differential equations (ODEs). In this paper the simple differential method (Luyben 1989) and the standard Runge-Kutta's method are used to solve the set of ODEs.

Simulation results are then used for control, in our case adaptive control (Åström 1989). The first step here is to

find an appropriate control method. The polynomial approach (Kucera 1993) has satisfied control requirements and moreover, it could be used for systems with negative properties such as non-minimum phase behaviour and processes with time delays. This method connected with pole-placement fulfills stability, asymptotic tracking of the reference signal and compensation of disturbances.

Two control system configurations (Grimble 1994) were used – the first with one degree-of-freedom (1DOF) which has regulator only in feedback part and the second with two degrees-of-freedom (2DOF) with feedback and feedforward parts. Adaptivity was ensured by estimating model parameters with recursive least-squares method with directional forgetting (Kulhavy and Karny 1984). Since the model is nonlinear, an external linear continuous-time model (Wahlberg 1990) was used as an input-output description of the controlled system.

The goal of this work is to describe the progress from simulation of the behaviour to control of a nonlinear process represented by the CSTR reactor taken from (Chen *et al.* 1995).

## THEORETICAL BACKGROUND

As it is written above, the goal of this paper is to describe the process from modeling and simulation to control of a dynamic process. This can be expressed for example as in Figure 1. There are four main steps connected mutually together.

### 1. Define goals

It is necessary to define goals of simulation study including collection of all the available knowledge and practical experience with the investigated type of the plant. Unfortunately, a major group of systems in nature are nonlinear or number of variables is very high. This complexity leads to introduction of some simplifications.

### 2. Modeling

Modeling is mostly the hardest step. A process is described by its characteristic variables (temperature, concentration, flow rate, pressure etc.) usually called *state variables* and mathematical relations between

these quantities then form a *mathematical model*. The mathematical model comes from balances inside the plant and could be expressed by a set of linear or nonlinear equations, ordinary or partial differential equations etc.

### 3. Simulation and validity of the model

Simulation experiment results are good ways how to test suitability of the obtained mathematical model. They usually consist of two parts – static and dynamic analysis. The static analysis results in appropriate working point. On the other hand, the dynamic analysis provides step, frequency responses etc. which display dynamic behaviour of the system and they are a base for choosing an external linear model. Comparison between outputs from real system and simulated outputs demonstrate validity of mathematical model. If the difference between the real system and the model is unacceptable, it is necessary to jump back to modeling and cancel some simplifications. The goal is to find the simplest model with satisfactory description of the real process.

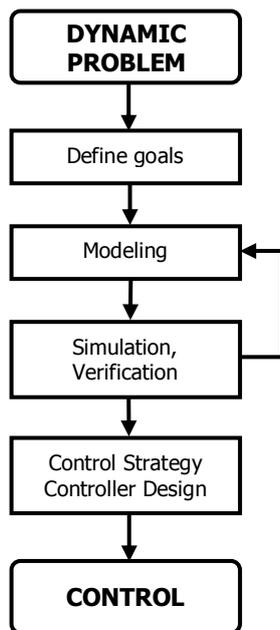


Figure 1: Steps from simulation to control of a dynamic system

### 4. Control strategy and controller design

The next step after simulation is to look on the problem from the control point of view. That means to find a regulator controlling our dynamic process in a defined way. The obtained regulator must e.g. fulfill requirements of stability, properness, asymptotic tracking of the reference or disturbance attenuation. There are lot of ways for designing such a controller. Configurations with one degree-of-freedom (1DOF) and two degrees-of-freedom (2DOF) and their combinations with polynomial methods satisfy control conditions

mentioned above. As it was shown (Dostal et al. 2002), these methods can be applied on systems with negative control properties, such as instability, nonminimum phase or on systems with transport delays. Obtained regulators are then subject to simulation experiments and if results are acceptable, they can be used to control dynamic problem.

### APPLICATION ON REAL SYSTEM

The procedure which is described in previous chapter was applied on real nonlinear process represented by a Continuous Stirred Tank Reactor (CSTR).

#### 1. Define goals

The first goal of the simulation study is to find the best working-point for control study, where the production is best balanced – i.e. obtain the biggest production with the smallest energy usage. An External Linear Model (ELM) is the second goal and we obtain it from dynamic study. We presume that both reactant and cooling liquid are perfectly mixed and volumes, densities, heat capacities of reactant and cooling liquid are supposed to be constant.

#### 2. Modeling

The examined reactor has real background and graphical diagram of the CSTR reactor is shown in Figure 2.

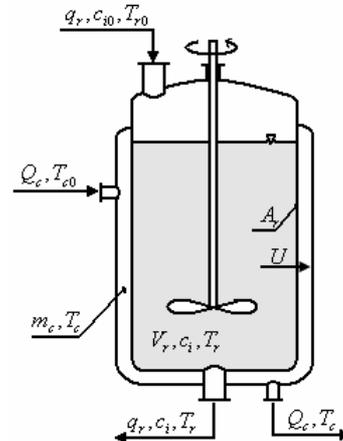
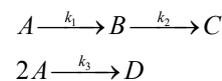


Figure 2: Continuous Stirred Tank Reactor (CSTR)

The reaction inside the reactor is called *van der Vusse reaction*. This reaction can be described by the following scheme:



The mathematical model of this reactor comes from balances inside the reactor. The first two equations come from material balances of compounds A and B:

$$\frac{dc_A}{dt} = \frac{q_r}{V_r}(c_{A0} - c_A) - k_1 c_A - k_3 c_A^2 \quad (1)$$

$$\frac{dc_B}{dt} = -\frac{q_r}{V_r} c_B + k_1 c_A - k_2 c_B \quad (2)$$

Where  $c_A \geq 0$ ,  $c_B \geq 0$ . The next balances represent heat balances of the reactive compound and cooling liquid

$$\frac{dT_r}{dt} = \frac{q_r}{V_r}(T_{r0} - T_r) - \frac{h_r}{\rho_r c_{pr}} + \frac{A_r U}{V_r \rho_r c_{pr}}(T_c - T_r) \quad (3)$$

$$\frac{dT_c}{dt} = \frac{1}{m_c c_{pc}}(Q_c + A_r U(T_r - T_c)) \quad (4)$$

In equations (1) – (4)  $t$  is the time,  $c$  are concentrations,  $T$  represents temperatures,  $c_p$  is used for specific heat capacities,  $q$  represents volumetric flow rate,  $Q_c$  is heat removal,  $V$  are volumes,  $\rho$  represents densities,  $A_r$  is the heat exchange surface and  $U$  is the heat transfer coefficient. Indexes  $(\cdot)_A$  and  $(\cdot)_B$  belong to compounds A and B,  $(\cdot)_r$  denotes the reactant mixture,  $(\cdot)_c$  cooling liquid and  $(\cdot)_0$  are feed (inlet) values.

Table 1: Parameters of the reactor

$k_{01} = 2.145 \cdot 10^{10} \text{ min}^{-1}$	$k_{02} = 2.145 \cdot 10^{10} \text{ min}^{-1}$
$k_{03} = 1.5072 \cdot 10^8 \text{ min}^{-1} \text{ mol}^{-1}$	$E_1/R = 9758.3 \text{ K}$
$E_2/R = 9758.3 \text{ K}$	$E_3/R = 8560 \text{ K}$
$h_1 = -4200 \text{ kJ.kmol}^{-1}$	$h_2 = 11000 \text{ kJ.kmol}^{-1}$
$h_3 = 41850 \text{ kJ.kmol}^{-1}$	
$V_r = 0.01 \text{ m}^3$	$\rho_r = 934.2 \text{ kg.m}^{-3}$
$c_{pr} = 3.01 \text{ kJ.kg}^{-1} \cdot \text{K}^{-1}$	$c_{pc} = 2.0 \text{ kJ.kg}^{-1} \cdot \text{K}^{-1}$
$U = 67.2 \text{ kJ.min}^{-1} \text{ m}^{-2} \text{ K}^{-1}$	$A_r = 0.215 \text{ m}^2$
$c_{A0} = 5.1 \text{ kmol.m}^{-3}$	$c_{B0} = 0 \text{ kmol.m}^{-3}$
$T_{r0} = 387.05 \text{ K}$	$m_c = 5 \text{ kg}$

The set of ordinary differential equations (ODE) then together with simplifications described in the previous part then mathematically represents examined CSTR reactor. The model of the reactor belongs to the class of *lumped-parameter nonlinear systems*. Nonlinearity can be found in reaction rates ( $k_j$ ) which are described via Arrhenius law:

$$k_j(T_r) = k_{0j} \cdot \exp\left(\frac{-E_j}{RT_r}\right), \text{ for } j = 1, 2, 3$$

where  $k_0$  represent pre-exponential factors and  $E$  are activation energies.

The reaction heat ( $h_r$ ) in equation (3) is expressed as:

$$h_r = h_1 \cdot k_1 \cdot c_A + h_2 \cdot k_2 \cdot c_B + h_3 \cdot k_3 \cdot c_A^2$$

where  $h_i$  means reaction enthalpies.

Parameters of the reactor are given in Table 1 (Chen *et al.* 1995).

### 3. Simulation and validity of the model

Two simulation studies were performed – steady-state analysis and dynamic analysis.

#### Steady-state analysis

Steady-state analysis for stable systems involves computing values of state variables in time  $t \rightarrow \infty$ , when changes of these variables are equal to zero. That means, that the set of ODEs (1) – (4) is solved with the condition  $\partial(\cdot)/\partial t = 0$ . A simple iteration method was used to solve this problem. Steady values of quantities were examined for various rates of the volumetric flow of the reactant,  $q_r$ , and heat removal of the cooling liquid,  $Q_c$ . The first study for volumetric flow rate of the reactant  $q_r [\text{m}^3 \cdot \text{min}^{-1}] = \langle 5 \cdot 10^{-4}, 3 \cdot 10^{-2} \rangle$  and heat removal  $Q_c = -18.56 \text{ kJ} \cdot \text{min}^{-1}$  displayed in Figure 3 shows strong nonlinear course of steady-state values of both concentrations  $c_A$  and  $c_B$ . Maximal concentration of the product,  $c_B = 1.09 \text{ kmol} \cdot \text{m}^{-3}$ , was obtained for flow rate  $q_r = 2.365 \cdot 10^{-3} \text{ m}^3 \cdot \text{min}^{-1}$ . Curves of steady-state temperatures  $T_r$  and  $T_c$  are displayed in Figure 4.

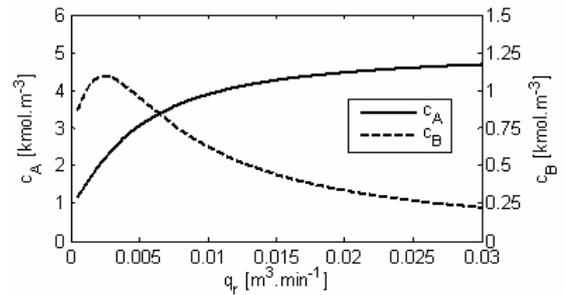


Figure 3: Steady-state values of concentrations  $c_A$  and  $c_B$  for various flow rate  $q_r$ .

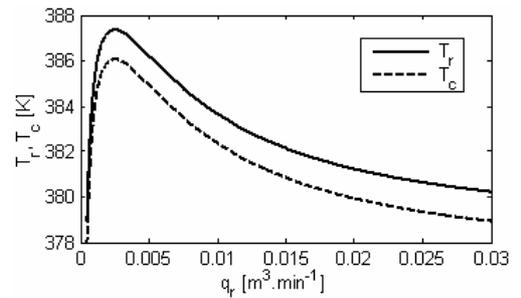


Figure 4: Steady-state values of temperatures  $T_r$  and  $T_c$  for various flow rate  $q_r$ .

Simulation for different heat removal of cooling liquid  $Q_c [\text{kJ} \cdot \text{min}^{-1}] = \langle -500; 500 \rangle$  and flow rate  $q_r = 2.365 \cdot 10^{-3} \text{ m}^3 \cdot \text{min}^{-1}$  has similar nonlinear behaviour as in the previous study – see Figure 5. This course has again a maximal value of product's concentration,  $c_B$ , in this case, the maximal concentration is achieved approximately for heat removal  $Q_c = -18.56 \text{ kJ} \cdot \text{min}^{-1}$ . Figure 6 shows steady values of temperatures  $T_r$  and  $T_c$ .

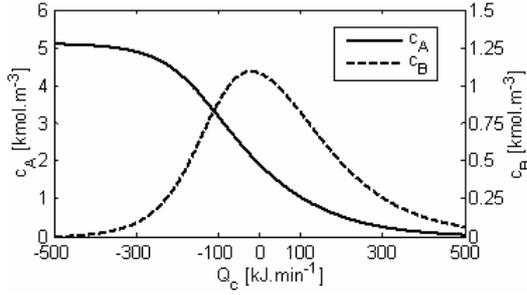


Figure 5: Steady-state values of concentrations  $c_A$  and  $c_B$  for various heat removal  $Q_c$

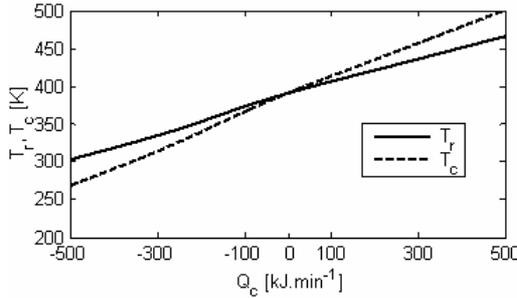


Figure 6: Steady-state values of temperatures  $T_r$  and  $T_c$  for various heat removal  $Q_c$

The goal of the steady-state study is to find an optimal working point, where the product's concentration is maximal and this point was found for volumetric flow rate  $q_r^s = 2.365 \cdot 10^{-3} \text{ m}^3 \cdot \text{min}^{-1}$  and heat removal  $Q_c^s = -18.56 \text{ kJ} \cdot \text{min}^{-1}$ . Steady values of the quantities are following

$$\begin{aligned} c_A^s &= 2.1403 \text{ kmol} \cdot \text{m}^{-3} & c_B^s &= 1.0903 \text{ kmol} \cdot \text{m}^{-3} \\ T_r^s &= 387.3397 \text{ K} & T_c^s &= 386.0551 \text{ K} \end{aligned} \quad (5)$$

#### Dynamic analysis

The next step after steady-state study is dynamic study. For this nonlinear lumped-parameter system dynamic analysis involves solving the set of nonlinear ODE. Computed steady-state values of (5) are used as an input for the dynamic study and Runge-Kutta's standard method with a fixed step was used for solving equations (1) to (4). The behaviour of the system was obtained after a step change of the input quantities  $\Delta q_r$  and  $\Delta Q_c$ . Simulation took 30 minutes and the integration step was set to 0.1 min. Output step responses of the values  $y_1$  and  $y_2$  in next figures illustrate the difference of variables  $c_B$  and  $T_r$  from their steady state values  $c_B^s$  and  $T_r^s$ , i.e.

$$y_1 = c_B - c_B^s; \quad y_2 = T_r - T_r^s$$

The first study for four values of step changes of volumetric flow rate  $\Delta q_r [\text{m}^3 \cdot \text{min}^{-1}] = -1.18 \cdot 10^{-3}$  (-50% of its steady-state value  $q_r^s$ ),  $-0.59 \cdot 10^{-3}$  (-25%),  $0.59 \cdot 10^{-3}$  (25%),  $1.18 \cdot 10^{-3}$  (50%) shows that both outputs  $y_1$  and  $y_2$  has negative properties from the control point of view – non-minimum phase behaviour and the

changing sign of gain. Step responses of these output variables are displayed in Figure 7 and Figure 8.

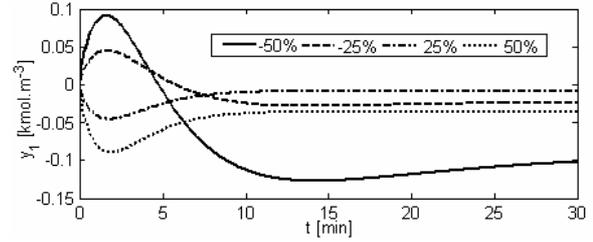


Figure 7: Course of output concentration  $c_B$  ( $y_1$ ) for various step changes of flow rate  $\Delta q_r$

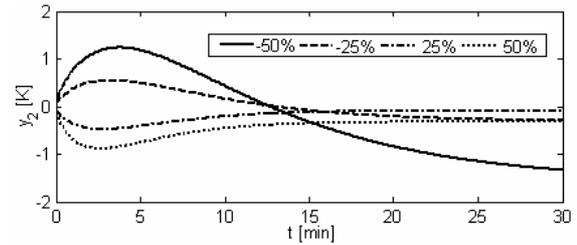


Figure 8: Course of output temperature  $T_r$  ( $y_2$ ) for various step changes of flow rate  $\Delta q_r$

Figure 9 shows step responses for different step changes of cooling heat removal  $\Delta Q_c [\text{kJ} \cdot \text{min}^{-1}] = 9.28$  (-50% of its steady value  $Q_c^s$ ),  $4.64$  (-25%),  $-4.64$  (25%) and  $-9.28$  (50%).

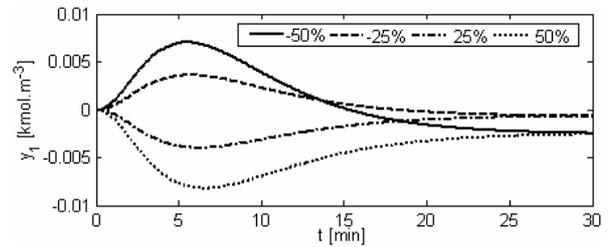


Figure 9: Course of output concentration  $c_B$  ( $y_1$ ) for various step changes of heat removal  $\Delta Q_c$

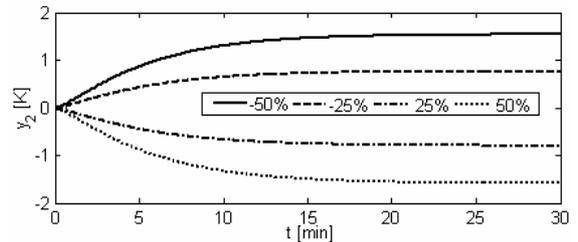


Figure 10: Course of output temperature  $T_r$  ( $y_2$ ) for various step changes of heat removal  $\Delta Q_c$

Again, the output product concentration  $c_B$  (output  $y_1$ ) has non-minimum phase behaviour and changes the sign of gain.

On the other hand, the output reactive temperature  $T_r$  (output  $y_2$ ) in Figure 10 could be expressed with a second-order linear model. This output was used as the controlled variable in the control section and a change of the heat removal was considered as a manipulated variable.

We did not have a real model of this plant at our disposal, there was no chance to check the validity of the model and we supposed that the model is satisfactory.

#### 4. Control strategy and controller design

The adaptive control with usage of polynomial synthesis and pole-placement method was used for control of this reactor. This method could be used for systems with negative control properties such as non-minimum phase behaviour or transport delay.

##### *External continuous-time linear model*

We used a continuous-time (CT) external linear model (ELM) as a description of the relation between the output and the input signals. The CT model is in time domain generally described by a differential equation

$$a(\sigma)y(t) = b(\sigma)u(t) \quad (6)$$

where  $\sigma = d/dt$  is the differentiation operator and if initial conditions are equal to zero it is possible to transform equation (6) to the complex domain using the Laplace transform (L-transform):

$$a(s)Y(s) = b(s)U(s) + o_1(s) \quad (7)$$

with  $s$  as a complex variable and  $o_1(s)$  denoting initial conditions.

Polynomials  $a(s)$  and  $b(s)$  have the following form

$$a(s) = \sum_{i=0}^{\deg a} a_i s^i, \quad b(s) = \sum_{j=0}^{\deg b} b_j s^j$$

If the initial values are equal to zero, we can introduce a transfer function

$$G(s) = \frac{Y(s)}{U(s)} = \frac{b(s)}{a(s)} \quad (8)$$

This transfer function must be proper, i.e.

$$\deg b \leq \deg a \quad (9)$$

##### *Parameter estimation*

It is clear that the equation (6) could not be used directly for parameter estimation because of derivatives of the input and output variables. It is necessary to introduce filtered variables  $u_f$  and  $y_f$  which are computed from differential equations

$$\begin{aligned} c(\sigma)u_f(t) &= u(t) \\ c(\sigma)y_f(t) &= y(t) \end{aligned} \quad (10)$$

where  $c(\sigma)$  is a stable polynomial which satisfies condition  $\deg c \leq \deg a$ .

The L-transform of (10) to complex domain yields

$$\begin{aligned} c(s)U_f(s) &= U(s) + o_2(s) \\ c(s)Y_f(s) &= Y(s) + o_3(s) \end{aligned} \quad (11)$$

where  $o_1, o_2$  are polynomials which take into account initial conditions of filtered variables. If we substitute the equation (11) into the equation (7), the result is

$$Y_f(s) = \frac{b(s)}{a(s)}U_f(s) + \Psi(s)$$

Where  $\Psi(s)$  is a rational function in  $s$  which takes into account initial conditions of filtered and non-filtered variables.

Dynamics of filters (10) must be faster than dynamics of the controlled process. The selection of filter parameters is problematic and it is closely connected with the knowledge about the system behaviour. If we do not have any knowledge about the system it is good to set these parameters a priori small.

Values of filtered variables were acquired in discrete time moments  $t_k = k \cdot T_v$  for  $k = 0, 1, 2, \dots$ , where  $T_v$  denotes a sampling period and the regression vector for  $n = \deg a$  and  $m = \deg b$  has the following form

$$\begin{aligned} \boldsymbol{\varphi}^T(t_k) &= [-y_f(t_k), -y_f^{(1)}(t_k), \dots, -y_f^{(n-1)}(t_k), \\ &u_f(t_k), u_f^{(1)}(t_k), \dots, u_f^{(m)}(t_k), 1] \end{aligned}$$

The vector of parameters is

$$\boldsymbol{\theta}^T(t_k) = [a_0, a_1, \dots, a_{n-1}, b_0, b_1, \dots, b_m]$$

Then, parameters of polynomials  $a$  and  $b$  can be estimated in discrete time intervals from

$$y_f^{(n)}(t_k) = \boldsymbol{\theta}^T(t_k) \cdot \boldsymbol{\varphi}(t_k) + \Psi(t_k)$$

Adaptivity of the process is fulfilled by recursive parameter estimation during the control. Recursive Least Squares (RLS) method with direct forgetting (Kulhavy and Karny 1984) was used for this parameter estimation. The discrete version of the algorithm could be described by following equations:

$$\begin{aligned} \varepsilon(k) &= y(k) - \boldsymbol{\varphi}^T(k) \cdot \boldsymbol{\theta}(k-1) \\ r(k-1) &= \boldsymbol{\varphi}^T(k) \cdot \mathbf{P}(k-1) \cdot \boldsymbol{\varphi}(k) \\ \mathbf{L}(k) &= \frac{\mathbf{P}(k-1) \cdot \boldsymbol{\varphi}(k)}{1 + r(k-1)} \\ \gamma(k) &= \frac{1}{\lambda_1(k-1) / \lambda_2 + \boldsymbol{\varphi}^T(k) \cdot \mathbf{P}(k-1) \cdot \boldsymbol{\varphi}(k)} \\ \beta(k-1) &= \begin{cases} \lambda_1(k-1) - \frac{1 - \lambda_1(k-1)}{r(k-1)} & \text{for } r(k-1) > 0 \\ 1 & \text{for } r(k-1) = 0 \end{cases} \end{aligned}$$

$$\mathbf{P}(k) = \mathbf{P}(k-1) - \frac{\mathbf{P}(k-1) \cdot \boldsymbol{\varphi}(k) \cdot \boldsymbol{\varphi}^T(k) \cdot \mathbf{P}(k-1)}{\beta(k-1)^{-1} + r(k-1)}$$

where  $\lambda_i$  is the forgetting factor computed e.g. via

$$\lambda_i(k) = 1 - K \cdot \gamma(k) \cdot \varepsilon^2(k)$$

for a small value of the constant  $K$ .

#### Control system configuration

Two control system configurations were adopted. The first 1DOF (one degree-of-freedom) configuration displayed in Figure 11 has a regulator only in the feedback part. On the other hand, configuration with two degrees-of-freedom (2DOF) has both feedback and feedforward parts – see Figure 12.

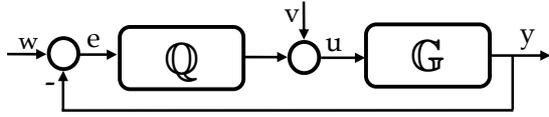


Figure 11: 1DOF control scheme

In both schemes  $G$  is an approximate transfer function from (8),  $Q$  is feedback and  $R$  feedforward part of the controller. The signal  $w$  is the reference signal,  $u$  is a control variable,  $e$  is an error,  $y$  represents an output variable and  $v$  is disturbance in the input to the system.

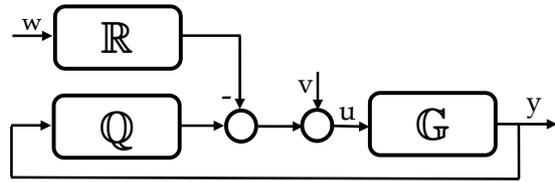


Figure 12: 2DOF control scheme

Both, control variable  $u$  and disturbance  $v$  are considered from the ring of step functions because of simplicity. For step chanted the transfer functions of the feedback and feedforward parts include integration component  $1/s$ :

$$Q(s) = \frac{q(s)}{s \cdot p(s)}, R(s) = \frac{r(s)}{s \cdot p(s)} \quad (12)$$

where  $q$ ,  $p$  and  $r$  are polynomials in  $s$ . Algebraic methods were used for computation of coefficients of the polynomials in (12). The first diophantic equation

$$a(s) \cdot s \cdot p(s) + b(s) \cdot q(s) = d(s) \quad (13)$$

is used for computation of coefficients of the polynomials  $p(s)$  and  $q(s)$  and coefficients of the polynomial  $r(s)$  in the feedforward part are obtained from the second diophantic equation

$$t(s) \cdot s + b(s) \cdot r(s) = d(s) \quad (14)$$

where  $t(s)$  is an additive stable polynomial with arbitrary coefficients, because these coefficients are not used for computation of the coefficients of the polynomial  $r(s)$ . In both equations,  $d(s)$  on the right side is a stable polynomial. The feedback regulator  $R(s)$  ensures stability, load disturbance attenuation for both configurations and asymptotic tracking for 1DOF configuration. On the other hand, feedforward part  $Q(s)$  ensures asymptotic tracking in 2DOF configuration.

A demand for a stable controller is fulfilled if the polynomial  $p(s)$  in the denominators of (12) is stable. Inner properness holds if all transfer functions are proper. Transfer functions of the feedforward and feedback parts (12) are proper if

$$\deg q \leq \deg p + 1, \deg r \leq \deg p + 1 \quad (15)$$

Degrees of the polynomials  $p$ ,  $q$  and  $r$  are computed with respect to conditions (9), (15) and solubility of diophantic equations (13) and (14) as follows

$$\deg q = \deg a, \deg p \geq \deg a - 1, \deg r = 0 \quad (16)$$

Roots of the polynomial  $d(s)$  on the right side of equations (13) and (14) are poles of the closed-loop and the control quality is determined by the placement of these poles. A method, where poles were connected to parameters of the controlled system was used to set poles of the characteristic polynomial. Then, the polynomial  $d(s)$  could be rewritten for aperiodical processes to the form

$$d(s) = n(s) \cdot (s + \alpha)^{\deg d - \deg n}$$

for  $\alpha > 0$  be an optional coefficient reflecting closed-loop poles and stable polynomial  $n(s)$  is obtained from the spectral factorization of the polynomial  $a(s)$

$$n^*(s) \cdot n(s) = a^*(s) \cdot a(s)$$

#### Simulation experiment

Changes of the heat removal were used as manipulated variable and the controlled output was the temperature of the reactant:

$$y(t) = T_r(t) - T_r^s(t) [K]; u(t) = 100 \cdot \frac{Q_c(t) - Q_c^s(t)}{Q_c^s(t)} [\%]$$

Dynamic analysis of the output temperature shows that ELM could be expressed by a second order system with the relative order one described in the continuous-time form by the differential equation

$$y^{(2)}(t) + a_1 y^{(1)}(t) + a_0 y(t) = b_1 u^{(1)}(t) + b_0 u(t)$$

And in the complex  $s$ -plane as the transfer function

$$G(s) = \frac{b_1 s + b_0}{s^2 + a_1 s + a_0} \quad (17)$$

Filtered variables  $u_f$  and  $y_f$  were computed from differential equations similar as in (10)

$$y_f^{(2)}(t) + c_1 y_f^{(1)}(t) + c_0 y_f(t) = y(t)$$

$$u_f^{(2)}(t) + c_1 u_f^{(1)}(t) + c_0 u_f(t) = u(t)$$

where parameters  $c_1$  and  $c_0$  were chosen equal to  $c_1 = 1.4$  and  $c_0 = 0.49$ . The recursive least squares method with directional forgetting was used for parameter estimation. The regression vector and vector of parameters for ELM (17) have the form

$$\boldsymbol{\varphi}^T(t_k) = [-y_f(t_k), -y_f^{(1)}(t_k), u_f(t_k), u_f^{(1)}(t_k)]$$

$$\boldsymbol{\theta}^T(t_k) = [a_0, a_1, b_0, b_1]$$

Starting values for the RLS method with direct forgetting were following: the vector of parameters  $\boldsymbol{\theta}^T(k) = [0.1, 0.1, 0.1, 0.1]$ ; the covariance matrix  $\mathbf{P}(0)$  with the dimension  $4 \times 4$  has  $10^{-6}$  on the diagonal; a starting value of the forgetting factor was  $\lambda_1(0) = 0.95$  and further,  $\varepsilon(0) = 0$ ,  $\gamma(0) = 0$ . Degrees of polynomials of the feedforward  $R(s)$  and feedback  $Q(s)$  parts of the controller were computed from the transfer function (17) via (16):  $\deg q = 2$ ,  $\deg p = 1$ ,  $\deg r = 0$  and then, transfer functions were

$$\tilde{Q}(s) = \frac{q_2 s^2 + q_1 s + q_0}{s \cdot (p_1 s + p_0)}, \tilde{R}(s) = \frac{r_0}{s \cdot (p_1 s + p_0)}$$

Coefficients of polynomials  $p$ ,  $q$  and  $r$  were computed from diophantic equations (13) and (14) where polynomial  $d(s)$  was of the fourth degree

$$d(s) = n(s) \cdot (s + \alpha)^2$$

With parameter  $\alpha > 0$ . Coefficients of the polynomial  $n(s)$  are computed via

$$n_0 = \sqrt{a_0^2}, n_1 = \sqrt{a_1^2 + 2n_0 - 2a_0}$$

The simulation took 300 min and three changes of the reference value were made during this time. The reference value was  $w(t) = 2$  K in the time domain  $0 < t < 100$  min,  $w(t) = -1$  K on the interval  $100 < t < 200$  min and  $w(t) = 1$  K in the last interval  $200 < t < 300$  min. A proportional regulator, with gain  $k_p = -1$ , was used for first 15 steps because of identification, which does not work well with oscillating values in the beginning.

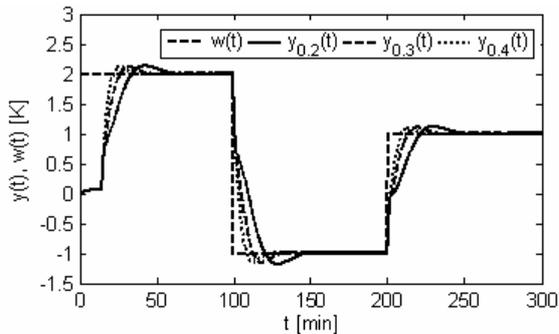


Figure 13: Course of output  $y(t)$  for 1DOF set-up and different values of the parameter  $\alpha = 0.2; 0.3$  and  $0.4$

The sampling period for the identification and the control action value was  $T_v = 0.5$  min. The manipulated variable was limited to the interval  $-75\% \leq u(t) \leq 75\%$ .

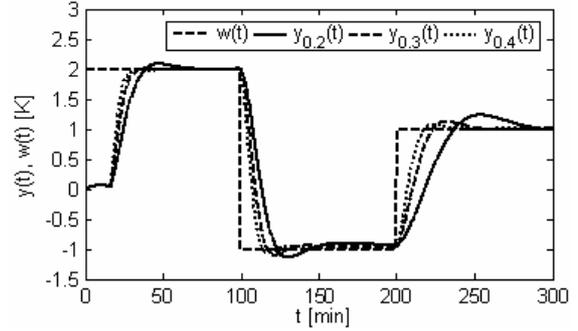


Figure 14: Course of output  $y(t)$  for 2DOF set-up and different values of the parameter  $\alpha = 0.2; 0.3$  and  $0.4$

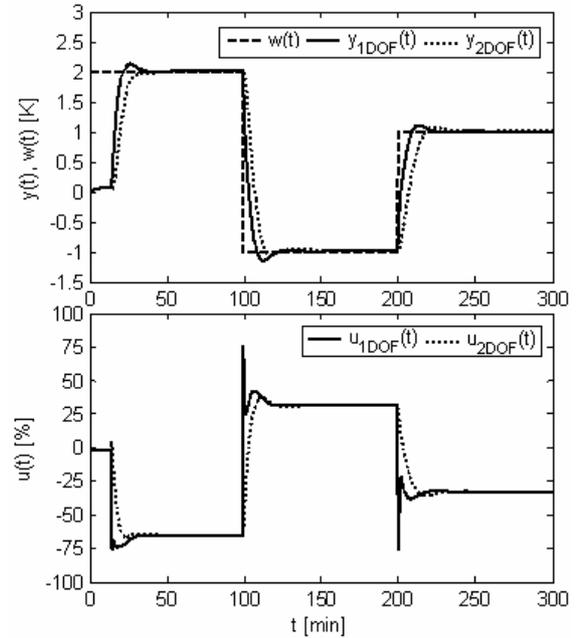


Figure 15: Comparison of outputs  $y(t)$  and action values  $u(t)$  for 1DOF and 2DOF set-ups with root  $\alpha = 0.4$

Figure 13 and Figure 14 show results for 1DOF and 2DOF configuration and different parameters  $\alpha$ . As it can be seen in both figures, this parameter affects mainly the overshoot and speed of the control. With increasing value of parameter  $\alpha$  the output response is quicker and the overshoot smaller in this case.

Figure 15 compares results for 1DOF and 2DOF configuration for the same placement of the parameter  $\alpha = 0.4$ . The main advantages of 2DOF configuration are lower overshoots and smoother changes of action value. Parameter estimation during the simulation for 1DOF configuration displayed in Figure 16 shows that the chosen RLS method with direct forgetting has a problem only in the beginning of the simulation because of small amount of information.

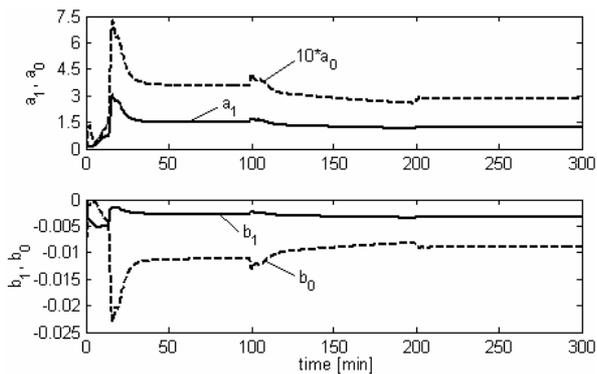


Figure 16: Course of estimated parameters  $a_0$ ,  $a_1$ ,  $b_0$  and  $b_1$  for 1DOF configuration and root  $\alpha = 0.4$

## CONCLUSION

The paper shows methodology from simulation of a nonlinear process represented by the CSTR reactor to adaptive control of this system. Simulation of the steady-state reveals the ideal working point and dynamic analysis shows step responses for more input values. Proposed adaptive control based on polynomial synthesis and the pole-placement method provides good control results although the system has negative control properties such as non-minimum phase behaviour and changes the sign of gain. The control process has quicker slope and overshoots lower with increasing value of the parameter  $\alpha$ , but increasing cannot be infinite because of limitation of the action variable. The main advantage of the 2DOF control configuration is represented by smaller overshoots and smoother changes of the action variables which is important from the practical point of view, where action value could be e.g. valve twist. Shock changes of the twist could destroy or damage the valve. The results demonstrate the usability of this control method, which is represented by a good quality and stability of the output response.

## REFERENCES

- Åström, K.J. and B. Wittenmark (1989). Adaptive Control. Addison Wesley, Reading, MA.
- Dostál, P., Bobál, V., Sysel, M. (2002): Design of controllers for integrating and unstable time delay systems using polynomial method, In: Proc. 2002 American Control Conference, Anchorage, Alaska, USA, 2773-2778.

- Grimble, M. J. (1994). Robust industrial control. Optimal design approach for polynomial systems. Prentice Hall, London.
- Chen, H., Kremling, A., Allgöwer, F. (1995): Nonlinear Predictive Control of a Benchmark CSTR. In: Proceedings of 3rd European Control Conference. Rome, Italy
- Ingham, J., Dunn, I. J., Heinzle, E., Přenosil, J. E. (2000): Chemical Engineering Dynamics. An Introduction to Modeling and Computer Simulation. Second, Completely Revised Edition, VCH Verlagsgesellschaft, Weinheim.
- Kučera, V. (1993). Diophantine equations in control – A survey. Automatica, 29, 1361-1375.
- Kulhavý, R., Kárný, M. (1984): Tracking of slowly varying parameters by directional forgetting, In: Proc. 9th IFAC World Congress, vol. X, Budapest, 78-83.
- Luyben, W. L. (1989): Process Modelling, Simulation and Control for Chemical Engineers. McGraw-Hill, New York.
- Wahlberg, B. (1990): The effects of rapid sampling in system identification, Automatica, vol. 26, 167-170.

## ACKNOWLEDGMENT

This work was supported by the Grant Agency of the Czech Republic under grants No. 102/03/H116 and by the Ministry of Education of the Czech Republic under grant MSM 7088352101.

## AUTHOR BIOGRAPHIES



**JIRI VOJTESEK** was born in Zlin, Czech Republic and went to Tomas Bata University in Zlin, where he got his master degree in chemical and process engineering in 2002. He went to 3 month study stays at University of Applied Science Cologne, Germany in 2003 and Politecnico di Milano, Italy in 2004. Now he is about to finish his Ph.D. which is focused on Modern methods for control of chemical reactors. You can contact him on email address vojtesek@ft.utb.cz.



**PETR DOSTAL** went to Technical University of Pardubice, where he obtained master degree in 1968 and PhD. degree in Technical Cybernetics in 1979. In the year 2000 he became professor in Process Control. His research interest are modeling and simulation of continuous-time chemical processes, polynomial methods, optimal, adaptive and robust control. You can contact him on email address dostalp@ft.utb.cz.

# COLLISION MODELLING FOR HIGH ENERGY BALL MILLS USING EVENT DRIVEN SIMULATION

Roland Reichardt, Stephan Adam and Wolfgang Wiechert  
Department of Simulation, Faculty 11/12, University of Siegen,  
Paul-Bonatz-Str. 9-11, 57068 Siegen, Germany  
{Reichardt, Adam, Wiechert}@simtec.mb.uni-siegen.de

## ABSTRACT

Mills in general are used for grinding processes. This field of mechanical process engineering is dominated by empirical equations, caused by the difficulty of measuring the motion of the grinding media. In this paper, a high energy ball mill is presented as the subject of event driven simulation algorithms. The main steps of the algorithm are described. In detail, the collision response model of the ball's collision is derived and the differences with respect to classical approaches are explained. In order to make the simulation as realistic as possible, the simulation uses experimental values, measured with real material pairings. High speed videos of a transparent grinding chamber have been analyzed and compared to the simulation's visualisation in order to figure out the relevant parameters of the collision response model, which are essential. Simulation results and potential pitfalls of these kinds of simulations are discussed. It can be shown that the assumption that rotation of the balls and tangential forces are absent is an oversimplification.

## INTRODUCTION

In materials engineering, high energy ball mills are used to develop new powder materials. The grinding medium, about 4000 steel balls in a lab scale mill, is accelerated by rotor blades in a horizontal drum (Fig. 1). Up to the present, only the overall energy balance has been measured experimentally (Zoz and Reichardt 1999). More substantial information such as, for example, the impact velocity distribution, still cannot be measured. Moreover, to understand the whole process, for example for the scale up from lab scale to industrial scale, it would be important to achieve a good understanding of the dynamics of the mill. As a matter of fact, up to now, the configuration of the process parameters has been carried out empirically.

From a simulationist's viewpoint, ball mills are, mechanically, many body systems that are usually simulated by time continuous algorithms such as the discrete element approach.

Event driven algorithms - even though they are known to be far more efficient - have not become as popular. Event driven molecular dynamics simulations have been successfully applied to typical physical topics such as granular gases (Gavrilova et al. 2002, Reichardt and Wiechert 2003).

The basic concept of a discrete event simulation is presented in the following section. After this, the collision

model is explained in detail. Furthermore different simulation experiments applying the different collision models are presented.

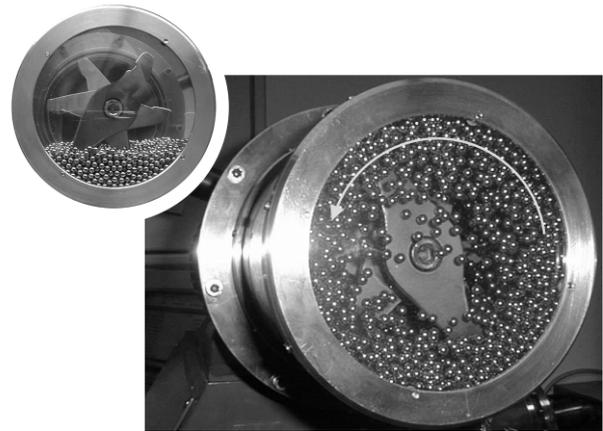


Figure 1: Operational mill; experimental transparent grinding chamber.

## DISCRETE EVENTS

In this work, the high energy ball mill is described by an event driven particle simulation using collisions as basic events. The balls behave like dilute granular systems, where the mean free path of inelastically colliding particles is much larger than the average particle size. Furthermore, it is assumed that the balls do not build clusters of balls, in other words, that there are no multiple contact points, but rather only binary collisions (Briliantov and Pöschel 2004). Pairwise collisions have been observed at typical rotor velocities for the grinding process. Of course, at low rotor speeds, the assumption of pairwise collisions becomes false, but the start phase of the milling process is very short in comparison to the total milling time. During the flight, it is assumed there is no energy dissipation due to air resistance. In fact the milling chamber is evacuated.

## Event List

Event driven molecular dynamics processes in general are modelled as a series of discrete instantaneous collisions. These events are stored in an event list ordered by time. Event lists are first mentioned in the late fifties of the last century (Alder and Winwright, 1959) and are still topics of research activities (Dahl et al. 2001, Miller and Luding 2004)

Generally, the event processing is an iteration over the steps 1-4 below:

1. The simulation clock jumps from one collision event to the next. The simulation time is updated by handling the next event of the event list, e.g. a ball to ball collision
2. The algorithm searches for the next possible collision time for each of the two balls involved. This is computed analytically. In a system of hard spheres, events refer to collisions, involving:
  - exactly two balls
  - one ball and the cylindrical walls of the drum
  - one ball and the flat side walls of the drum
  - one ball and one of the rotor blades

The collision time is the flight time from the actual event to the new event.

3. The locations of the two new collisions are calculated using the flight time just computed to the collision point. The calculation of the collision response leads to the new flight direction at this point caused by the collision event.
4. These new events are inserted into the event list. Hereby, already predicted collisions may become invalid as a result of another ball crossing its flight path. In this case, the event list handling routine must ensure consistence by deleting the invalid event, and all events following.

The interested reader is referred to a previous publication with a more detailed description of this procedure (Reichardt and Wiechert 2003).

### Collision Detection

In time continuous simulations, collisions are detected by object penetration checks after each time step. Contrary to this, in discrete event simulations, the collision detection (collision time) is computed analytically. The collision time is found when the distance function of two objects is zero for the first time. Mathematically speaking, the value sought is the minimum positive root of the distance over time. Taking into account the knowledge of physically impossible situations, the collision time of two balls can be calculated directly without case differentiation (Reichardt and Wiechert 2003). With regard to the ball to rotor collision, finding the first positive real root is more difficult (caused by the transcendental equation for the distance function).

Because the algorithm will observe only pairs of objects, ignoring the presence of the others, several collision times can be found. The minimum of these collision times is the only valid collision time, because this collision will occur first.

The efficient handling of the event list is a substantial task for discrete event simulation. New approaches for data structures and improvements of the algorithm have been presented in earlier publications (Marin et al. 1993, Reichardt and Wiechert 2003). The current implementation is able to calculate about 17000 collisions per sec-

ond for 5000 balls, in other words; the calculation time is about 6 to 7 minutes for 1 second of real time.

### COLLISION RESPONSE

The collision response model is fundamental to the accurate representation of a hard sphere simulation. Because the collisions are modelled to be instantaneous, the change in the flight direction and the energy dissipation during the collision are modelled to have zero duration. Mathematical equations which describe the collision response are the topic of recent theoretical investigations (Brilliantov and Pöschel 2004), but, to date, most of the equations implemented are oversimplified, as shown later.

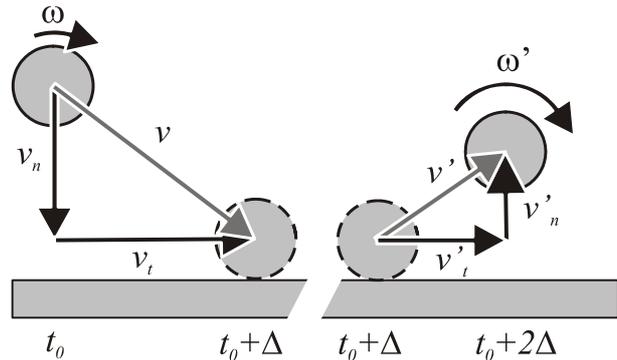


Figure 2: Velocity components of a ball before the collision (left) and after the collision (right).

The velocity of a ball is given by the velocity vector,  $v$ , which can be divided into its component in the normal direction,  $v_n$ , and in the tangential direction,  $v_t$ . Furthermore, the ball's initial rotation is given by the rotational vector,  $\omega$ . The direction of  $\omega$  can be understood as the rotational axis of the ball; the magnitude of  $\omega$  is the rotational speed. The desired values are the velocity vector,  $v'$ , and the rotational vector,  $\omega'$ , just after the collision. Figure 2 illustrates the situation described.

### Normal Coefficient of Restitution

Energy will be dissipated during collision, which results in a lower kinetic energy after the collision. The ratio of the (relative) velocity in the normal direction before and after the collision is the normal restitution coefficient,  $\varepsilon_n$ . A fully elastic collision, without any energy dissipation, is represented by  $\varepsilon_n = 1$ , whereas a fully plastic collision is represented by  $\varepsilon_n = 0$ . Physically, the loss of kinetic energy in the normal direction will cause the ball to bounce back at a lower angle.

$$\vec{v}'_n = -\varepsilon_n \vec{v}_n \quad (0 \leq \varepsilon_n \leq 1) \quad (1)$$

Even if, in equation (1),  $\varepsilon_n$  appears to be a constant, in reality it is velocity dependent. By dimensional analysis it was first proven that a constant coefficient of restitution is not consistent with physical reality (Tanaka et al. 1991). Later, this fact was proven analytically (Brilliantov et al. 1996). Unfortunately, the theoretical equations

require additional experimental investigations to assign values to the material dependent parameters.

Usually, energy dissipation caused by collisions is modelled to be velocity independent, whereas in reality it increases at higher velocities. In order to make the simulation as realistic as possible, an experimental setup has been designed and constructed to measure the velocity dependent energy dissipation during the collision of objects. The collision zone has been observed by a high speed video camera. The resulting sensor data were analyzed by image processing, which recognizes the balls within each frame of the movie. Using these coordinates, the velocity just before and just after the collision can be calculated – their quotient is the desired value (restitution coefficient). In a prior project, an experimental setup has been designed to measure the restitution coefficient in the normal direction of ball bearing spheres and a stainless steel plane. The material pairings are exactly those of the real mill. The velocity dependent restitution coefficient shown in Figure 3 has been measured.

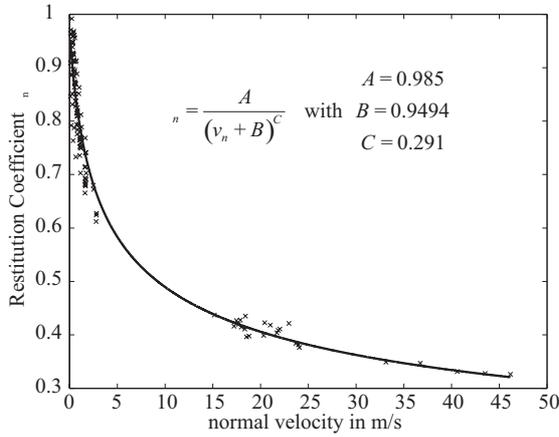


Figure 3: The velocity dependent restitution coefficient in the normal direction.

The simulation results presented in this paper are calculated using both constant and velocity dependent restitution coefficients.

The collision response of two balls with the same mass can be calculated using equation (2). Here,  $\vec{e}$  is the unit vector through the balls' centres during the collision.

$$\vec{v}' = \vec{v} \pm \frac{1}{2}(\varepsilon_n + 1)(\vec{v}_{rel} \cdot \vec{e}) \vec{e} \quad (2)$$

### Tangential Coefficient of Restitution

By introducing rotation, it is not the velocity components of the centres of mass that are important, but rather the velocity of the surfaces of the balls at the point of contact in the normal and in the tangential direction. This is, for identical balls with the same mass and the radius  $R$ ,

$$\begin{aligned} \vec{g}_n &= (\vec{v}_{rel} \cdot \vec{e}) \vec{e} \\ \vec{g}_t &= \vec{v}_{rel} - \vec{g}_n + R(\vec{e} \times (\vec{\omega}_1 + \vec{\omega}_2)) \end{aligned} \quad (3)$$

Note that  $\vec{g}_n$  is simply the relative velocity in the normal direction, whereas  $\vec{g}_t$  is the relative velocity in the tangential direction and, additionally, the velocity of the surfaces caused by the rotation of the balls.

To describe the change of the tangential component of the (relative) velocity, the coefficient of tangential restitution has to be introduced. Rotation of the balls contributes to the tangential component of their relative motion at contact. Hence, the balls' collision response is completely described by two coefficients of restitution:

$$\begin{aligned} \vec{g}_n' &= -\varepsilon_n \vec{g}_n & (0 \leq \varepsilon_n \leq 1) \\ \vec{g}_t' &= \varepsilon_t \vec{g}_t & (-1 \leq \varepsilon_t \leq 1) \end{aligned} \quad (4)$$

This yields two velocities after the collision:

$$\vec{v}' = \vec{v} \pm \frac{1}{2}((\varepsilon_n + 1)\vec{g}_n + (1 - \varepsilon_t)\vec{g}_t) \quad (5)$$

For the frictionless case,  $\varepsilon_t = 1$ , equation (5) is reduced to equation (2). In reality, during a collision, the normal force is not a constant. It increases from zero at the beginning of the collision to the moment of maximum compression, and decreases during the rest of the collision. Therefore, the tangential type of contact, sliding and rolling, may change during the impact. The complicated dynamics of such collisions is known (Stronge 1990).

Simplified, the collision time is assumed to be zero, and, therefore, the tangential restitution coefficient accounts for the energy dissipation of the collision. Note that, due to the discrete event approach, collision durations are not permitted. Further, it is assumed that the loss of kinetic energy in the tangential direction is equal to the increase of rotation per ball:

$$\vec{\omega}' = \vec{\omega} + \frac{(1 - \varepsilon_t)(\vec{e} \times \vec{g}_t)}{2R} \quad (6)$$

Hence, energy is dissipated in the normal direction only, not in the tangential direction.

It should be noted that the tangential velocity component is reduced by the tangential restitution coefficient. Therefore, the ball is pushed away from a wall because the response angle is increased at the moment of bounce.

### SIMULATION EXPERIMENTS

Distributed simulations have been performed using 256 processor computer clusters. Without applying the discrete event algorithm, these experiments could not be concluded within a reasonable time. We have conducted variation studies of velocity dependent and velocity independent restitution coefficients. Furthermore, we have carried out parametric studies to show the influence of the tangential restitution coefficient, and thus the influence of the balls' rotation.

Initially, the balls are equally distributed in space and the directions of the velocity vectors are also equally

distributed. The magnitude of the velocity vector – the absolute velocity – has been set to one meter per second for all balls. When setting up the initial conditions, the algorithm will guarantee that there are no object penetrations. Of course, this setup is not realistic. Therefore, the simulation has to run until the initial setup no longer influences the simulation results. This is measured by observing the histogram of the absolute velocities of the balls, which are represented by their median.

These histograms are generated every 0.03 seconds of simulated time, which has been shown to be a reasonable time interval. As well as the histogram of the absolute values, several other histograms are generated during simulation time, e.g. the histogram of the impact velocities in the normal direction, which are important for the understanding of the milling efficiency. In this paper, the focus is set on the median of the absolute velocity as a representative value.

### Steady State of the Simulation

The steady state of the simulation is reached when the simulation has “forgotten” its initial setup. From this point on, the median of the absolute velocity has reached an asymptotic value, which will not change significantly with increasing simulation time. The median values are smoothed by a median filter over the past three measurement points.

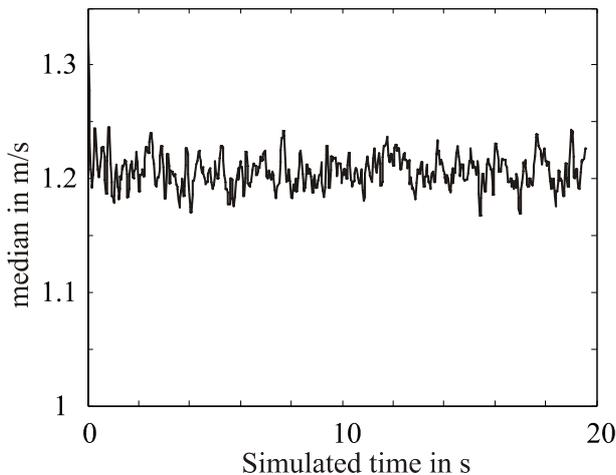


Figure 4: Representative progression of a median over simulation time.

The asymptotic behaviour of the median of a typical simulation is shown in Figure (4). Although shown indirectly by the y-axis scale, the range is between 1.0 – the initial setup of the simulation – and above 1.35, which is caused by the initial impulse of the rotor blades. A stationary behaviour is observed beyond 0.5 seconds of simulation time. Nevertheless, every simulation has been calculated for 30 seconds, even if this is not shown in every figure.

The simulations have been visualised and animated using the ray tracer software program POWRAY 3.6 (Fig. 6). The scripts for these animations are generated by the simulation software.

## RESULTS

### Constant Restitution Coefficient

As assumed in most simulations of this type, the rotational influence is ignored and the normal restitution has been assumed to be constant, even though this has already been explained previously to be false. Various simulations with this approach have been performed for a range of  $\varepsilon_n = 0.75 \dots 0.95$ , which is a popular range in the literature for arbitrary kinds of material pairings.

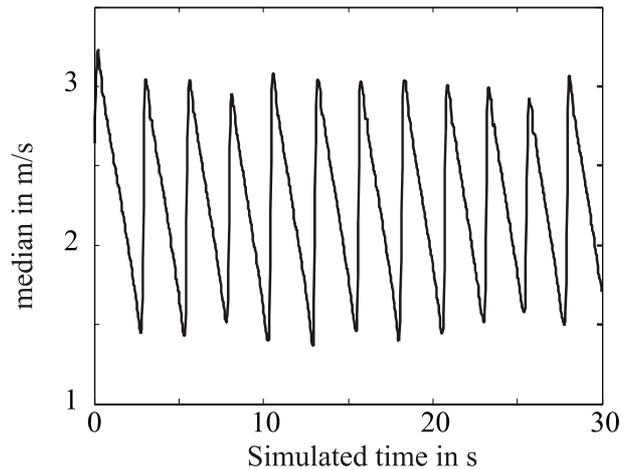


Figure 5: Simulation with constant restitution coefficient. The saw tooth curve is not realistic.

Surprisingly, a synchronising effect was observed. Just after the start of the simulation, the balls are pushed by the rotor blades towards the inner surface of the grinding chamber and then “roll” – this is not really possible in discrete event simulations – along the surface, where their velocity is decreasing to a certain threshold (Zoz et al. 2002). This critical velocity is reached when the gradient of the parabolic flight path is lower than the gradient of the cylindrical grinding chamber. After this, they fall in a parabolic flight path into the grinding chamber.

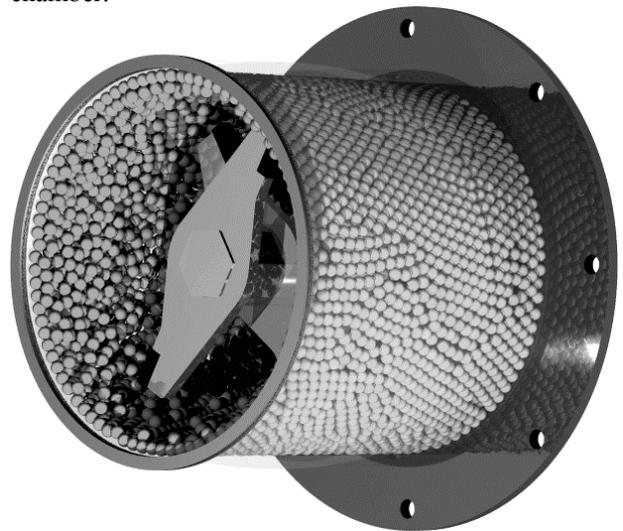


Figure 6: Balls rolling along the grinding chamber.

At every collision with the wall, the ball's trajectory moves closer to the wall, as explained earlier. Hence, all balls move along the inner surface of the grinding chamber. In the outer regions of the grinding chamber, the balls are no longer within the reach of the rotor blades, as there is a distance of three ball diameters between the rotor tips and the inner surface.

Due to their tangential trajectories, only ball to ball collisions reduce their speed. As a consequence of the high ball density in the outer regions, the balls synchronise their speed – see Figure 6. This behaviour can be observed at all relevant rotational speeds.

Summarizing, it can be said that, as well the high ball density at the outer regions of the grinding chamber, the periodic accelerations are not realistic. Hence, the collision response model used in these simulations is oversimplified.

### Velocity dependent Restitution

The first improvement to the classical damping model is the implementation of the velocity dependent restitution coefficient based on the measured values of Figure 3. This leads to high energy loss at high relative velocities, e.g. at the rotor collisions, and to low energy loss at low relative velocities in the normal direction, e.g. the bouncing of a ball along the inner surface of the grinding chamber. Varying the rotor speed results in a narrower spectrum of velocity distributions. Even if the motion is more realistic, the balls are still located at the outer regions of the grinding chamber, which is not realistic.

### Including tangential Restitution

As demonstrated by the previous experiments, the classical collision response model must be changed to achieve more realistic results. This is achieved by considering the rotation of the balls and the tangential restitution coefficient.

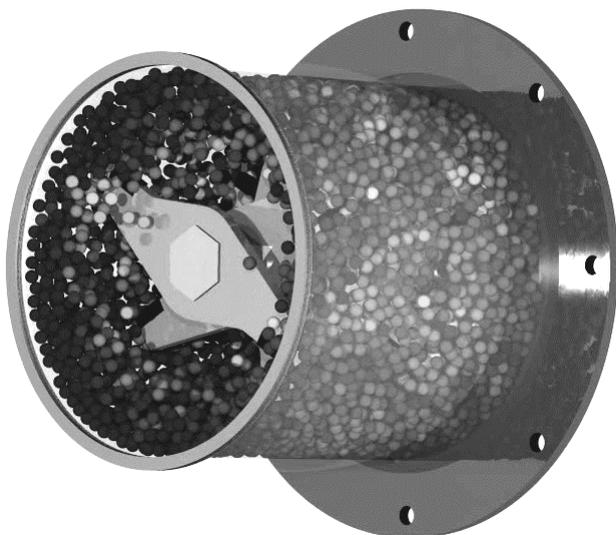


Figure 7: Realistic ball motion

The collision response model of the simulation has been expanded to take into account the dynamic normal restitution coefficient and a constant tangential restitution coefficient.

The analysis of the median values – see Figure 4 – has shown an asymptotic behavior, as in the real mill. Observations of the simulation visualizations have confirmed this realistic ball motion, because the balls' trajectories are very similar to those observed by the high speed video camera – see Figure 1. Consequently, the simulation using both restitution coefficients is able to reproduce the grinding media's motion.

In addition to the higher degree of realistic representation of these simulations, the CPU time has been reduced. This is caused by the more infrequent, but more realistic, collisions. The “rolling” effect of the prior set of simulation described in the previous section implies collisions with a very short distance, and, therefore, is time consuming.

In the current configuration, the simulation calculates approximately 6 minutes for 1 second of simulated time, including the generation of the various histograms

## CONCLUSION

In this paper, we present a new application for discrete event simulation, namely, a high energy ball mill. A brief description of the algorithm using collision events as discretization markers, and the explanation of the analytical collision detection algorithm has been given. In detail, the collision response model has been explained. It has been shown that the energy lost during a collision can be described by two parameters, the tangential and normal restitution coefficients.

In two simulation experiments, the influence of the collision response model on the validity of the simulation has been shown. The classical simplified approach to calculate the collision response using solely a constant restitution coefficient in the normal direction has been shown to be an oversimplification. In a second experiment, we have implemented a velocity dependent restitution coefficient in the normal direction, and a constant restitution coefficient in the tangential direction. This approach leads to realistic ball motion inside the mill. The advantage of this simulation technique, namely the high efficiency, has been demonstrated.

## OUTLOOK

In future work, the collision response model has to be expanded by implementing a velocity dependent tangential restitution coefficient. Unfortunately, significant experimental efforts have to be expended to measure these values. Nevertheless, this simulation is able to reproduce the balls' motion. Hence, it can be used to assist design engineers to optimize rotor blades or the shape of the grinding chamber. Furthermore, it can be used to calculate the motor's energy consumption based on the milling chamber's geometry.

## REFERENCES

- Alder, J. and Winwright, T. ~ E. 1959. "Studies in Molecular Dynamics. I. General Method", J. Chem. Phys. 31, 459
- Brilliantov, N.V. and Pöschel, Th. 2004. "Kinetic Theory of Granular Gases", Oxford University Press
- Brilliantov, N.V.; Spahn, F.; Hertzsch, J.-M. and Poeschel, Th. 1996. „Model for collisions in granular gases”, Phys. Rev. E, 53, 5382-5392.
- Dahl, J.; Chetlur, M. and Wilsey, P.A. 2001. "Event List Management In Distributed Simulation", European Parallel Computing Conference (Euro-Par '01).
- Gavrilova, M.; Rokne, J.; Gavrilov, D. and Vinogradov O. 2002. "Optimization Techniques in an Event-Driven Simulation of a Shaker Ball Mill", International Workshop on Computational Geometry and Applications'02, Lecture Notes in Computer Science LNCS 2331, 115-124
- Marin, M. ; Risso, D. and Cordero, P. 1993 "Efficient Algorithms for Many-Body Hard Particle Molecular Dynamics", Journal of Computational Physics, 109, 306
- Miller, S. and Luding, S. 2004. "Event-driven molecular dynamics in parallel", Journal of Computational Physics, Vol.193 , Issue 1, 306 - 316
- Ramírez, R.; Pöschel, T.; Brilliantov, N. V. and Schwager T. 1999. "Coefficient of restitution of colliding viscoelastic spheres", Phys. Rev. E, 60, 4465
- Reichardt, R. and Wiechert, W. 2003. "Simulating High Energy Ball Mills using Discrete Events", Simulation News Europe, Issue 38/39, pp. 3-9
- Reichardt, R. and Wiechert, W. 2003. "Event driven simulation of a high energy ball mill", ASIM 2003, 17. Symposium Simulationstechnik, 249-254
- Stronge, W.I. 1990. "Rigid body collision with friction", Proc. Roy. Soc. A, 431, 169
- Tanaka, T.; Ishida, T. and Tsuji, Y. 1991. "Direct numerical simulation of granular plug flow in a horizontal pipe", Trans. Jap. Soc. Mech. Eng., 57, 456
- Zoz, H. and Reichardt, R. 1999. "Energy Balance During Mechanical Alloying, Measurement and Calculation Method supported by the Maltoz-Software", In *Sixth International Conference on Composite Engineering (ICCE/6)*, Orlando, USA
- Zoz, H; Reichardt, R and Kim, J.-S. 2002 "Application and Design of Drummills" PM2TEC'2002, In Proc. *World Congress on Powder Metallurgy & Particulate Materials* June 16-21

## AUTHOR BIOGRAPHIES



**ROLAND REICHARDT** studied mechanical engineering at the University of Siegen, Germany. He worked from 1997 to 2001 for a mechanical process engineering company of mills of varying type, initially as the head of software department, and, later, in the company management. In 2002, he returned to the University of Siegen, where he is now a scientific assistant at the Chair of Simulation and Computer Science. Currently, he is working on his doctoral thesis.



**WOLFGANG WIECHERT** studied mathematics and computer science at the University of Bonn and obtained his diploma degree in 1985. Subsequently, he joined the Department of Theoretical Biology at the University of Bonn, where he obtained his PhD in 1991. From 1991 to 1996, he worked in the Institute of Biotechnology at the Jülich Research Center, where he earned his postdoctoral lecturer's qualification. Since 1996, he has been professor for simulation at the Institute of Systems Engineering at the University of Siegen.

# MODELLING AND SIMULATION OF HYDRAULIC LOAD-SENSING SYSTEMS USING OBJECT-ORIENTED PROGRAMMING ENVIRONMENT

Gunnar Grossschmidt  
Mait Harf  
Tallinn University of Technology  
Institute of Cybernetics  
Akadeemia tee 21, 12618 Tallinn, Estonia  
E-mail [gunnarg@edu.ttu.ee](mailto:gunnarg@edu.ttu.ee), [mait@cs.ioc.ee](mailto:mait@cs.ioc.ee)

## KEYWORDS

Hydraulic load-sensing system, multi-pole model, visual programming, automatic program synthesis.

## INTRODUCTION

Modelling and simulation of hydraulic systems has been investigated in Tallinn University of Technology for several decades. General principles of the research have been published in (Grossschmidt 1997; Grossschmidt et al. 2003).

In this paper we consider a way of modelling and simulation of hydraulic load-sensing systems that enables to optimise such systems.

Hydraulic systems, in which working pressure (pressure in pump output) is kept proportional to load, are called hydraulic load-sensing systems. Such systems are mainly used in mechanisms containing numerous drives to run with purpose to save energy. These are quite complicated automatically regulating systems and until now optimal solutions for static, steady-state motion and dynamics for such systems have not been found. In this paper we consider a simulation system that enables to describe and search hydraulic load sensing systems and find better solutions to those problems. A high-level programming environment NUT (developed in the Institute of Cybernetics, Tallinn) is used as a tool for building up and managing the simulation system.

## NUT PROGRAMMING ENVIRONMENT

The NUT system is a programming tool, which supports declarative programming in a high-level language, automatic program synthesis and visual programming.

The NUT programming language rests on two paradigms: procedural object-oriented programming and the automatic synthesis of programs from declarative specifications. The NUT language is object-oriented. Concepts in it are specified as classes, and then used either in computations or for specifying new concepts. There is one big difference between the classes in ordinary object-oriented

language and in NUT. The latter contain more knowledge than ordinary classes, and can be directly used as specifications for problem solving. The description of NUT classes may contain specifications of their components, methods, initial values and other properties. Due to an equation solver built into the language processor, the system is able to interpret arithmetic equations as multi-way procedures for computing the unknown components of the equation. Each class can have a visual representation as well, so that much computing can be described visually.

Automatic synthesis of programs is a technique for the automatic construction of programs from the knowledge available in specifications of classes. Having a specification of a class, we are, in general, interested in solving the following problem: find an algorithm for computing the values of components  $y_1, \dots, y_n$  from the values of components  $x_1, \dots, x_m$ . The automatic synthesis of programs, as practised in NUT, is based on proof search in intuitionistic propositional logic.

The NUT graphics facilities include Graphics Editor, a set of graphics functions in the language, and the Scheme Editor. The Scheme Editor is a tool for visual programming that allows the user to define and use classes by means of graphical schemes. In order to draw schemes of problem specifications, we must have, for each class, an icon in the palette and an image, which will represent an object in a scheme. So there must be an icon and an image for every class. This can be done immediately after specifying a class. After specifying all the classes together with their icons and images one can specify and use for computations a number of different schemes using defined classes. There are numerous built-in features of the scheme editor, which support visual programming:

- connection lines between ports which represent equalities binding the ports;
- an interactive zoom-in window can be opened for showing or editing of any object of a scheme (this window is formatted automatically on the basis of the class specification);

- requests for computing elements of a schema can be given from menus.

### MODELLING OF HYDRAULIC SYSTEMS

A number of packages for hydraulic systems simulation have been implemented in the NUT system. Multi-pole models of hydraulic elements have been described as NUT classes together with their icons and images. Besides multi-pole model classes, several supporting classes as "clock" for the time, "source" for the disturbance, "process" for organizing the whole computing process, have been specified. Using visual specifications of described multi-pole models one can graphically compose models of various hydraulic systems.

When solving specific simulation problem, model has to be adjusted by evaluating different parameters of the elements and adding sources to elements of the model that describe disturbances of the necessary shape and values.

During the simulation, several elements of the model need parameters, which values cannot be computed at the moment they are required. For computing values of such parameters a special method has been

used. When starting the process, approximate values of such critical parameters have been given as initials. At each step of the simulation process we try to refine these initial approximate parameters using a special iteration procedure. We use the NUT system to synthesize a program for recomputing some parameters and try to recompute them iteratively until precise values of the parameters have been attained.

A special element "disp" can be used in the scheme for graphical displaying of dependencies we are interested in.

The simulation is organized as computing of static, steady-state motion and dynamics of the hydraulic device. In general, model for static and steady-state motion differs from the model for dynamic responses. Nevertheless fragments for static calculations can expand the model for dynamic responses.

### MODELLING A HYDRAULIC LOAD-SENSING SYSTEM

The scheme of a hydraulic load-sensing device of Bosch GmbH we investigate is shown in Fig. 1.

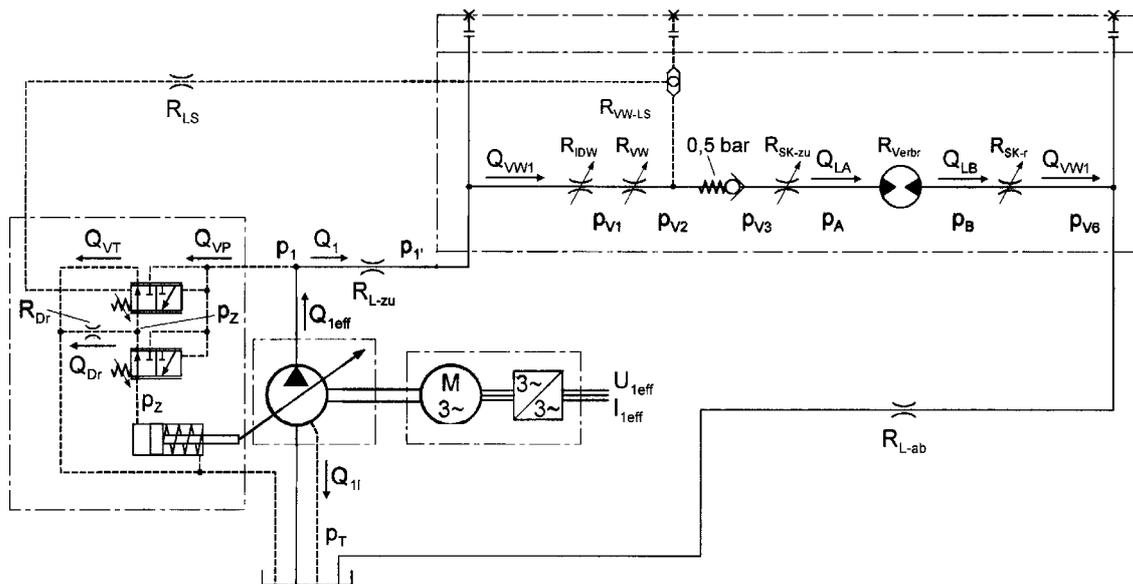


Figure 1.

The variable displacement axial piston pump is driven by an electric motor *M*. Hydraulic-mechanical control of the pump volumetric flow is performed by regulating valves and hydraulic cylinder. The feeding chain of the hydraulic motor *Rverbr* contains tube *Rl-zu*, pressure compensator *Ridw*, measuring valve *Rvw*, check valve, meter-in throttle edge *Rsk-zu*. The output chain of the hydraulic motor contains a meter-out throttle edge *Rsk-r*, and tube *Rl-ab*. The device contains load-sensing pressure feedback with resistance *Rls*.

Mathematical models of the following components of a hydraulic load-sensing system have been

developed: hydraulic-mechanical controller components, variable displacement axial piston pump, electric motor, pressure compensator, measuring valve, meter-in throttle edge, hydraulic motor, meter-out throttle edge, tubes, multiple tube connection elements, etc. All the components of the hydraulic load-sensing system have been described as NUT classes.

First, all the models of components had to be composed and tested separately. For this purpose, for each component a computing model was composed, input signals were chosen and finally,

action of the component was simulated to make sure everything is correct.

Second, the separately tested components were connected into more complicated computing models and tested in behaviour. At this and following stages, problems of growing difficulty rise. Typical is the appearance of feedback chains, which make dependencies between parameters essentially complicated and more difficult to observe.

Finally, model of the whole load-sensing system was composed

### EXAMPLE OF A COMPONENT

Simulation problem description for testing variable displacement axial piston pump PV is shown in Fig.2. The input values of pump pressure (4) and angle velocity of the electric motor (5) are given as constants and a range of values for position angle of the pump swash plate is given as static (1).

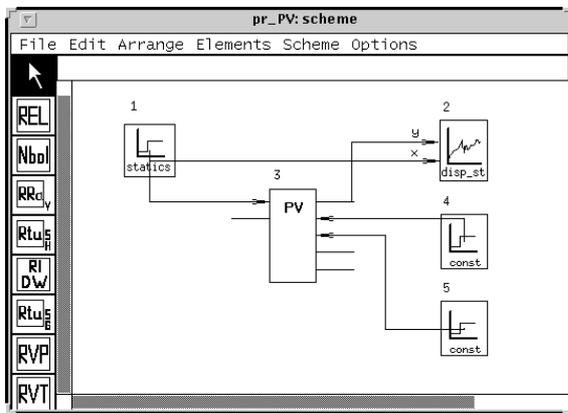


Figure 2.

The volumetric flow in dependence of the position angle of the pump swash plate is calculated, and then visualized by disp (2). The result is shown in Fig. 3.

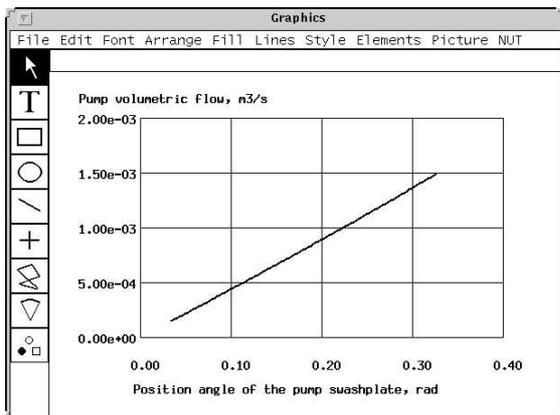
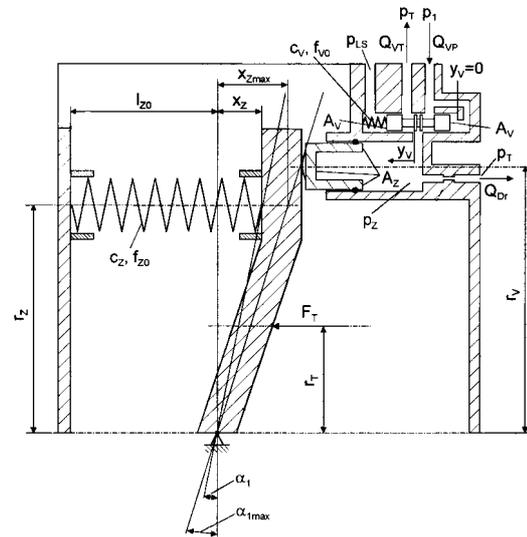


Figure 3.

We need to choose input parameters and parameters of the pump itself to make pump operate in the

manner required by other components of the whole system. A great number of different problems with different parameter values have to be solved to test the pump.

### EXAMPLE OF A SUBSYSTEM



The scheme of a hydraulic-mechanical controller (Fig. 4.) contains spool valve  $A_v$  with two slots, constant resistor  $Q_{dr}$ , positioning cylinder  $A_z$ , and swash plate with spring.

Figure 4.

Simulation problem description for testing hydraulic-mechanical controller together with pump and electric motor is shown in Fig. 5.

The scheme contains variable displacement axial piston pump  $PV$ , electric motor  $ME$  and pump controlling device which consists of pump control spool valve  $VP$ , inflow spool valve slot  $RVP$ , outflow spool valve slot  $RVT$ , interface element  $IEH$ , constant resistor  $REL$  and positioning cylinder  $ZV$ .

Input values load-sensing pressure (3) and controller output pressure (10) are given as constants and a range of values for pump pressure is given as statics (1).

In Fig. 6 the graph of pump volumetric flow depending on pump pressure is shown. The volumetric flow value is maximal if the difference of pump pressure and load-sensing pressure is less than approximately 15 bars. The volumetric flow value is minimal if the difference of pump pressure and load-sensing pressure is more than approximately 18 bars. In interval 15 to 18 bars the dependence is linear.

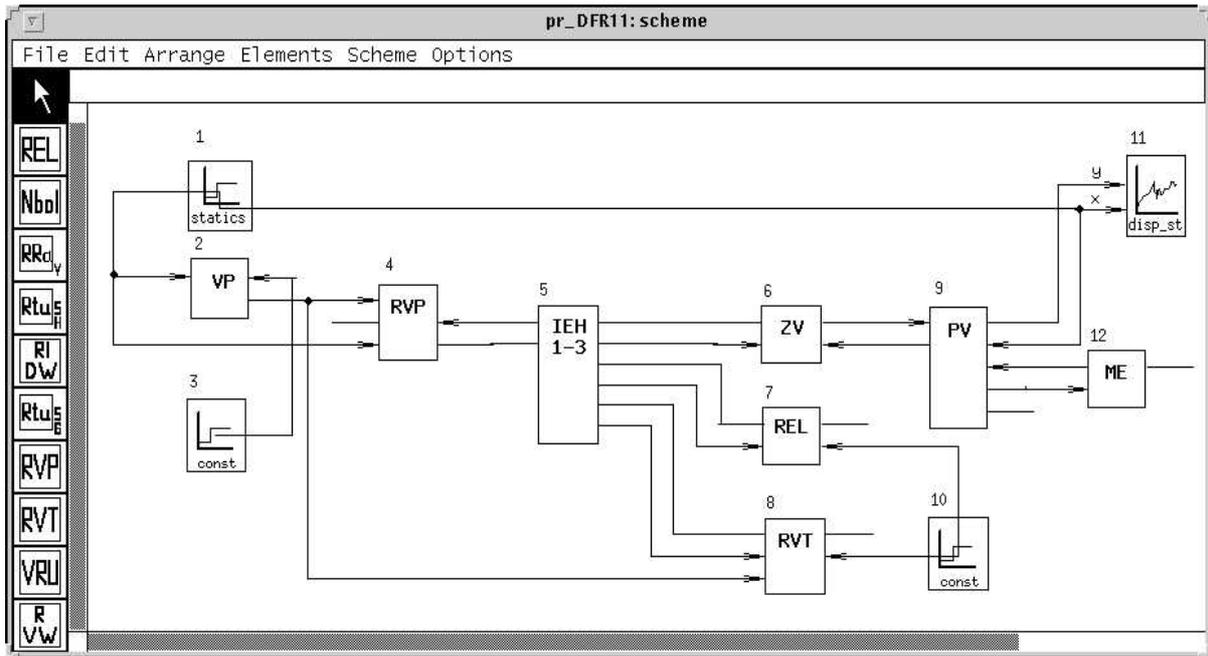


Figure 5.

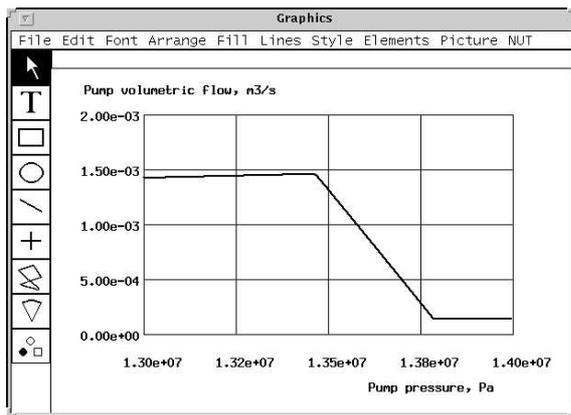


Figure 6.

All the subsystems must be simulated and tested separately. At this stage number of different schemes and problems grows considerably.

### MODEL OF THE WHOLE LOAD-SENSING SYSTEM

The simulation problem description of the whole load-sensing hydraulic system steady-state calculations is shown in Fig. 7.

The model contains 19 component models and several constants and static parameter components.

Elements 1-6,8,9,11 are the components of the subsystem, described above. Besides them the scheme contains relief valve (7), pressure compensator with measuring valve (14), meter-in throttle edge (21), hydraulic motor (22), meter-out throttle edge (23),

tubes (12, 13, 25), interface elements (10, 16), efficiency coefficients calculator (18), constant inputs (9, 15, 24), range input for displacement of the direction valve (19) and range input for hydraulic motor load moment (24).

In Fig. 8 the dependency graph of the efficiency coefficient of the whole system on the displacement of the direction valve in the case of constant value of hydraulic motor load moment is shown.

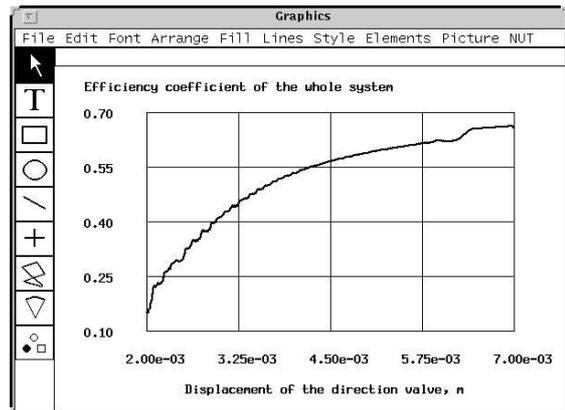


Figure 8.

All the simulations were performed on the SunUltra10 workstation in the UNIX environment.

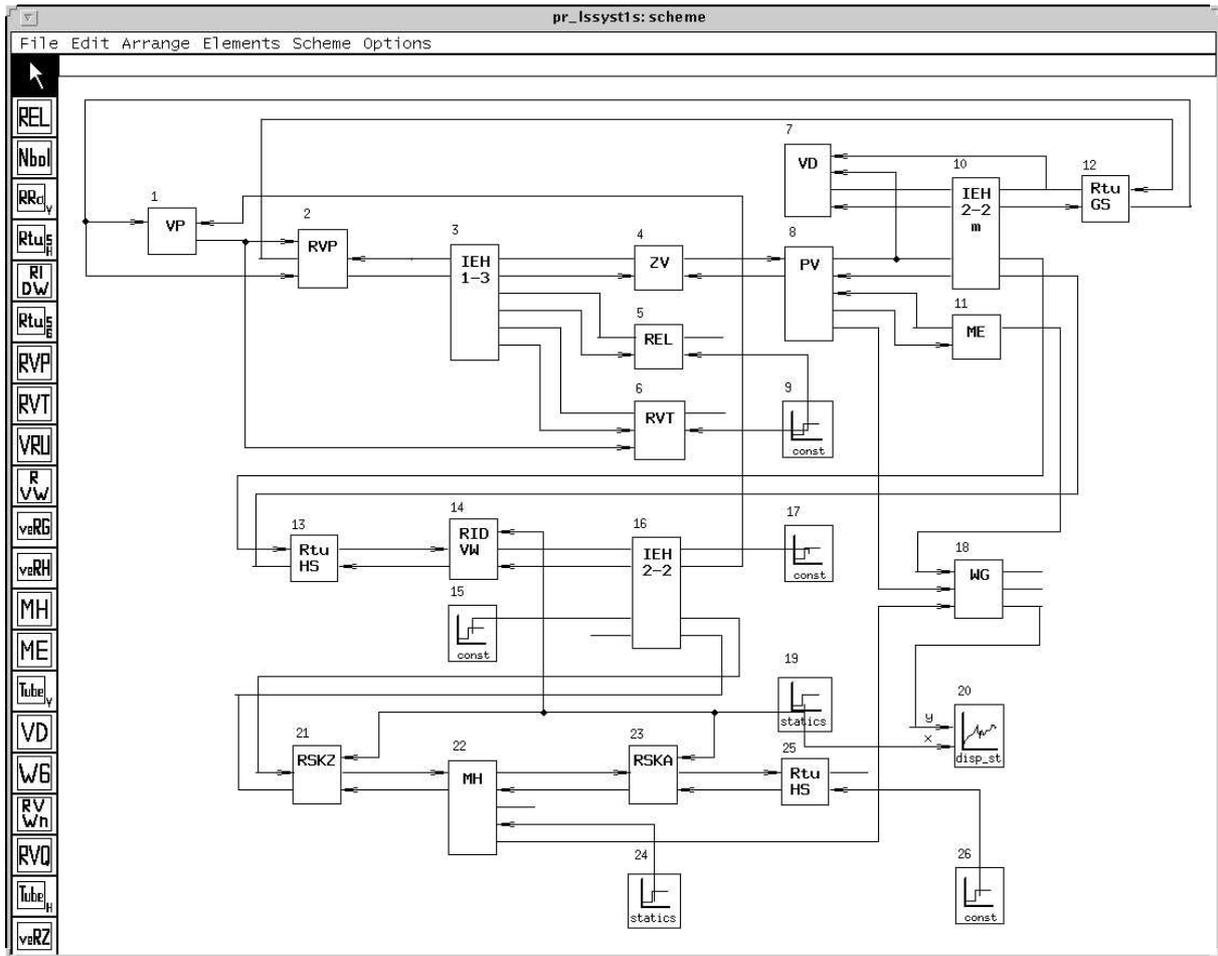


Figure 7.

## CONCLUSIONS AND FUTURE WORK

The approach used for simulation of hydraulic devices described above has the following main features:

- mathematical models of the functional elements are as multi-pole models, having various causality and take into account the signal propagation in both direction;
- mathematical model of the whole system carries the full information about connections of input/output variables, which express the considered mathematical causalities and guarantees the completeness of the mathematical model;
- modelling and simulation is built up in object-oriented way using the NUT programming environment, which enables to visualize and automate the simulation process;
- simulation is performed step by step, starting from simulation of components and moving to more complicated subsystems;
- iteration methods have been used in cases of loop dependencies which may appear if models are connected together into more complicated ones.

The hydraulic load-sensing systems require very precise parameter setting, especially for resistances

of hydraulic valve spools. Proposed in the paper simulation system enables quite fast and not expensive way to get optimal solutions to these problems.

In the future problems of modelling dynamics of such systems will be considered.

## ACKNOWLEDGEMENT

This research was partially supported by Estonian Scientific Foundation (Grant No. 5867).

## REFERENCES

- Grossschmidt G. 1997. Principles of Modelling and Simulation of Technical Chain Systems. Proc. of the 9<sup>th</sup> European Simulation Symposium & Exhibition, Simulation in Industry, October 19-23, 1997, Passau, Germany, pp. 705-712.
- Grossschmidt G., Harf M. 2003. Multi-pole Modelling and Simulation of an Electro-hydraulic Servo-system with a Non-linear Regulator in NUT Environment. Fünftes Deutsch-Polnisches Seminar "Innovation und Fortschritt in der Fluidtechnik", 18. -19. September 2003, Warszawa, pp. 146-161.

# SIMULATION OF DIFFUSION PROCESSES IN LABYRINTHIC DOMAINS BY USING CELLULAR AUTOMATA

Udo Buschmann and Thorsten Rankel and Wolfgang Wiechert  
Department of Simulation  
University of Siegen  
Paul-Bonatz-Str. 9-11, 57076 Siegen, Germany  
E-mail: buschmann@simtec.mb.uni-siegen.de

## KEYWORDS

internal corrosion, cellular automata, diffusion, obstacle modelling

## ABSTRACT

Components of gas turbines are exposed to different damaging factors during their lifetime. Besides the mechanical stress caused by the high centrifugal force another important attack is initiated by the surrounding gas atmosphere. At high temperatures gas molecules dissociate and the atoms diffuse inwards the metallic components and react with the solved alloy constituents. The reaction products precipitate inside the component and pose an obstacle for the succeeding gas atoms. In order to analyse this effect this process has been projected onto a cellular automaton. For that purpose an existing diffusion model has been selected and an extension to handle obstacles has been developed. This extension has been inspected thoroughly and its sensitivities regarding its parameters have been explored.

## INTRODUCTION

High temperature corrosion (HTC) is a damage process which occurs where metallic materials get into contact with gas atmospheres at high temperatures, e.g. gas turbines in power plants (Krupp 1999). This process can be divided into external and internal corrosion where the internal process is the critical one, since it can only be observed with difficulties.

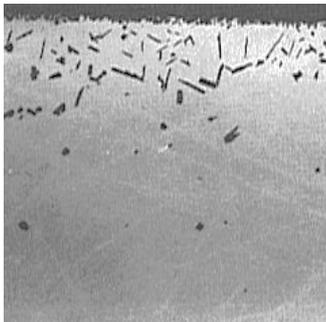


Figure 1: Cross-section of a nickel-base alloy with nitride precipitations

HTC can be regarded as an interaction of two elementary processes, inwards oriented diffusion of dissociated gas atoms on the one hand and the chemical reaction between these gas atoms and the alloy components on the other hand. The reaction products are oxides and nitrides that remain at the location inside the material where they precipitate. Figure 1 shows a cross-section of a specimen which evinces internal precipitations. These oxides and nitrides cause changes of the microstructure and this in turn can lead to failure of the attacked structure element.

The kinetics of this corrosion process is mainly determined by the gas diffusion towards the reaction zone. Since the high temperatures provide lots of thermal energy the chemical reaction accelerates and so the reaction takes place almost instantaneously. Thus the amount of available gas atoms is the primal factor that limitates the process. When the formation of internal precipitation has consumed all existing alloy ingredients until they are locally exhausted, the precipitation front moves on inwards. As a consequence the gas atoms have to diffuse a larger distance from the surface to reach the reaction zone and their courses are more and more bulked by precipitations.

From the macroscopic point of view this effect leads to a change of the diffusion velocity (Schnaas 1978). The aim of this work is to examine the diffusion in the region of the precipitations and to give an evidence for the influence of the oxides and nitrides on the gas diffusion. It is intended to express this influence as an effective diffusion coefficient inside the precipitation zone. The heterogenous occurrence of oxides and nitrides will be represented by a homogenized coefficient. By using this coefficient in a homogenized domain the influence of the microscopic obstacles could be regarded in a macroscopic way. With that it is possible to transfer various geometries, locations and orientations of the precipitations into one system depended quantity.

## DIFFUSION MODEL

Diffusion is a kinetic process which reduces the concentration gradient of chemical components inside a phase.

Contrary to the diffusion in liquid media, where the diffusion is not restricted, the diffusion in solids, particularly with regard to metals, is adapted to the atomistic structure. Regarding the diffusion microscopically there are some elementary mechanisms that govern the transport in atomistic scale. The most important mechanisms are interstitial, ring and vacancy-assisted diffusion. From the macroscopic point of view the mass flux can be expressed as a vector field  $\mathbf{j}$  as noted in Equation (1). Its magnitude is proportional to that mass quantum which flows during a unit time through a unit area. The constant coefficient  $D$  describes the extent of the flux caused by the concentration gradient:

$$\mathbf{j} = D \nabla c \quad (1)$$

Regarding the mass conservation in a finite volume element and assuming that in the domain there is neither a mass sink nor a source, the divergence of the mass flux is the only term which changes the concentration inside the volume, as expressed in Equation (2):

$$\frac{\partial c}{\partial t} = -\nabla \cdot \mathbf{j} \quad (2)$$

Combining the mass conservation and the mass flux leads to Fick's 2<sup>nd</sup> law (3), relating the temporal variance of the concentration to the second spatial derivation of the concentration profile. In cases of isotropic diffusion the factor  $D$  can be reduced to a scalar coefficient, otherwise it becomes a diffusion tensor.

$$\frac{\partial c}{\partial t} = D \Delta c \quad (3)$$

Since Fick's 2<sup>nd</sup> law is a partial differential equation with only a few analytical solutions available for special cases, numerical methods have to be used to apply it to technical problems. Nevertheless one significant analytical solution needs to be presented here (Glicksman 2000). Regarded is a 1-dimensional half space. At the left boundary the constant concentration is  $c_\Gamma$  and the initial concentration on the domain is assumed to be zero. For these conditions an analytical solution of Fick's 2<sup>nd</sup> law is given by Equation (4):

$$c(t, x) = c_\Gamma \cdot \operatorname{erfc}\left(\frac{x}{2\sqrt{Dt}}\right) \quad (4)$$

This time- and position-dependent solution will be used to validate the implementation of the cellular automaton and to evaluate the results of the simulations.

## EXPERIMENTAL PROCEEDING

To describe the diffusion in the range of the precipitations in a macroscopic way an effective diffusion coef-

ficient has to be developed. It has to represent the diffusion conditions inside the region of obstacles. The diffusion across this range will be modelled by means of cellular automata. Performing simulations with this method will result in concentration profiles across the precipitation zone as well as the undisturbed area behind it. By fitting the parameter  $D$  of the analytical solution in Equation (4) with the concentration profile in region of obstacles an effective diffusion coefficient  $D_{\text{eff}}$  can be determined. Relating this coefficient to the coefficient  $D$  of the undisturbed region leads to the labyrinth factor  $\Gamma$  as introduced in Equation (5).

$$\Gamma = \frac{D_{\text{eff}}}{D} \quad (5)$$

This quantity expresses the extent of obstruction caused by the precipitations. Since it is impossible to get universal results for all possible geometries of obstacles with this method the simulations will be performed with selected test configurations. However, these results can be interpreted as indicators concerning the diffusion behaviour in technically relevant systems.

## CELLULAR AUTOMATA

Cellular automata are mathematical models which describe the dynamics of spatially distributed systems with simple rules on given domains (Gaylord 1996). Hence, they can be alternatively applied in those disciplines where up to now only partial differential equations have been used in combination with FEM. However the cellular automata are more advantageous concerning to complex geometrical structures and the possibility of handling obstacles and several phases only by changing the policies.

A sophisticated approach to simulate diffusion process with means of cellular automata was introduced in (Chopard and Droz 1998). In this model the cells can be occupied by a maximum of 4 particles provided that the regarded domain is 2-dimensional. In a 3-dimensional domain this number increases to 6 corresponding to the number of closest neighbour-cells. The discrete values specify the number of particles inside a cell. A fundamental problem of this approach is to guarantee that the number of particles inside a cell is not more than the admitted one. To elude these difficulties the iterating evaluation is divided into the two phases mixing and moving.

In the first phase the incoming particle vectors are processed. From this starting configuration the sequence of outwards oriented particle vectors can be determined for each cell as permutations of those which came in. In 2 spatial dimensions  $(2 \cdot 2)! = 24$  permutations are possible in principle. However this approach is restricted to 4 configurations as shown in Figure 2.

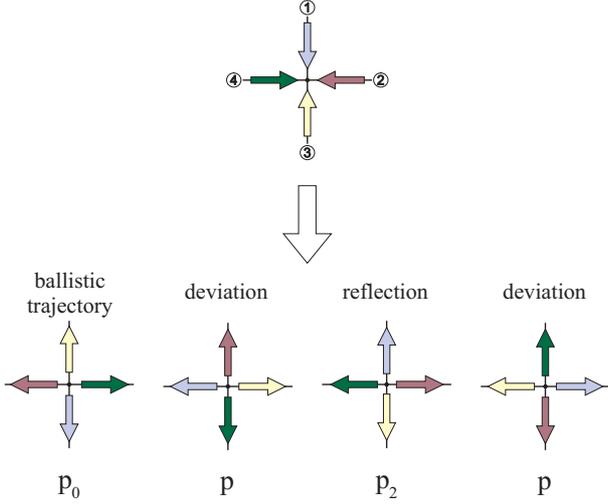


Figure 2: Possibilities of particle movement in the model of Chopard and Droz

The probability of these potential permutations can be weighted separately. It is postulated that the probability of a  $90^\circ$  deviation is independent of the rolling direction and so the two parameters for both possibilities can be combined together to only one called  $p$ . In addition the probability of a ballistic trajectory is denoted as  $p_0$  and for reflection with  $p_2$ . Summing up all probabilities must result in 1. Another constraint is that none of the probabilities is allowed to be zero.

$$p_0 + 2p + p_2 = 1, \quad p, p_0, p_2 > 0 \quad (6)$$

With these parameters it is possible to influence the behaviour of the cellular automaton considerably. Increasing  $p_0$  and consequently decreasing  $p$  and  $p_1$  leads to a high velocity of the diffusing particles. Chopard succeeded to show that the cellular automaton indeed converges to a Fick diffusion process when both the spatial increment  $\Delta x$  and the time step  $\Delta t$  tend to zero. Moreover he derived a functional correlation between the set of probabilities and the diffusion coefficient as shown in Equation (7):

$$D = \lim_{\Delta t \rightarrow 0, \Delta x^2 \rightarrow 0} \left( \frac{\Delta x^2}{\Delta t} \cdot \frac{p + p_0}{4(1 - p - p_0)} \right) \quad (7)$$

With validity of Equation (6) in mind, the probabilities  $p$ ,  $p_0$  and  $p_1$  can be varied. By this means the diffusion coefficient of the particles can be adjusted without changing the grid and time step.

## VERIFICATION OF THE IMPLEMENTED CELLULAR AUTOMATA

For this approach the comparison between the cellular automaton results and the analytical solution (4) of the diffusion equations has been carried out. For this pur-

pose a 2-dimensional cellular automaton has been implemented in MATLAB. With this the configurations as discussed for the analytical solution (4) have been imitated and simulations have been performed with 100 repetitions. To make a comparison between the 2-dimensional results with the 1-dimensional analytical solution possible, the average concentrations depending to the distance from surface have been evaluated. In Figure 3 the different states of the cellular automaton and the according analytical solutions are shown.

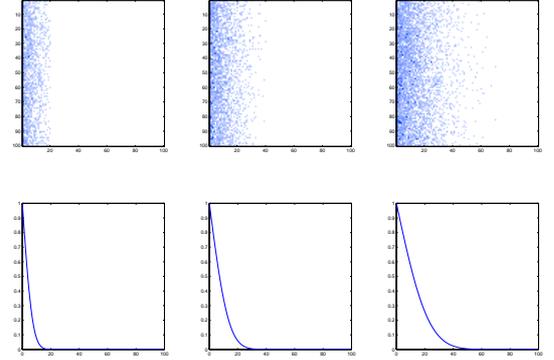


Figure 3: Comparison between the analytical solution and the output of the cellular automaton

To analyse the results statistically the concentration profile of the last time step is regarded again. In addition to this the 98% confidence interval has been exemplary computed and plotted in Figure 4 for this simulation series. It shows the narrow confidence range that widens a bit far from surface. This can be led back to the numerical inaccuracy.

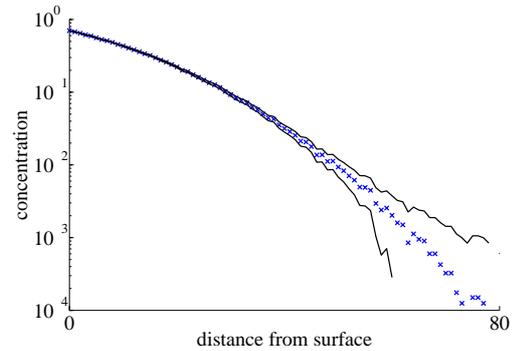


Figure 4: CA- results with its 98% confidence range

## EXTENSION TO DOMAINS WITH OBSTACLES

Since the basic model of Chopard and Droz does not consider the treatment of obstacles, new rules to handle those disturbances of diffusion were added. It is important that the rules of the cellular automata were only supplemented with this new aspect, but not changed. So the proven physical correctness in the free volume is maintained. In the presence of one adjacent obstacle four cases have to be discerned as shown in Figure 5.

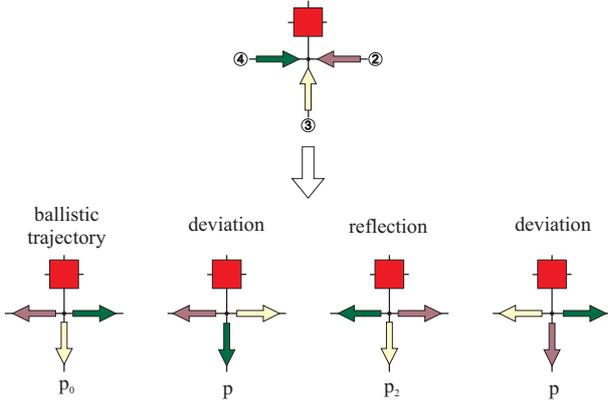


Figure 5: CA policy concerning one adjoining obstacle

To follow the exclusion principle only three particles are admissible in the regarded cell. Analogous to the basic model the output directions are determined by the permutations of the particle vectors in conformity with their probabilities. If two obstacles are adjoining to a cell the only two different possible configurations are illustrated in Figure 6.

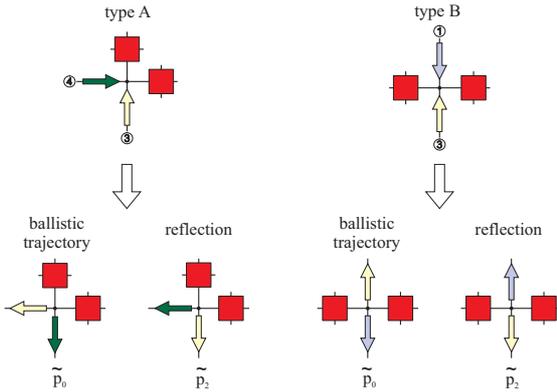


Figure 6: CA policies concerning two adjoining obstacles

In the first scene constituted as type A two obstacles are close to each other. Even though a straight-line course is impossible in this case, the only potential permutation is still called ballistic trajectory. The other arrangement of two obstacles is an opposition denoted as type B. Here it is clear, that no deviation can take place. As a consequence the probabilities for the admitted particle vectors have to be recalculated as realized in Equation (8):

$$\tilde{p}_0 = \frac{p_0}{p_0 + p_2}, \quad \tilde{p}_2 = \frac{p_2}{p_0 + p_2} \quad (8)$$

The last two remaining configurations are quite straightforward. If three obstacles are neighbouring a cell, the only possible particle vector will be inverted. And finally in the case of four obstacles no particle can get into the cell.

## PARAMETER VARIATIONS

The aim of the following investigations is to ascertain the influence of the numerical and physical parameters on the simulated diffusing behaviour. For this purpose a special 2-dimensional test arrangement has been designed. At the initial state only blocks of obstacles are arranged in the examined domain. They are assumed to be immobile. The boundary conditions for the bottom and the top are supposed to be cyclic. The right boundary is determined to be isolated and at the left hand side a constant density of particles is defined. This can be effected by deleting the actual content of the cells in the first column and assigning them by a random generator adjusted with the required probability at the beginning of each iteration. Figure 7 shows the final state of the cellular automaton.

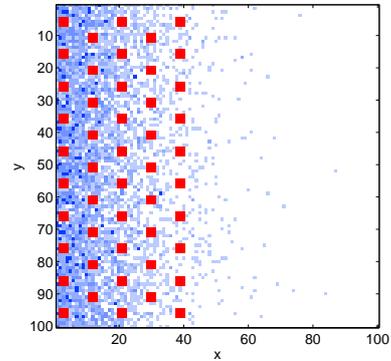


Figure 7: Diffusion CA with obstacles

In the range of the obstacles the diffusion is hampered. To quantify this effect an effective diffusion coefficient for this range has to be determined as explained below.

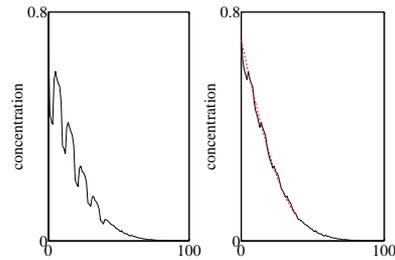


Figure 8: Evaluation of raw CA results. 8a (left): Without correction. 8b (right): Corrected for reduced volume due to obstacles

Firstly the raw concentration profiles as shown in Figure 8a have to be corrected. In the first evaluation step the obvious defect caused by reduced volume due to the obstacles has to be corrected. This can be adjusted by relating the average concentrations to the actual area the particles have at their disposal for diffusion. With these corrected values the parameter  $D_{\text{eff}}$  can be calculated with means of the least square method in the range of obstacles as shown in Figure 8b. With this quantity a

labyrinth factor  $\Gamma$  can be calculated as shown in Equation (5). It relates the measured effective diffusion coefficient with the adjusted one.

## SIMULATION EXPERIMENTS

Regarding Equation (7) again it is apparent, that there are in principle two different ways to determine the diffusion coefficient  $D$ . At first the ratio of the time step width  $\Delta t$  and the spatial increment  $\Delta x$  can be varied. Increasing  $\Delta x$  leads to an increasing diffusion coefficient and analogue decreasing  $\Delta t$ . Since the implemented cellular automaton consists of a fixed grid the variation of this quotient will be performed by changing the time step size. The second way to adjust the diffusion coefficient is to change the relation between the probabilities  $p$ ,  $p_0$  and  $p_2$  when the range of settings is restricted by Equation (6).

In order to analyse the consequences of adjusting in both ways several simulations have been carried out. In a first test series the diffusion coefficient has been determined with three different time step sizes. To get comparable results the number of time steps  $n_t$  for a given simulation duration have to be adapted. In the second test series the same diffusion coefficients as above have been adjusted by means of varying the set of probabilities. In Table 1 the results of the performed simulations are shown.

Table 1: Varying diffusion coefficients with the time step size and set of probabilities

D	200	400	600
$n_t$	800	1600	2400
$\Gamma$	0.81	0.79	0.77
$p_0$	0.1	0.4	0.6
$\Gamma$	0.96	0.83	0.70

Comparing the results a tendency is obvious in both tests. An increasing diffusion coefficient effects a decreasing  $\Gamma$ . Considering the ratio in Equation (5) this means, that the extent of obstruction increases with a increasing velocity of the diffusing particles. When this effect can be observed rather slightly by varying the time step size it is enormous in the case of changing the set of probabilities. To investigate this sensitivity a next test series has been performed. Now the diffusion coefficient has been hold constant but the relations between the probabilities have been modulated as presented in Table 2.

Table 2: Constant diffusion coefficients with several sets of probabilities

D	400	400	400
$p_0$	0.10	0.25	0.40
$p$	0.40	0.25	0.10
$\Gamma$	0.94	0.88	0.79

This test indicates a strong influence of the probabilities. Even though the diffusion coefficient as computed by Equation (7) is identical in all cases the resulting labyrinth factors  $\Gamma$  differ. Apparently a decreasing probability  $p$  causes an increasing hindrance of diffusion. To illustrate the consequences of this fact another test has been executed. For that the diffusion coefficient has been fixed constant again and the probabilities of a ballistic trajectory  $p_0$  and of reflection  $p_1$  have been assumed to be equal. Now the parameter  $p$  has been systematically varied in a range from  $10^{-4}$  to  $10^{-2}$  and the other parameters have been adapted. For this test another configuration has been used. All boundaries have been assumed to be isolated and at the initial state a square of particles has been centred on the domain. In this test are no obstacles present. Figure 9 shows the computed results at different times.

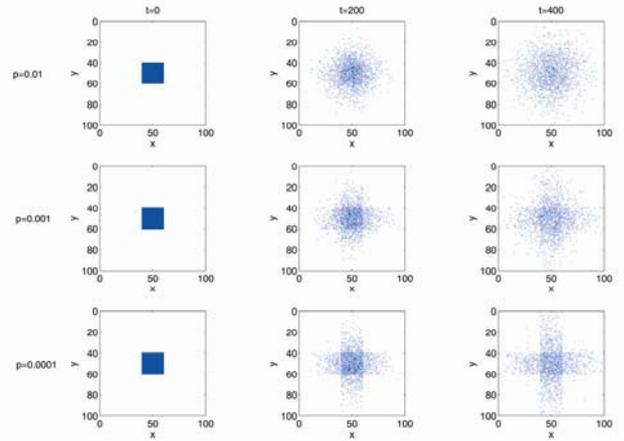


Figure 9: CA behaviour depending on probability set

While the probability  $p = 10^{-2}$  leads to an isotropic diffusion in the other cases a preferred diffusing direction is identifiable. This effect becomes clear by considering the free path length of a particle. The parameter  $p$  determines the probability of a  $90^\circ$  deviation. In principle increasing  $p$  leads to a decreasing free path length. Figure 10 shows the comparison of two particles with different values of  $p$ .

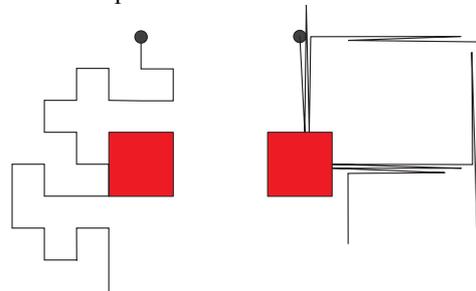


Figure 10: Particles passing obstacles

While the particle on the left with the higher value of  $p$  often changes its direction and so easily passes the obstacle, the one on the right side bounces several times against the obstacle until it finally turns and passes it

also. Since the assumption that  $\Delta x$  has to tend to zero in Equation (7) is not complied in this case, it has to be interpreted from a different point of view. To compare the diffusion behaviour of particles with different sets of parameters the regarded domain has to be resolved with a sufficiently fine grained net. Refining the grid leads in this case to a convergency of the results. In other words, the simulation is too far away from the critical limit in Equation (7).

To underline this, a further test has been carried out. For this the geometry of the obstacles has been changed, they became larger. In order to get comparable results their number has been decreased and so density of obstacles regarding the whole domain has not changed. In Table 3 the outcome of this test is presented.

Table 3: Refinement of the CA

D	400	400	400
$p_0$	0.10	0.25	0.40
p	0.40	0.25	0.10
$\Gamma$	0.83	0.80	0.77

This series confirms the thesis that the resolution of the grid has to be sufficient fine in relation to the obstacles. Applying larger obstacles in a constant grid corresponds to a reduction of the regarded area resp. increasing the spatial increment  $\Delta x$ . The tendency conforms to the results presented in Table 2 but the extent has significantly decreased. Finally, a series has been performed to figure out the influence of the boundary concentration  $c_\Gamma$ . Several simulations have been executed with different diffusion coefficients adjusted by the time step size and various boundary concentrations. Table 4 shows the outcome of these tests.

Table 4: Effects of the boundary concentrations

		D		
		200	400	600
$c_\Gamma$	0.1	0.81	0.79	0.77
	0.5	0.81	0.79	0.77
	0.7	0.81	0.79	0.77

As a result can be determined, that the sensitivity of the cellular automaton regarding the boundary concentration can be neglected.

## CONCLUSION

In this paper an extension of a cellular automaton that treats diffusion processes has been developed. A short overview of the existing model has been given. After that the development of the policy extension was discussed in detail. We have exposed the influence of all concerning parameters and their technical meaning. In order to quantify the effect of obstacles the labyrinthic factor  $\Gamma$  has been introduced. With that special adjusted

simulation experiments have been performed to investigate the sensitivity of the cellular automaton concerning the numerical parameters. It has been observed that a sufficient resolution has to be the basis for the simulations even though the experimental results seem to be plausible. The performed investigations have shown that the developed extension of the diffusing model is considered suitable to be applied.

## REFERENCES

- Chopard, B and Droz, M. 1998. "Cellular automata of physical systems." *Cambridge University Press*. Cambridge.
- Gaylord, R. 1996. "Modeling nature." *Springer*. New York
- Glicksman, M. 2000. "Diffusion in solids." *John Wiley and sons*, New York.
- Krupp, U and Christ H.-J. 1999. "Internal Nitridation of Nickel-Base Alloys. Part I: Behavior of Binary and Ternary Alloys of the Ni-Cr-Al-Ti-System". *Oxidation of Metals*. Vol. 52, Nr. 3, pp 277-298.
- Schnaas, A. and Grabke H.J. 1978. "Changes in material properties of austenitic CrNiFe-alloys by carburization." *Werkstoffe und Korrosion*, Vol. 29, pp 635-644.
- Zhou, L. and Xiuqin, W. 1999. "A randomwalk-cellular automaton simulation of internal oxidation and its transition to external oxidation." *Scripta Materialia*, Vol. 40, Nr. 3, pp 365-374.

## Author Biographies



**UDO BUSCHMANN** studied mechanical engineering at the University of Siegen, Germany from 1995 to 2001. After his Diploma he became a scientific coworker at the Chair of Simulation at department of systems engineering. Actually he is working on his doctorate thesis.



**THORSTEN RANKEL** studies mechanical engineering at the University of Siegen. After his student research project at the Institute of Systems Engineering he has been working on his diploma thesis at the Robert Bosch GmbH in Stuttgart.



**WOLFGANG WIECHERT** studied mathematics and computer science at the University of Bonn and obtained his diploma degree in 1985. Afterwards he joined the Department of Theoretical Biology at the University of Bonn where he got his PhD in 1991. From 1991 to 1996 he worked at the Institute of Biotechnology at the Research Center Jülich where he earned his postdoctoral lecture qualification. Since 1996 he is a professor for simulation at the Institute of Systems Engineering at the University of Siegen.

# MICRO ARRAY DATA ANALYSIS BASED ON BUSINESS OBJECTS AS PART OF A WORKFLOW RELATED GENE EXPRESSION

Dietmar P. F. Moeller  
University of Hamburg, Department Computer Science  
Chair Computer Engineering, AB TIS  
Vogt Kölln Str. 30, D-22527 Hamburg, Germany  
dietmar.moeller@informatik.uni-hamburg.de

## ABSTRACT

The paper presents the application of Computational Modeling and Simulation in the Science and Engineering (CMSSE) domain of gene expression. Applied to micro array data analysis one may introduce soft computing algorithm for CMSSE as part of sensitization or preprocessing. While sensitization enable the conditioning of case study specific classifiers, scientists are able using specific linguistic IF-THEN rules to create appropriate fuzzy sets, that can be helpful for use in micro array data analysis. Henceforth, scientists will be empowered handling this classifiers in situ, phased to their measuring equipment and/or case study specific parameters, under case investigation.

## INTRODUCTION

We live in a world of an imposing complexity and variety, a world where events more or less never repeat exactly. Human-world interaction based on a scientific approach seems to have a normalization in models, an abstract representation, as a powerful tool to understand real world phenomena. Therefore a big part of scientific work consists in formalization, which yields models of real world systems studied. This task clearly is scientifically oriented, in the sense to gain sufficient understanding of real world phenomena, generating respective representations, based on experiments and observations. Because the scientist attempts to create representations and laws that formalize verified hypothesis concerning real world phenomena.

The formalizations are only useful if they succeed in seizing the essential features of the real world. They permit extrapolation, that allows to generalize, often correctly, from past experience to future events from which we can learn how the real world system can be manipulated for own purposes, which is a kind of uncertainty. In our world which is more or less precise understandable or predictable, we are more conscious of uncertainty, that appears in form of imprecision, vagueness and ill defined, ill separable, and doubtful data. For this kind of uncertainty, or better soft information, we have to learn to understand the intrinsic systems dynamic.

Keeping formalizations running or doing extrapolation, deals with effectively information processing,

which is a task, done by computing machines, has been directly introduced as a suitable tool of the scientific approach. But using non-precise information, which is called soft information, e.g. the blood pressure is lower than normal, represented by linguistic terms such as low, high, small, medium, large, big and so on, needs a specific form of computation, which is called soft computing. Soft computing, deals with fuzzy sets, neuronal nets, genetic algorithms, evolution strategy, probabilistic methods etc. Usually, these approaches in uncertainty, combining soft information with conventional scientific methods in a so called ad hoc manner, can be investigated using computation to show the validity of the approaches in relation to the specific case study. Therefore, during the past years, processing of uncertainty or soft information processing had been applied by different disciplines for a large variety in formal representations in the several scientific application domains. Applying soft computing techniques for those formalizations, one can impart an understanding that the formalization itself can not provide. Because soft computing is a collection of methods which can be expressed in terms of algorithms, belonging to the respective disciplines, that has been proved to be of vital importance to progress in all fields of endeavor.

In general, the common problems arising from formalization in science, and especially the possibility of applying in a wide range of scientific research the same methods while solving problems, has improved the cooperation between different disciplines and removed the rigid barriers of the past between them. However, although it seems that engineers and scientists, like physicians, will have the same goals in studying their systems. But there is still an essential difference between them, formalizing a real world research problem. For instance, the engineer is interested in system formalization reflecting normal operating conditions. His aim is to use the model in case of normal operating conditions, e.g. for optimized system control, or at least to keep it in a relative close vicinity of safe operating conditions and avoid the danger resulting from the formalized system running out of control. Anyhow there is no outstanding interest for engineers formalizing plants behavior outside its allowable operating conditions range.

In contrast, the scientist, like the physician, is not solely interested in formalization a real world problem

under normal conditions. He would prefer that the formal description adequately describes the systems un-nominal behavior, i.e. the systems behavior outside of normal limitations, like pathological states in case of hypertension versus normotension, or gene expression analysis in relation to the type or growth of a tumor, and there are serious limitations. But engineering techniques dealing with uncertainty are sometimes as much statements comparable to cognitive and linguistic sciences as they are about engineering, and hence they are comparable to science, like medicine.

In practice, the formalization of models itself is an iterative process, consisting of measurements at the real world system –if possible–, and computing strategies by changing the structure of the formal description in an effort to closely match the complex dynamic systems behavior. The computing strategy may be based on the category in the nearest neighbor sense, if the adapted representation is close enough to the previous one. In fact the formalization has served its purpose when an optimal match is obtained between computed results and data obtained from the real world system under test.

Soft information processing generate the basic insight that categories are not absolutely clear cut, they belong to lesser or greater degree to that category. Hence soft computing systems break with the tradition, that real world phenomena can be precisely and unambiguously characterized, which means divided into categories, and then manipulated according to precise and formal rules. From the mathematical point of view soft computing means multi-valuedness or multi-valence. Logical paradoxes and the Heisenberg uncertainty principle led to the development of multi-valence, and in the 1930s quantum theorists allowed for indeterminacy by including a third truth value in the bivalent logical framework. Systems scientist Zadeh in 1965 introduced the term fuzzy into the technical literature, and inaugurated a second wave of interest in multivalued structures –from systems to topologies– extending a bivalent indicator function  $i_A$  of non-fuzzy subset  $A$  of  $X$  to a multi-valued indicator or membership function  $m_A: X \rightarrow [0,1]$ . This allows to combine multi-valued or fuzzy sets with point-wise operators of indicator functions for the large variety of fuzzy systems.

## SOFT COMPUTING SYSTEMS APPLIED FOR PREPROCESSING

Soft computing systems is a name for systems with directly relationship to soft computing concepts like fuzzy sets, neuronal nets, genetic algorithms etc. The soft computing concepts based on fuzzy sets can be classified into pure fuzzy systems, Takagi and Sugeno fuzzy systems and fuzzy systems with fuzzification and defuzzification.

### Pure Fuzzy Systems

The basic configuration of a pure fuzzy systems is based on a fuzzy rule base that consists of a collection of fuzzy IF-THEN rules, and the fuzzy inference engine that uses these fuzzy IF-THEN rules in order to determine a mapping output universe of discourse  $U \subset \mathbb{R}^n$  to fuzzy sets in the output universe of discourse  $V \subset \mathbb{R}$  based on fuzzy principles. Fuzzy IF-THEN rules are of the following form:

$$R(k): \text{ IF } x_1 \text{ is } F_1^{(k)} \text{ AND } \dots \text{ AND } x_n \text{ is } F_n^{(k)} \text{ THEN } y \text{ is } G^k \quad (1)$$

where  $F_i^{(k)}$  and  $G^{(k)}$  are the respective fuzzy sets,  $x = (x_1, \dots, x_n)^T \in U$  and  $y \in V$  are input and output linguistic variables, respectively, and  $k = 1, 2, \dots, m$ .

Each fuzzy IF-THEN rule defines fuzzy set  $F_1^{(k)} \times \dots \times F_n^{(k)} \rightarrow G^{(k)}$  in the product space  $U \times V$ . Let  $A$  be an arbitrary fuzzy set in  $U$ , then the output determined by each fuzzy IF-THEN rule of equation (1) is a fuzzy set  $A \circ R^{(k)}$  in  $V$  whose membership function is

$$\mu^{A^{(k)} \circ R^{(k)}}(\mu) = \sup_x \mu U [\mu A^{(x)} * \mu F_1^{(k)} \times \dots \times \mu F_n^{(k)} \rightarrow G^{(k)}(x,y)] \quad (2)$$

with  $*$  as operator such as MIN, MAX, PRODUCT, or others.  $\mu A$  is used to represent the membership function of a fuzzy set  $A$ .

The final output of a pure fuzzy system is a fuzzy set  $A \circ (R(1), \dots, R(m))^{(k)}$  in  $V$ , a combination of the respective fuzzy set. Hence a pure fuzzy system constitutes the essential part of fuzzy systems as a general framework in which linguistic information is quantified and fuzzy principles are used to realize systematic use of linguistic information.

### Takagi and Sugeno Fuzzy System

Instead considering fuzzy IF-THEN rules in form of equation (1), Takagi and Sugeno in 1985 proposed using fuzzy IF-THEN rules in the form:

$$L^{(k)}: \text{ IF } x_1 \text{ is } F_1^{(k)} \text{ AND } \dots \text{ AND } x_n \text{ is } F_n^{(k)} \text{ THEN } y^k = c_0^k + c_1^k x_1 + \dots + c_n^k x_n \quad (3)$$

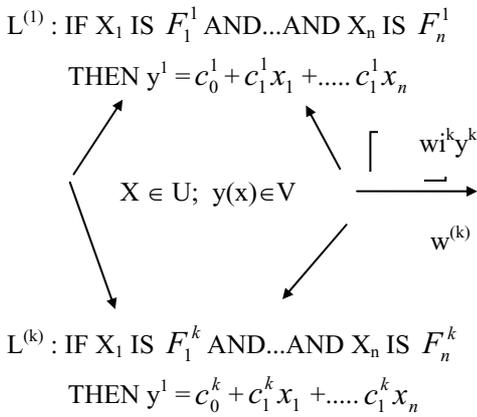
where  $F_i^{(k)}$  are fuzzy sets,  $c_i$  are real-valued parameters,  $y^{(k)}$  is the Takagi-Sugeno fuzzy system output due the rule  $L^{(k)}$ , and  $k=1, 2, \dots, m$ . That is, they considered rules whose IF part is fuzzy but whose THEN part is crisp. For a real-valued input vector  $x=(x_1, \dots, x_n)^T$ , the output  $y(x)$  of Takagi and Sugeno fuzzy systems is a weighted average of  $y^{(k)}$ :

$$y(x) = \frac{\sum_{\ell=1}^k w^{(\ell)} y^{(\ell)}}{\sum_{\ell=1}^k w^{(\ell)}} \quad (4)$$

where weight  $w^{(k)}$  implies the overall truth value of the premise of rule  $L^{(k)}$  for the input and is calculated as

$$w^{(k)} = \prod_{i=1}^n \mu_{F_i^{(k)}}(x_i) \quad (5)$$

which is shown in the following representation.



### Fuzzy Systems with Fuzzification and Defuzzification

Compared with the pure fuzzy system we may add a fuzzifier to the input and a defuzzifier to the output of the pure fuzzy system. The fuzzifier maps crisp points in  $U$  to fuzzy sets in  $U$ , and the defuzzifier maps fuzzy sets in  $V$  to crisp points in  $V$ . The fuzzy rule base as well as the fuzzy inference engine are the same as those in pure fuzzy logic systems, as shown in Figure 1.

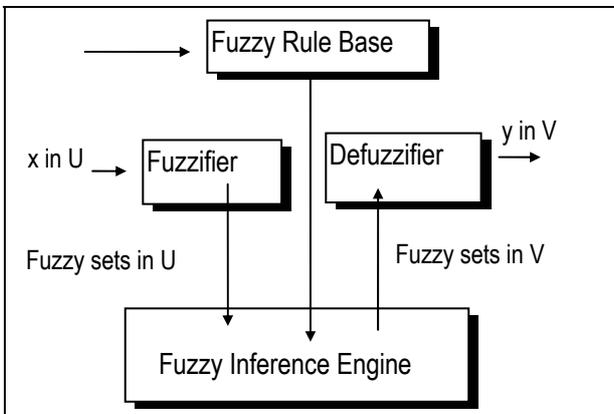


Figure 1: Fuzzy system with fuzzifier and defuzzifier

### Neuronal Nets

A neural net consists of input variables and weighting factors, activation layers and output variables. The physiological pendant of the inputs are the dendrites as part of the anterior motoneurons extend for one-half to one millimeter in all directions from the neuronal soma. Therefore, these dendrites can receive signals from a fairly spatial area around the motoneuron. This provides vast opportunity for summation of signals from many separate presynaptic neurons. The weighting functions physiological pendant are the synapses. The synapse could be interpreted as the juncture between one neuron and the next, based on three major parts, the soma, which is the main body of the neuron; a single axon, which extends from the soma into the peripheral nerve; and the dendrites, which are thin projections of the soma that extend up to one millimeter, into the surrounding areas of the cord. The output has its physiological pendant in the axon, which is the central core of a nerve fiber. The biological neuron and the artificial neuron are shown in Figure 2.

From Figure 2.b one can assume that the synapses of an artificial neuron receive an activation  $x_i$  with a specific strength  $w_i$  from another artificial neuron, which will be part of the summing process of the output, the axon. The basic concept results in an input vector  $\mathbf{x} = (x_1, \dots, x_n)^T$ , a weighting vector  $\mathbf{w} = (w_1, \dots, w_n)^T$  and the resulting activity as sum of the weighted input, which could be assigned as activity function  $z$ :

$$z(\mathbf{w}, \mathbf{x}) = \sum_j w_j x_j = \mathbf{w}^T \mathbf{x}$$

Often there exists a threshold, which has to be passed, to activate the output. Modeling the threshold results in the relation

$$z(\mathbf{w}, \mathbf{x}) = \mathbf{w}^T \mathbf{x} - T$$

where  $T$  indicates the threshold. Assuming

$$\mathbf{x} \rightarrow \mathbf{x} = (x_1, \dots, x_n, 1)^T$$

and

$$\mathbf{w} \rightarrow \mathbf{w} = (w_1, \dots, w_n, -T)^T$$

we receive the scalar product

$$z(\mathbf{w}, \mathbf{x}) = \sum_j w_j x_j - T = (w_1, \dots, w_n, -T) (x_1, \dots, x_n, 1)^T = \mathbf{w}^T \mathbf{x}$$

which can be rearranged as follows

$$z = \mathbf{w}^{(0)} + \sum_i \mathbf{w}_i^{(1)} x_i$$

whereby the notation  $^{(k)}$  indicates the correlation's of the  $x$  components

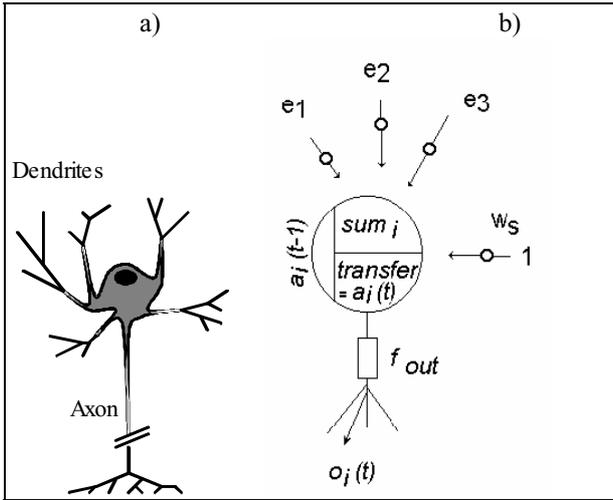


Figure 2: Biological neuron (a) and artificial neuron (b); for more details see text

Modeling high-order synapses then can be directly derived from the above equations as follows:

$$z = w^{(0)} + \sum_i w_i^{(1)} x_i + \sum_{ijk} w_{ijk}^{(2)} x_j x_k + \sum_{ijkl} w_{ijkl}^{(3)} x_j x_k x_l + \dots$$

This type of artificial neurons are the so called sigma-pi-units.

The output of an artificial neuron can be described by the function  $S(\cdot)$  depending on the internal activity  $z$ ,

$$y = S(z)$$

The complete output of an artificial neuron hence can be stated as transfer function of type  $y$

$$y = f(\mathbf{x}, \mathbf{w}, z, S).$$

## THEORY OF SENSITIZATION AS PART OF PRE-PROCESSING

The idea of Sensitization can be found in cognitive psychology in the context of chunking. Chunking is more or less the adaptation of a new fact or a so far unknown situation with the help of knowledge facts or models. Only out of old facts or acting strategies one can develop new strategies for understanding of so far unknown. Transforming this idea to the handling with neural nets means, that first a net has to learn a basic concept. To prevent that the net used includes typical output ranges in its classification behavior due to the measured data for analysis, and therefore looks for measures with a high output, it is necessary to normalize the input data set by using an appropriate-preprocessing method. To handle changes in the global state of the case study it is favorable to use a pre-processing step that results in a gradient vector that is calculated by the difference between the data when no

pathological situation is present minus the present actual data.

By presenting the weaker states after the basic concept is settled it can be ensured that the net will be forced to change its classification structure slightly out of its former structure, without destroying the older structure. This means that the net will be sensitized. Especially when a back propagation network is used the learning rules force the net to sensitize its structure in such a way that only the case study specific state representing structure is modified, as the classification results have to be the same over the whole sensitization period. If weaker and weaker states will be presented successively the classification structure will change accordingly, until the similarity of the different evolutionary state representations will be so little, that the net can not be forced to change its structure anymore.

Figures 3.a and 3.b show the sensitized net working structure. Since the classification potential is changed locally, the net changes its classification behavior not in general by learning the evolutionary data sets, but shows the according adaptive behavior. This local change surely can lead to the unification of so far divided concepts, a fact which will open the door to a wide range of so far unknown or unnoticed intrinsic relations of the micro array data sets, representing the case study states.

Henceforth, sensitization in case of micro array data analysis can be introduced as intelligent pre-processing for clustering analysis that means normalization and filtering, that is necessary due to

Henceforth, sensitization in case of micro array data analysis can be introduced as intelligent pre-processing for clustering analysis that means normalization and filtering, that is necessary due to

- Systematic experimental errors,
- Uneven hybridization gel,
- Background variations,
- Wavelength dependency,
- Intensity dependency.
- Image processing algorithm-dependency
- Etc.

Hence, the importance of using intelligent pre-processing algorithms is really based on the hypothesis underlying micro array analysis that the measured intensities for each arrayed gene represent its relative expression level. However, before the levels can be appropriately compared, one generally performs a number of transformations on the data to eliminate questionable or low quality data, to adjust measured intensities to facilitate comparisons, and to select those genes that are significantly differentially expressed, which explains the need for a pre-processing methodology beyond.

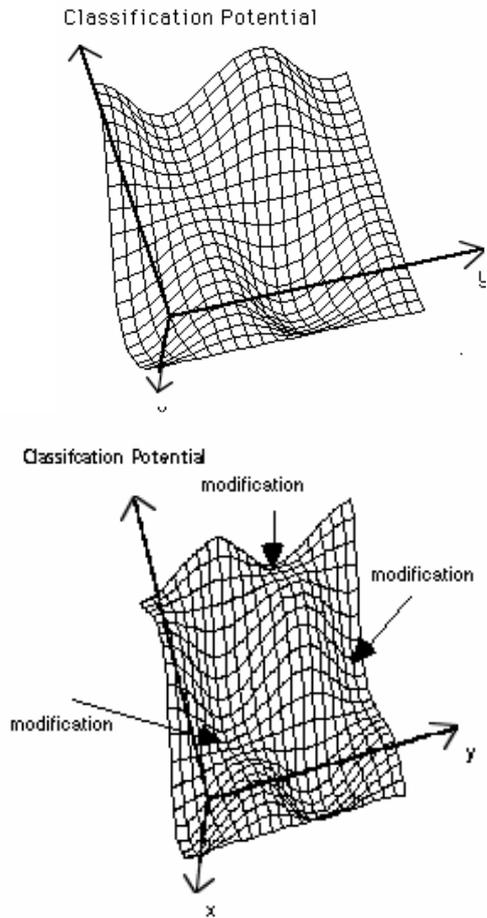


Figure 3: Classification-potential a) E before sensitization and b) E' after sensitization

The sensitized neuronal nets classifier in general is able to separate all trained states representing a powerful concept of weaker evolutionary states of different case study situations, to be trained

Figure 4 shows the time differences between an early warning of a common neural classifier and a sensitized neural network. Both nets have had the same warning criteria, setting an alarm when the probability for an pathological state is higher then 85%. It can be seen that a sensitized neural classifier is able to decrease the alarm time by a factor 5, as the net classifies the evolutionary state of the begin of an auricular fibrillation rather early.

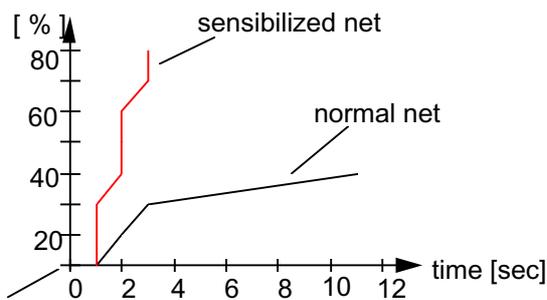


Figure 4: Difference between early warning of a sensitized and a normal neural net

In case of micro array data analysis sensitized nets can be developed for usage in

- Total Intensity normalization,
- Iterative linear regression normalization,
- Standard deviation regularization,
- Cross Slide Replicates T-test,
- Signal/Noise checking,
- Slice analysis,
- Etc.

As an measure example one can assume that two factors contributed to the gene X, the biological factors one is interested in, and experimental factors, one is not interested in. This requests for a possibility to extract the experimental factors which can be done initializing the pre-processing task of the statistical analysis. The statistical analysis behind can be

- Pre-processed local linear regression model,
- Pre-processed least squares,
- Etc.

To handle changes in the global state of cases it is favorable to use a pre-processing step that results in a gradient vector that is calculated by the difference between the data of a case study as requested when no pathological situation is present minus the present actual data.

Procedure of sensitized gene analysis in micro arrays

- Define a gene array window
- Sensitize window along  $\log(\text{IntensityProduct})$  axis
- Calculate  $\log\text{RatioMean}$  and  $\log\text{RatioSD}$  of gene data of the pre-processed samples
- Calculate Z-scores of each gene data point
- $Z\text{-score} = (\log\text{Ratio} - \log\text{RatioMean}) / \log\text{RatioSD}$
- Trim data with Z-scores beyond interested range

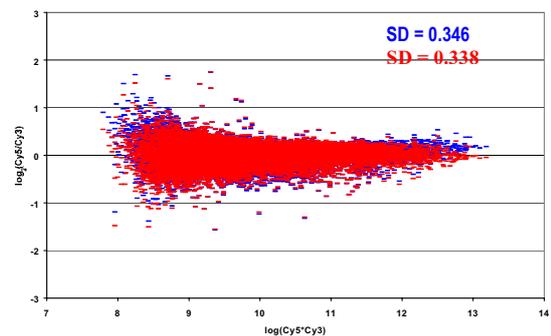


Figure 5 Sensitized Gene Expression Analysis

## EXPRESSION ANALYSIS AND WORKFLOW ANALYSIS

Workflow management systems provide the foundation for defining and executing business processes. A combined workflow process is embedded to overcome the

individual influences on data. At present there are no standardised methods available that can be used as a workflow-based quality assurance system.

- data generation,
- data handling,
- data processing,
- data interpretation, manually as well as automated.

in order to coordinate the information, based on a process control view, that includes the integration of data, documents, interpretation, execution of work and computer assisted work such as statistics. This can be achieved by developing a methodological concept for data management and data analysis, which has to be workflow based, embedded in business objects. While using the method of business objects, as shown for example in Fig. 6., the multiple steps of expression analysis or CGH will be adapted and standardised. A business object is defined as a representation of a thing active in the business domain, including at least its business name and definition, attributes, behavior, relationships, rules, policies and constraints.

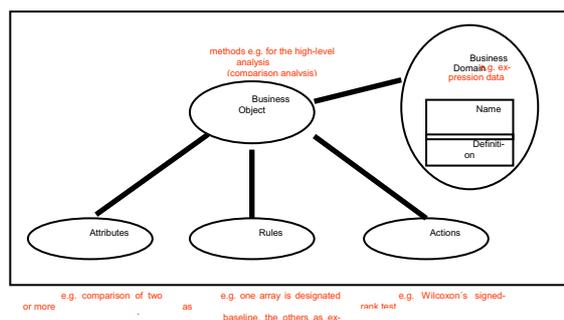


Figure 6: Business Object Concept for Gene Expression Analysis

The fundamental benefits inherent to workflow-based applications are

- Flexibility in changing the model of the underlying business process
- Integration capabilities for even disparate applications
- Reusability of activity implementations and process models
- Scalability of application development and execution

Henceforth the workflow forms the dataflow of the business objects which represent the name, definition, attributes, behavior, relationships, rules, policies and constraints of the gene expression algorithms. The fundamental benefits inherent to business objects in sensitized genes expression analysis are

- Flexibility in changing the model of the underlying business object
- Integration capabilities for even disparate applications
- Reusability of activity implementations and business objects
- Scalability of application development and execution due do the object oriented programming paradigm of business objects

## CONCLUSIONS

The potential of soft computing workflow analysis as well as business objects for micro array data analysis is huge. We only scratched the surface of the complex due to a brief view insight possible medical application domains. The potentially of soft computing and pre-processing contains an incredible number of solutions to the several problem depending domains.

## REFERENCES

- Kaufmann, A., Gupta, M. M. "Introduction to Fuzzy Arithmetic", Van Nostrand Reinhold, 2001
- Kosko, B. "Neural Networks and Fuzzy Systems", Prentice-Hall International, 2002
- Möller, D.P.F. "Soft Computing Methods", In: *Mathematical and Computational Modeling and Simulation*, Springer Verlag, Heidelberg, 2003
- Takagi, H., Sugeno M. "Fuzzy Identification of Systems and its Applications to Modelling and Control", *IEEE Trans. on Man, and Cybern.*, SMC-15(1), 1985, pp. 116-132
- Wang, L. X. "Adaptive Fuzzy Systems and Control", Prentice Hall International, 1994
- Zadeh, L. A. „Fuzzy Sets“, *Informat. Control, Vol. 8*, 1965, pp. 338-353

## AUTHORS



**DIETMAR P. F. MÖLLER** was born in Preetz, Germany. He enrolled at the Universities of Lübeck, Bremen Mainz and Bonn, where he studied Electrical Engineering and Human Medicine. He obtained his doctoral degree in 1980. From 1985 to 1991

Dr. Möller lead the [anaesthesia](#) division of Dräger AG in Lübeck, Germany. 1991 he has been elected as Full Professor for Computer Engineering (TI) at the Technical University of Clausthal. In 1998 he has been elected as Full Professor for Computer Engineering Systems (AB TIS) at the University of Hamburg where he holds the chair for Computer Engineering. Since 1998 he also is Adjunct Professor at the California State University Chico. His E-Mail address is [dmoeller@informatik.uni-hamburg.de](mailto:dmoeller@informatik.uni-hamburg.de) and his Web-page can be found at

<http://www.informatik.uni-hamburg.de/TIS/>

# INTERPOLATION FOR NON-REGULARLY LOCATED WELLS OF HYDROGEOLOGICAL MODELS

Aivars Spalvins, Janis Slangens and Inta Lace  
 Environment Modelling Centre  
 Riga Technical University  
 1/4 Meza Str., Riga, LV-1048, Latvia  
 E-mail: emc@egle.cs.rtu.lv

## KEYWORDS

Hydrogeological models, interpolation.

## ABSTRACT

Locations of production and monitoring wells do not coincide with nodes of hydrogeological model (HM) and these locations may be considered as non – regular points that should be attached to HM by interpolation. The paper is devoted to this type of interpolation that improves accuracy of HM. New results are reported that are especially important for regional HM where approximation grids are coarse.

## INTRODUCTION

The vector  $\varphi$  of the piezometric head is the numerical solution of a boundary field problem which is approximated in nodes of a HM grid of by the formula:

$$A\varphi = b, \quad A = A_{xy} + A_z - G, \quad b = \beta_\psi - G\psi \quad (1)$$

where the matrices  $A_{xy}$ ,  $A_z$ ,  $G$  represent, correspondingly, current transmittivity  $a_{xy}$  of aquifers (these links are arranged in  $xy$ -planes), the vertical ties  $a_z$  originated by aquitards (if the semi-3D scheme is used), the elements  $g_{xy}$ ,  $g_z$  connecting nodes of the grid with the boundary conditions  $\psi$ , the vector  $\beta$  accounts for boundary flows. They also include the vector  $\beta_w$  of groundwater discharge/recharge from wells.

The  $\varphi$  and  $\psi$ -distribution of (1) must reproduce values of the head measured at monitoring wells. As a rule, locations of production and monitoring wells do not coincide with nodes of the HM grid. These wells should be attached to the grid by interpolation. The roughest interpolation method moves them to the one nearest node. This method not only worsens the accuracy of  $\varphi$  (due to shifting positions of production wells), but also deteriorates the role of monitored head values as calibration targets. These effects may be considerable for regional HM where the plane approximation step  $h$  is large (500 m – 4000 m).

This paper is devoted to interpolation for wells of the HM grid. The reported results represent development of methods described in (Lace et al., 1995). Interpolations for non – regular points are conditionally named as the forth and back ones if they are used for forming HM

and for transferring obtained results to these points, respectively.

## INTERPOLATION FOR PRODUCTION WELLS

Forth interpolation for production wells is considered by using the scheme of Figure 1 for an elementary  $h \times h$  block of a uniform grid. Within the block, a single flow source 0 is sited. Its flow  $\beta_0$  should be interpolated among nodes  $n = 5, 6, 7, 8$ , as follows:

$$\beta_0 = \sum_{n=5}^8 \beta_0^n, \quad \beta_0^n = c_{0n} \cdot \beta_0, \quad \sum_{n=5}^8 c_{0n} = 1 \quad (2)$$

where the position of the source within the block depends on the local coordinates  $h_{0i}$ ,  $i = 1, 2, 3, 4$  ( $i$  - projections of 0 on edges),  $c_{0n}$  – the interpolation coefficients transferring  $\beta_0$  to the nodes  $n = 5, 6, 7, 8$ .

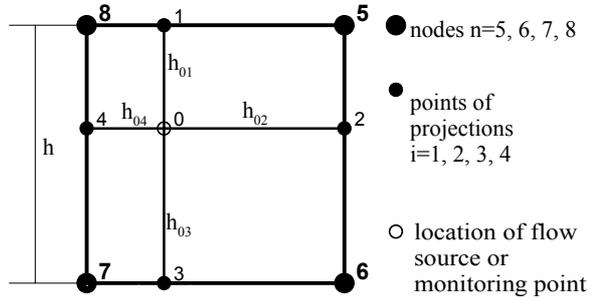


Figure 1: An elementary  $h \times h$  block with a flow source or a monitoring point

The following structure of  $c_{0n}$  results from two interpolation stages ( $0 \rightarrow i$ ;  $i \rightarrow n$ ):

$$\begin{aligned} c_{05} &= c_{01} \cdot c_{15} + c_{02} \cdot c_{25}, & c_{06} &= c_{02} \cdot c_{26} + c_{03} \cdot c_{36} \\ c_{07} &= c_{03} \cdot c_{37} + c_{04} \cdot c_{47}, & c_{08} &= c_{04} \cdot c_{48} + c_{01} \cdot c_{18}. \end{aligned} \quad (3)$$

The coefficients  $c_{0i}$  and  $c_{in}$  represent the stages  $0 \rightarrow i$ ,  $i \rightarrow n$ , respectively;  $c_{0i}$  are obtained, by applying the inverse distance method (IDM):

$$c_{0i} = \frac{a_{0i}}{a_{00}}, \quad a_{00} = \sum_{i=1}^4 a_{0i}, \quad a_{0i} = \frac{\sigma_i}{(h_{0i}/h + \varepsilon)^\nu}. \quad (4)$$

For the original version of (4),  $\nu=1.0$  was used. Due to reasons explained later,  $\nu=1.4$  provides better results. In (4), the constant  $\varepsilon \sim 10^{-5}$  averts the division by zero if  $h_{0i} = 0$ . Similar measures are needed for all interpolation

formulas to be considered further. To simplify their description, the necessary  $\varepsilon$ -protection is not displayed. The current transmittivity  $\sigma_i$  at the point  $i$  depends both on its position on the edge and on the transmittivities  $\sigma_n$  at nodes ending the edge:

$$\sigma_1 = \left( \frac{h_{02}}{h\sigma_8} + \frac{h_{04}}{h\sigma_5} \right)^{-1}, \quad \sigma_2 = \left( \frac{h_{03}}{h\sigma_5} + \frac{h_{01}}{h\sigma_6} \right)^{-1},$$

$$\sigma_3 = \left( \frac{h_{04}}{h\sigma_6} + \frac{h_{02}}{h\sigma_7} \right)^{-1}, \quad \sigma_4 = \left( \frac{h_{01}}{h\sigma_7} + \frac{h_{03}}{h\sigma_8} \right)^{-1}. \quad (5)$$

If  $h_{0i} \rightarrow 0.5 h$ ,  $\sigma_i \rightarrow a_{xy}$  of  $A_{xy}$ . The first version of (4) applied  $\sigma_i = a_{xy}$  (Lace et al., 1995). The coefficient  $c_{in}$  depends only on the position of the point  $i$  on the edge:

$$c_{15} = c_{36} = \frac{h_{04}}{h}, \quad c_{18} = c_{37} = \frac{h_{02}}{h},$$

$$c_{25} = c_{48} = \frac{h_{03}}{h}, \quad c_{26} = c_{47} = \frac{h_{01}}{h}. \quad (6)$$

By introducing normalized distances  $\rho_{0i} = h_{0i}/h$  and the local normalized coordinates  $\xi$  and  $\eta$ , with the node  $n=7$  as the origin:

$$\xi = \frac{h_{04}}{h} = \rho_{04}, \quad 1 - \xi = \frac{h_{02}}{h} = \rho_{02},$$

$$\eta = \frac{h_{03}}{h} = \rho_{03}, \quad 1 - \eta = \frac{h_{01}}{h} = \rho_{01} \quad (7)$$

and accounting for (6), the expression (3) takes the form:

$$c_{05} = c_{01} \cdot \xi + c_{02} \cdot \eta, \quad c_{07} = c_{03} \cdot (1 - \xi) + c_{04} \cdot (1 - \eta),$$

$$c_{06} = c_{03} \cdot \xi + c_{02} \cdot (1 - \eta), \quad c_{08} = c_{01} \cdot (1 - \xi) + c_{04} \cdot \eta. \quad (8)$$

If in (4)  $\nu=1.0$  and  $\sigma = \text{const}$ , the system (8) becomes much simpler:

$$c_{05} = \xi \cdot \eta, \quad c_{06} = \xi \cdot (1 - \eta),$$

$$c_{07} = (1 - \xi) \cdot (1 - \eta), \quad c_{08} = (1 - \xi) \cdot \eta \quad (9)$$

The system of (9) represents the set of rectangular hyperbolas projected on the normalized block  $1 \times 1$ . As an example, contours of  $c_{07} = \text{const}$  of (9) are shown in Figure 2. These contours have the following features:

- forth interpolation of  $\beta_0$  is linear on any line parallel to the edges of the block;
- $c_{07} = 0$  if  $\xi = \eta = 1$  (edges 8 - 5 and 5 - 6); therefore, the influence region for the node  $n = 7$  represents the  $2 \times 2$  area containing four elementary blocks surrounding the node;
- if  $\xi$  or  $\eta = 0$  (edges 6-7 and 7-8) then  $\beta_0$  gets distributed between two nodes ending the

edges.

However, the contours of Figure 2 are not circular with respect to the node  $n = 7$ , at its vicinity This drawback can be corrected if  $\nu = 1.4$  is used for  $c_{0i}$  of (4). The improved contours are shown in Figure 3:

- their shape is still close to rectangular hyperbolas if  $k_{07} < 0.35$ ;
- interpolation is linear on edges of the elementary block.

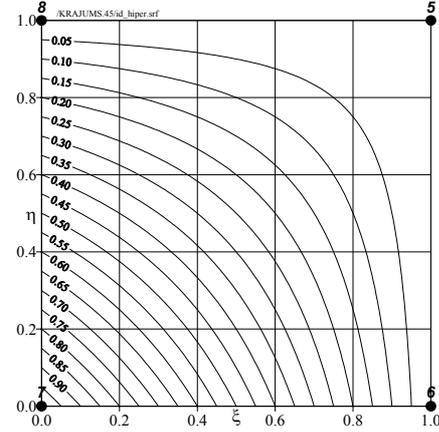


Figure 2. Contours of  $c_{07}$  as rectangular hyperbolas obtained by (9),  $\sigma = \text{const}$

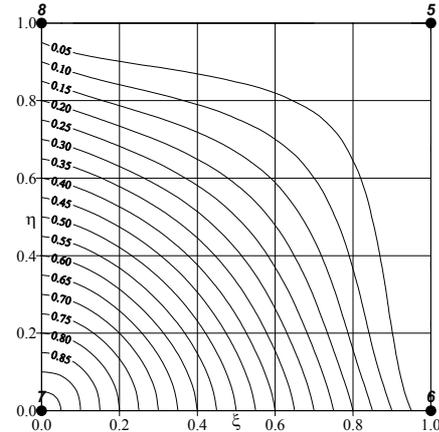


Figure 3. Improved contours of  $c_{07}$  obtained by (8) if  $\nu=1.4$ ,  $\sigma = \text{const}$

Not any forth interpolation method possesses useful features of (8). For example, the classic IDM gives:

$$c_{07} = \left( \rho_{07} \cdot \sum_{n=5}^8 \frac{1}{\rho_{0n}} \right)^{-1} \quad (10)$$

where  $\rho_{0n} = r_{0n} / h$ . In Figure 4, the contours  $c_{07}$  of (10) are shown. Their drawbacks are obvious:

- interpolation is nonlinear in any direction;
- no borderline  $c_{07} = 0$  exists;  $c_{07} = 0$  only at nodes  $n = 5, 6, 8$ ; it is not possible to set justly the area of influence of the node  $n = 7$ ;
- if the source  $\beta_0$  is located on an edge of the elementary block then not only the endpoints

of the edge, but at least four neighboring nodes should be accounted for.

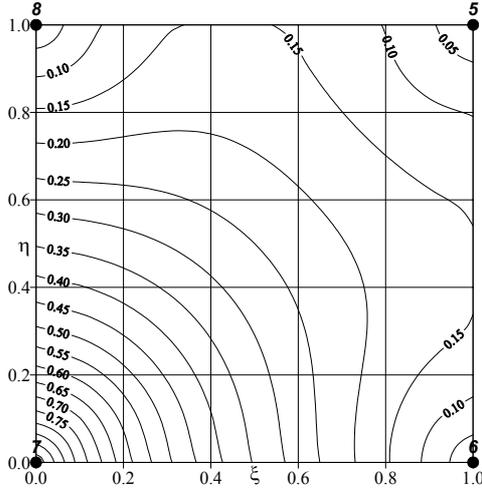


Figure 4. Contours of  $c_{07}$  obtained by the inverse distance method (10),  $\sigma = \text{const}$

For a node  $n$ , the summary flow  $\beta_n$  resulting from interpolation of irregular  $\beta_j$ , which are located within the  $2h \times 2h$  area of influence, is given by the formula:

$$\beta_n = \sum_{j=1}^J \beta_j^n, \quad (11)$$

where  $\beta_j^n$  are the partial flows  $\beta_j$  of (2);  $J$  is the number of sources accounted for.

Forth interpolation of  $\beta_w$  improves accuracy of HM. Unfortunately, this advantage can be exploited only then if back interpolation is available for irregularly located production and monitoring wells.

## RESORATION OF HEADS FOR PRODUCTION WELLS

In comparison with forth interpolation of  $\beta_w$ , back interpolation for non-regular points is more complex. The value of  $\varphi_0$  for the source of  $\beta_0$  interpolated into the neighboring nodes must be restored. The following assumption is used:

$$\varphi_0 = \sum_{n=5}^8 \varphi_n \cdot c_{0n} + \tau_0 \cdot \beta_0 = \varphi_0^n + s_0 \quad (12)$$

where  $\varphi_n$  are the computed heads at four nodes of Figure 1;  $c_{0n}$  are the improved coefficients of (8);  $\tau_0$  is the local hydraulic resistance for the source;  $\varphi_0^n$ ,  $s_0$  are the head and local depression, caused by the grid solution and the source, respectively. The value of  $\tau_0$  should be predicted for any location of  $\beta_0$  within the elementary block. To simplify this task, it is assumed that  $\sigma_n = 1$ , temporarily.

For any node of the grid,  $\tau_0 = 0$ . The maximum of  $\tau_0$  is expected at the centre of the block where  $c_{05} = c_{06} = c_{07}$

$= c_{08} = 0.25$ . The other characteristic locus is the middle of an edge where  $\beta_0$  is distributed in equal parts between the two nodes ending the edge. These two special values of  $\tau_0$  were obtained experimentally, as described below.

The elementary block was conditionally placed at the central part of a homogenous grid ( $\sigma = 1.0$ ) containing  $100 \times 100$  nodes; on the borderline of the grid, the condition  $\psi = 0$  was specified. A single movable unity source of  $\beta_0 = 1.0$  was applied as the flow condition to be positioned and interpolated within the elementary block. Then the grid solution of (1) can be interpreted as the resistances  $\tau_{int}$  at nodes with respect to the nullified borderline. The maximal possible value  $\tau_m = 0.8874$  was obtained when the source was located exactly at the node. This value was practically constant for all nodes of the grid, but the ones located nearby the borderline. If the source was sited at the centre of the elementary block then the minimal value  $\tau_{int} = 0.6842$  appeared at four nodes where the interpolated partial flows  $\beta_0^n = 0.25$  were applied. The value  $\tau_{int} = 0.7634$  was obtained for two nodes if the source was in the middle of an edge. The local resistance  $\tau_0$  to be found is  $\tau_0 = \tau_m - \tau_{int}$ . Results of the experiment are summarized in Table 1.

Table 1. Computed resistances for various positions of the unity source

Nr	Position of source	Resistance at node $\tau_{int}$	Local resistance $\tau_0$	Equivalent radius $r_s$ of source
1	node	0.8874	0	0.1972 $h$
2	edge	0.7634	0.1240	0.4299 $h$
3	centre	0.6842	0.2032	0.7071 $h$

The following analytic formula for the resistance  $\tau$  between two coaxial cylinders ( $R$  and  $r$  are, correspondingly, radii of the outer and inner cylinders)

$$\tau = \frac{1}{2\pi\sigma} \ln \frac{R}{r} \quad (13)$$

is applied to find the equivalent radius  $r_s$  of the interpolated source. If  $\tau = \tau_{int}$ ,  $\sigma = 1$  then  $r_s = R / \exp(2\pi\tau_{int})$  where  $R = 52.059 h$  approximated the borderline of the grid area containing  $100 \times 100$  nodes.

The values of  $\tau_0$  from Table 1 are exactly repeated by (13) if  $R = r_s$ ,  $r = 0.1972 h$ . There the ratio  $R/r$  does not include  $h$ . Therefore,  $\tau_0$  depends only on the position of the source within the block  $h \times h$ .

It follows from Table 1 that any source located exactly in a node has the equivalent radius  $r_s = 0.1972 h$ . It may be assumed that a non – regularly located source also has  $r_s = 0.1972 h$ . As a rule,  $r_s > r_w$  where  $r_w$  is the real radius of the well. Due to this reason, the summary resistance  $\tau_{0w}$  of the source is, as follows:

$$\tau_{0w} = \tau_0 + \tau_w, \quad \tau_w = \frac{1}{2\pi\sigma_0} \ln\left(\frac{0.1972}{\rho_w}\right),$$

$$\rho_w = \frac{r_w}{h}, \quad \rho_w \leq 0.1972,$$

$$\sigma_0 = \frac{\sum_{i=1}^4 \frac{\sigma_i}{\rho_{0i}}}{\sum_{i=1}^4 \frac{1}{\rho_{0i}}} \quad (14)$$

where  $\tau_w$  is obtained by (13) if  $R = 0.1972$ ;  $r = \rho_w$ ; the current  $\sigma_0$  is IDM interpolation on  $\sigma_i$  of (5).

The surface  $\tau_0$  is the main element enabling to restore heads at production wells by using (12). The initial version of the empiric formula for computing of  $\tau_0$ , within the normalized block, was as follows:

$$\tau_0 = \left(0.3444 + 0.4960a_{0i}a_i^{-1}\right) \Big/ \sum_{i=1}^4 a_{0i}, \quad a_{0i} = \frac{\sigma_i}{\rho_{0i}} \quad (15)$$

where  $a_i$  were given by the expressions:

$$a_1 = \frac{\sigma_1}{\xi(1-\xi)}, \quad a_2 = \frac{\sigma_2}{\eta(1-\eta)},$$

$$a_3 = \frac{\sigma_3}{\xi(1-\xi)}, \quad a_4 = \frac{\sigma_4}{\eta(1-\eta)}. \quad (16)$$

The formula (15) confirms the experimental values from Table 1. In Figure 5, the contours  $\tau_0$  of (15) are shown on the quarter of the normalized elementary grid block if  $\sigma=1$ . Contours of (15) have two disadvantages:

- in the vicinity of nodes, the contours are not circular towards the nodes as their origins;
- on edges, as borders between neighboring blocks, the values of  $\tau_0$  may not coincide when  $\sigma_n \neq \text{const}$ .

These drawbacks are eliminated in the following improved formula:

$$\tau_0 = \left(0.3444 + 0.5697a_{0i}c_{0i}a_i^{-1.1}\right) \Big/ \sum_{i=1}^4 a_{0i}c_{0i},$$

$$c_{0i} = \sigma_i \cdot \rho_{0i}^{-1.05} \Big/ \sum_{i=1}^4 \sigma_i \cdot \rho_{0i}^{-1.05}. \quad (17)$$

Due to introduction of  $c_{0i}$ , values of  $\tau_0$  for neighboring blocks coincide on edges bordering them. The elements  $a_{0i}$ ,  $a_i$  are common for (15) and (17). In Figure 6, the contours  $\tau_0$  of (17) are shown.

However, the formula (12) cannot give full value of  $s_0$  if other nearby located flow sources are present. To account for this situation, the surface  $s_{0j}$  of the local depression cone caused by  $\beta_j$  is necessary. This task is solved in the next section devoted to computing heads at monitoring wells.

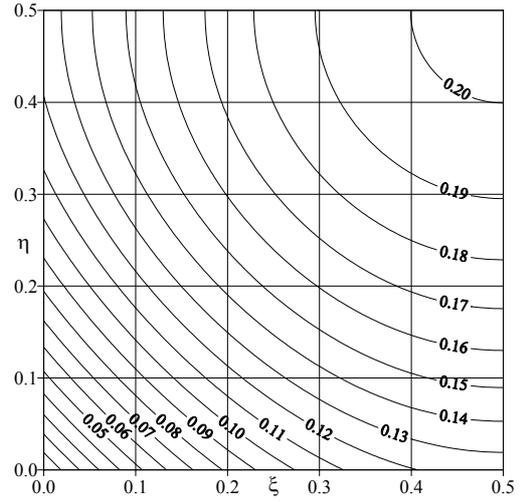


Figure 5. Contours of  $\tau_0$  if (15) is used,  $\sigma=1.0$

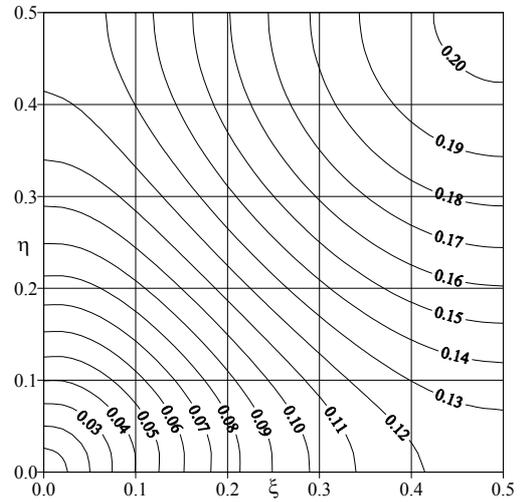


Figure 6. Improved contours of  $\tau_0$  if (17) is used,  $\sigma=1.0$

## COMPUTING OF HEADS FOR MONITORING WELLS

The task of computing  $\varphi_0$  for a monitoring well of Figure 1 is more universal than the one devoted to restoring of the local maximum  $s_0$  at the production well. It is assumed that the expression should be used:

$$\varphi_0 = \sum_{n=5}^8 \varphi_n \cdot c_{0n} + s_{0j}, \quad s_{0j} = \tau_{0j} \cdot \beta_j, \quad (18)$$

where  $\tau_{0j}$  is the transfer resistance of the source  $\beta_j$  towards the monitoring point 0. If the distance  $\rho_{0j} \rightarrow 0$  then  $\tau_{0j} \rightarrow \tau_0$  and (18)  $\rightarrow$  (12).

The value of  $\tau_{0j}$  must be zero at any node. The surface of  $\tau_{0j}$  must be flat on the level  $\tau_j$  (given by (17)) within a circle with the centre  $\beta_j$  and the radius  $\rho = 0.1972$ . Such a surface may be approximated by the modified IDM, as follows:

$$\tau_{0j} = \frac{\tau_j \rho_{0j}^{-3}}{\rho_{0j}^{-3} + \sum_{p=1}^P \rho_{0p}^{-1.5}}, \quad \rho_{0j} \leq 2.0 \quad (19)$$

where  $\rho_{0j}$  and  $\rho_{0p}$  are the normalized distances between the monitoring point 0 and the source  $\beta_j$  and the nearby nodes  $p = 1, 2, \dots, P$ , correspondingly; these distances should not exceed 2.0. The expression (19) is empiric. It was calibrated for the elementary block by accounting for the two characteristic source positions when  $\sigma_n = 1.0$ : 1) the center; 2) the middle of an edge. The results for the centre are shown in Figure 7 and Figure 8. In Figure 7, the contours of  $\tau_{0j}$  are exposed. The top  $\rho_{0j} \leq 0.1972$  of the surface  $\tau_{0j}$  is not ideally flat on  $\tau_{0j} = 0.2032$  and this fact causes errors. In (19), the powers -3.0 and -1.5 are chosen to minimize the error  $\Delta_{0j}$ , along the diagonal of the normalized block  $1 \times 1$  (Figure 8):

$$\begin{aligned} \Delta_{0j} &= \frac{1}{2\pi} \ln \frac{1}{\sqrt{2} \rho_{0j}} - \tau_{0j}, \text{ if } \frac{1}{\sqrt{2}} \geq \rho_{0j} \geq 0.1972, \\ \Delta_{0j} &= 0.2032 - \tau_{0j}, \text{ if } 0.1972 > \rho_{0j} > 0. \end{aligned} \quad (20)$$

where the analytic standard of (20) is represented by (13) if  $1/\sqrt{2} \geq \rho_{0j} \geq 0.1972$ ,  $R = 1/\sqrt{2}$ .

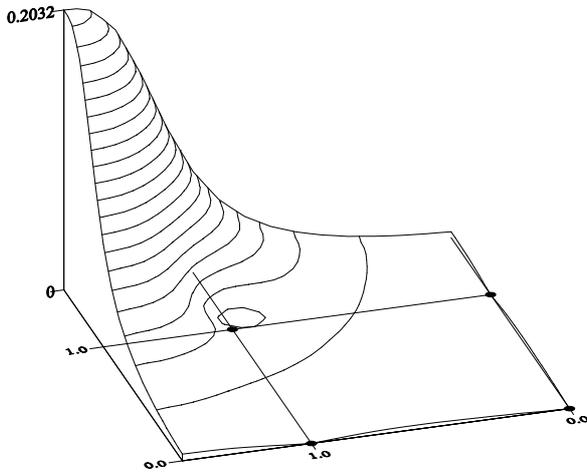


Figure 7. The source at the center of the normalized block. Contours of  $\tau_{0j}$  on the quarter of the region  $3 \times 3$ ,  $\sigma = 1.0$ ; the isoline step is 0.01

It follows from Figure 8 that the graph of  $\Delta_{0j}$  has two maximal values 0.014 and -0.012 when  $\rho_{0j} = 0.2$  and 0.35, respectively. Therefore, the relative error  $100 \Delta_{0j} / 0.2032$  given by (20) does not exceed 7%. If  $\rho_{0j} < 0.1972$ , the following analytic correction is necessary:

$$(\tau_{0j})_w = \tau_{0j} + (\tau_{0j})_{0.2h},$$

$$(\tau_{0j})_{0.2h} = \frac{1}{2\pi\sigma_j} \ln \left( \frac{0.1972}{\rho_{0j}} \right) \quad (21)$$

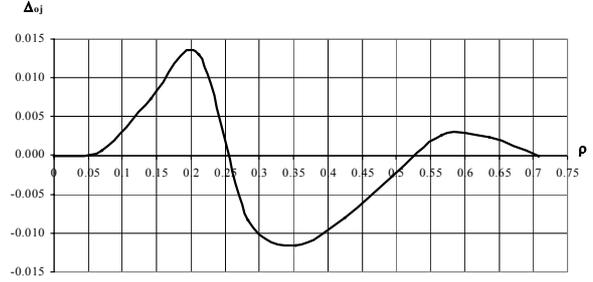


Figure 8. Source at the center of the block  $1 \times 1$ . The graph  $\Delta_{0j}$  along the diagonal if  $1/\sqrt{2} \geq \rho \geq 0$ ,  $\sigma = 1.0$

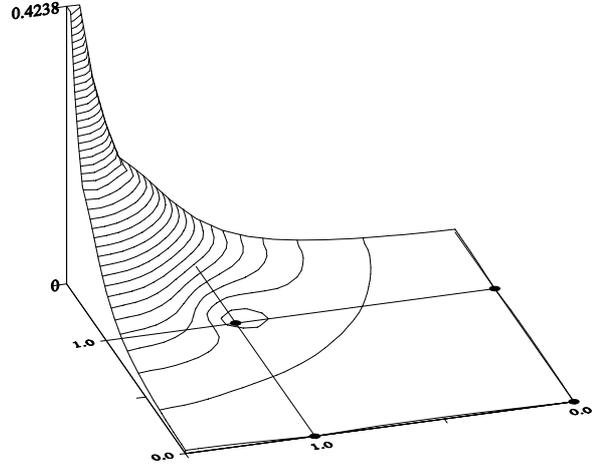


Figure 9. The source at the center of the normalized block. Contours of  $\tau_{0j} + (\tau_{0j})_{0.2h}$  on the quarter of the region  $3 \times 3$ ,  $\sigma = 1.0$ ; the isoline step is 0.01

There  $(\tau_{0j})_{0.2h}$  represents the analytic complement provided by (13) if  $R = 0.1972h$ . In Figure 9, contours of  $(\tau_{0j})_{0.2h}$  are shown if the minimal  $\rho_{0j} = 0.25 \times 0.1972$ . Due to the error  $\Delta_{0j}$  caused by the slantwise top of  $\tau_{0j}$ , the junction of the surfaces  $\tau_{0j}$  and  $(\tau_{0j})_{0.2h}$  is not smooth. In Figure 10, Figure 11 the results are presented when  $\beta_j$  is at the middle of the edge. Contours of  $\tau_{0j}$  are shown in Figure 10. For the source  $\beta_j$ ,  $\tau_{0j} = \tau_j = 0.124$ . The error  $\Delta_{0j}$  of  $\tau_{0j}$  is evaluated on the edge where  $\beta_j$  is positioned:

$$\begin{aligned} \Delta_{0j} &= \frac{0.837}{2\pi} \cdot \ln \frac{1}{2\rho_{0j}} - \tau_{0j}, \text{ if } 0.5 \geq \rho_{0j} \geq 0.1972, \\ \Delta_{0j} &= 0.124 - \tau_{0j}, \text{ if } 0.1972 > \rho_{0j} > 0. \end{aligned} \quad (22)$$

It follows from the graph of  $\Delta_{0j}$  provided by (22) that its maximum  $\Delta_{0j} = 0.0145$  when  $\rho_{0j} \sim 0.2$  (Figure 11). It is caused by the slantwise top of the surface  $\tau_{0j}$  if  $\rho_{0j} < 0.2$ . This maximum is practically the same as for the graph

of Figure 8.

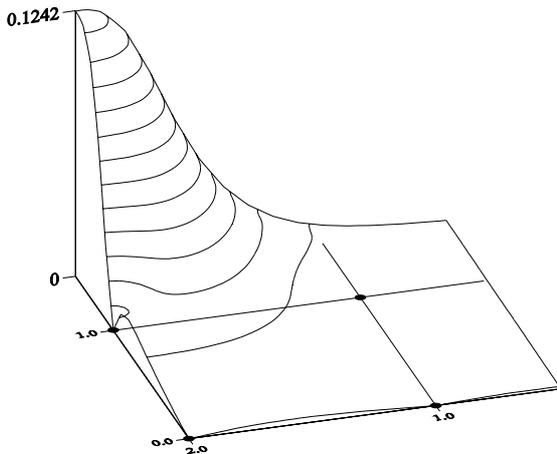


Figure 10. The source at the middle of the edge of the block  $1 \times 1$ . Contours of  $\tau_{0j}$  on the quarter of the region  $3 \times 3$ ,  $\sigma = 1.0$ ; the isoline step is 0.01

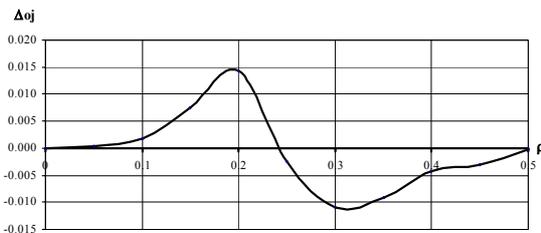


Figure 11. The source at the middle of the edge of the block  $1 \times 1$ . The graph of  $\Delta_{0j}$  along the edge if  $0.5 \geq p \geq 0$ ,  $\sigma=1.0$

If the surface  $\tau_{0j}$  of (19) is used as a tool for computing of  $s_{0j}$  at the monitoring well 0 then it is possible to account for the influence of various sources  $\beta_j, j = 1, 2, \dots, J$ . They are located within a circle of the radius  $\rho_{0max} = 2.0$  with the centre 0 where superposition of  $s_{0j}$  is applied. The final formula is generalization of (18):

$$\varphi_0 = \sum_{n=5}^8 \varphi_n \cdot c_{0n} + \sum_{j=1}^J s_{0j}, \quad s_{0j} = \tau_{0j} \cdot \beta_j. \quad (23)$$

It is supposed that the nearest source is  $\beta_1, j = 1, \rho_{01} = \min$ . If  $\rho_{01} < 0.1972$  then the complement of (21) should be used for obtaining of  $s_{01} = (\tau_{01})_w \beta_1$ . When  $\rho_{01} \rightarrow \rho_{w1}$  then  $s_{01} \rightarrow s_1 = \tau_{1w} \beta_1$  of (14), as the maximum of the local depression caused by  $\beta_1$ .

If compared with the original version of back interpolation, the expression (23) is more universal. Formulation of its main components  $\tau_j$  and  $\tau_{0j}$  have been improved considerably.

## CONCLUSIONS

1. Methods for forth and back interpolation have been

developed for non-regularly located production and monitoring wells.

2. The methods improve accuracy of hydrological models, especially, if grid plane steps are large.
3. The improved interpolation methods have been implemented in software developed by the Environment Modelling centre of the Riga Technical University.

## REFERENCES

- Lace, I., Spalvins, A. & J. Slangens. 1995. "Algorithms for Accounting Groundwater Discharge in the Regional Hydrogeological Model and Interpolation of Simulation Results at Observation Wells". *Proc. of International Seminar on "Environment Modelling". Boundary Field Problems and Computers*. Riga-Copenhagen, Vol. 36, part 1, 201-216

## AUTHOR BIOGRAPHIES

**AIVARS SPALVINS** was born in the district of Bauska, Latvia. In 1959 he entered Riga Politechnical Institute (now Riga Technical University), where he studied computer engineering. In 1968 he obtained his Dr. sc. ing. degree. A. Spalvins has been with the University until now. He is leading the Environment Modelling center (EMC) involved in hydrogeological modelling. E-mail address: emc@egle.cs.rtu.lv. EMC Web-page: [http://www.rtu.lv/www\\_emc](http://www.rtu.lv/www_emc).

**JANIS SLANGENS** was born in the district of Tukums, Latvia. In 1964 he entered Riga Politechnical Institute (now Riga Technical University), where he studied computer engineering. In 1985 he obtained his Dr. sc. ing. degree. J. Slangens has been with the University until now. He is working in EMC. E-mail address: emc@egle.cs.rtu.lv. EMC Web-page: [http://www.rtu.lv/www\\_emc](http://www.rtu.lv/www_emc).

**INTA LACE** was born in the district of Jekabpils, Latvia. In 1966 she entered Riga Politechnical Institute (now Riga Technical University), where she studied computer engineering. In 1971 she obtained her Mg. sc. ing. degree. I. Lace has been with the University until now. She is working in EMC. E-mail address: emc@egle.cs.rtu.lv. EMC Web-page: [http://www.rtu.lv/www\\_emc](http://www.rtu.lv/www_emc).

# PORTFOLIO MODELLING USING THE THEORY OF COPULA IN LATVIAN AND AMERICAN EQUITY MARKET

Vladimirs Jansons  
Konstantins Kozlovskis  
Natalja Lace  
Faculty of Engineering Economics  
Riga Technical University  
Kalku iela 1, Riga, LV1658, Latvia

## KEYWORDS

Theory of copula, Markowitz's approach, portfolio modelling.

## ABSTRACT

Portfolio building is the most important process in investment management. In the world there are many methods which can help both to estimate the investment characteristics of a financial instrument and to build a portfolio. The overwhelming majority of them are based on the assumption that analyzed data are normally distributed. The theory of copula allows to study non-linear dependences between selected assets and to build unified distribution function based on the distribution functions of each asset. In the theory of copula the set of financial instruments can be considered as one asset. The authors of this paper analyzed the effectiveness of copula implementation in the financial markets represented by the American market and the Latvian equity market. The authors compared the cumulative returns of two portfolios formed by traditional Markowitz's approach and simulating copula. Different data time formats were used in this research.

## INTRODUCTION

The problem of portfolio building is divided into two sub-problems:

- 1) the estimation of the investment characteristics of an asset;
- 2) forming of portfolio.

Many specialists work out and offer their own methods like digital portfolio theory which consider the asset movements as an electric signal. Other methods are the modifications of traditional Markowitz approach or Value-at-Risk methodology. Anyway, the overwhelming majority of these methods are based on the assumption that analyzed data are normally distributed. In fact, this assumption significantly simplifies calculations and gives acceptable results. In addition, traditional Markowitz's approach takes into account the correlation and linear dependence between the assets.

Modern data analysis techniques and the latest achievements of information technologies allow us to

analyze the financial markets from another point of view. Now there are a lot of scientific papers which are the evident of new features of the financial markets. They show that the usage of normal distribution function in data analysis is not correct because the kurtosis and skewness of data distribution is too far from normal. Also the assets can have non-linear dependences between each other or their investment characteristics.

The theory of copula is one of the interesting methods which can facilitate to solve a part of the problems mentioned above. This theory allows us to analyze non-linear dependences between assets and to merge different distribution functions in one unified.

## MARKOWITZ'S APPROACH

In 1952 year Harry Markowitz published his famous work named "Portfolio Selection". Markowitz's paper is the first mathematical formalization of the idea of diversification of investments: the financial version of "the whole is greater than the sum of its parts". Through diversification, risk can be reduced (but not generally eliminated) without changing expected portfolio return. Markowitz postulates that an investor should maximize expected portfolio return ( $\mu_p$ ) while minimizing portfolio variance of return ( $\sigma_p^2$ ). Later it became a part of modern portfolio theory.

According to traditional approach worked out by Markowitz portfolio return can be calculated as

$$\mu_p = \sum_j w_j \mu_j \quad (1)$$

where the  $\mu_j$  is expected security's return.

The variance of portfolio return can be calculated as

$$\sigma_p^2 = \sum_j w_j^2 \sigma_j^2 + \sum_j \sum_{k \neq j} w_j w_k \rho_{jk} \sigma_j \sigma_k \quad (2)$$

where the  $w_j$  are the portfolio proportions and  $\rho_{jk}$  is the pairwise correlation of the returns of securities  $j$  and  $k$ .

## THEORY OF COPULA

An n-dimensional copula is basically a multivariate cumulative distribution function with uniform distributed margins in  $[0, 1]$ .

### Sklar's Theorem

Let  $H$  denote a n-dimensional distribution function with margins  $F_1, \dots, F_n$ . Then there exists a n-copula  $C$  such that for all real  $(x_1, \dots, x_n)$

$$H(x_1, \dots, x_n) = C(F(x_1), \dots, F(x_n)) \quad (3)$$

If all the margins are continuous, then the copula is unique. A copula is thus a function that, when applied to univariate marginals, results in a proper multivariate probability distribution function: since this probability distribution function embodies all the information about the random vector, it contains all the information about the dependence structure of its components. Using copulas in this way splits the distribution of a random vector into individual components (marginals) with a dependence structure (the copula) among them without losing any information. It is important to highlight that this theorem does not require  $F_1$  and  $F_n$  to be identical or even to belong to the same distribution family.

### Archimedean Copulas

In this paper the authors utilized the Archimedean copulas. The Archimedean copulas provide analytical tractability and a large spectrum of different dependence measure. These copulas can be used in a wide range of applications for the following reasons:

1. The many parametric families of copulas belonging to this class.
2. The great variety of different dependence structures.
3. The ease with which they can be constructed, and the nice properties possessed by the members of this class.

An Archimedean copula can be defined as follows: let us consider a function  $\varphi: [0; 1] \rightarrow [0; 1]$  which is continuous, strictly decreasing  $\varphi'(u) < 0$ , convex  $\varphi''(u) > 0$ , and for which  $\varphi(0) = \infty$  and  $\varphi(1) = 0$ .

We then define the pseudo inverse of  $\varphi^{[-1]}: [0; \infty] \rightarrow [0; 1]$  such that:

$$\varphi^{[-1]}(t) = \begin{cases} \varphi^{-1}(t) & 0 \leq t \leq \varphi(0) \\ 0 & \varphi(0) \leq t \leq \infty \end{cases} \quad (4)$$

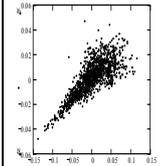
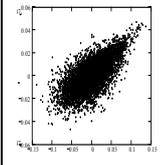
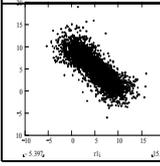
As  $\varphi$  is convex, the function  $C: [0; 1]^2 \rightarrow [0; 1]$  defined as

$$C(u_1, u_2) = \varphi^{-1}[\varphi(u_1) + \varphi(u_2)] \quad (5)$$

is an Archimedean copula and  $\varphi$  is called the "generator" of the copula.

In this paper the authors used three Archimedean copulas (see Table 1).

Table 1: Used bivariate copulas and generators

	<p><b>Clayton copula</b></p> $C(u_1, u_2) = (u_1^{-\theta} + u_2^{-\theta} - 1)^{-1/\theta}$ $\varphi(t) = \frac{t^{-\theta} - 1}{\theta}$ <p>with <math>\theta \in (0, \infty)</math></p>
	<p><b>Gumbel copula</b></p> $C(u_1, u_2) = \exp\left\{-\left[(-\ln u_1)^\theta + (-\ln u_2)^\theta\right]^{1/\theta}\right\}$ $\varphi(t) = (-\ln t)^\theta$ <p>with <math>\theta \geq 1</math></p>
	<p><b>Frank copula</b></p> $C(u, v) = -\theta^{-1} \ln\left(1 + \frac{g(u) \cdot g(v)}{g(1)}\right)$ $\varphi_\theta(t) = -\ln \frac{g(t)}{g(1)}$

As mentioned above the traditional portfolio theory based on multivariate normal distribution assumes that investors can benefit from diversification by investing in assets with lower correlations.

### Kendall's Rank Correlation

The pairwise correlation coefficient takes into account the linear dependence between the assets and also it is limited by some specific restrictions. In the theory of copula there is another correlation coefficient called Kendall's Tau. Kendall's rank correlation for the sample can be calculated by

$$\hat{\tau} = \binom{n}{2}^{-1} \sum_{i < j} \text{sign}[(X_i - X_j)(Y_i - Y_j)] \quad (6)$$

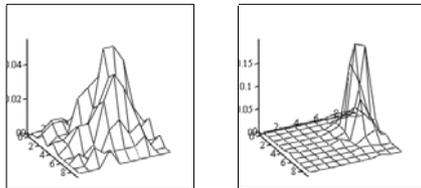
$\tau(X, Y)$  can be considered as measure of the degree of monotonic dependence between  $X$  and  $Y$ .

Table 2: Kendall's Tau for Some Copulas

	<b>Kendall's Tau</b>
Clayton copula	$\tau(\theta) = \frac{\theta}{2 + \theta}$
Gumbel copula	$\tau(\theta) = 1 - \frac{1}{\theta}$
Frank copula	$\tau(\theta) = 1 - \frac{4}{\theta} + \frac{4}{\theta^2} \int_0^\theta \frac{t}{e^t - 1} dt$

## Optimal Copula

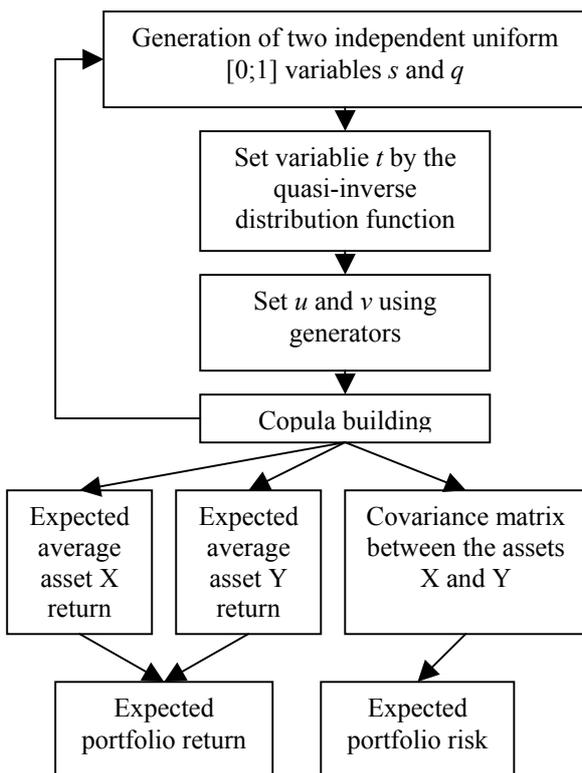
The main problem is how to choose the best copula from the set of estimated ones. The authors used the following method. Two density distributions are built: theoretical and empirical. If maximal distance between two densities is minimal for different copulas then this copula is the best one.



Figures 1: Two Distribution Densities: Theoretical (left) and Empirical (right)

## THE PROCESS OF PORTFOLIO BUILDING BASED ON COPULA SIMULATING

Copula implementation in portfolio building is shown in the Figures 2.



Figures 2: Theory of Copula in Portfolio Building

According to this modelling process the assets returns and covariance matrix are calculated separately. In the case of traditional Markowitz's approach the asset return is calculated as simple average (see formula 1) and covariance matrix as product of correlation and standard deviations (see formula 2). In the case of modeled portfolio the asset return is calculated as

simple average on  $x$  simulations based on the corresponding copula and covariance matrix is calculated as average covariance based on  $x$  simulations of non-linear dependences between two assets.

## EMPERICAL RESULTS

The authors analyzed three stocks traded in the American stock market (COKE – Coca-Cola Bottling Co Consolidated, F – Ford Motor Co, Microsoft CP) and three more active stocks traded in the Latvian equity market (GZE1R – Latvian Gas, VNF1R – Ventspils Oil, GRD1R - Grindex). D1, W1 and M1 data represent the average daily, weekly and monthly prices, respectively over the corresponding time period. The general information about analyzed periods and test periods is shown in the Table 3.

Table 3: Periods for Data Analyzing

Data	Analyzed period	Test period
D1	January 3, 2000 -	January 3, 2004 -
W1	December 31, 2003	December 31, 2004
M1	January 3, 1990 -	January 3, 2003 -
	December 31	December 31, 2004

Thus, the data of analyzed period are used for portfolio building. Then the cumulative returns of two portfolios are compared within the test period.

Using the formulas (1) and (2) it is very simple to calculate portfolio risk and return against its structure and then to build a portfolio set. In the same way another portfolio set, based on modelling, is built. In the Table 4 we can see the portfolio sets sorted by data time formats and financial markets.

Table 4: Behaviour of Two Portfolios (White – Modelled Portfolio, Black – Markowitz's Portfolio)

	American market	Latvian market
D1		
W1		
M1		

In the Table 6 we can see that modelled portfolio offers

1) less return and risk on D1 data, more risk on W1 data and more return and less risk on M1 data in the American market;

2) more return and risk on D1 data, practically identical return and risk on W1 and M1 data in the Latvian equity market.

To compare the effectiveness of two portfolio management it is necessary to choose an optimal portfolio from the optimal portfolio set. The authors used the tangent to the optimal portfolio set going from the point with zero risk and return. The structures of optimal portfolios are shown in the Table 5.

Table 5: The Structures of Two Portfolios

	COKE	F	MSFT
D1			
Traditional portfolio	46%	28%	26%
Modelled portfolio	54%	36%	10%
W1			
Traditional portfolio	44%	28%	28%
Modelled portfolio	52%	24%	24%
M1			
Traditional portfolio	70%	22% sell	8%
Modelled portfolio	74%	18% sell	8%
	<b>GZE1R</b>	<b>GRD1R</b>	<b>VNF1R</b>
D1			
Traditional portfolio	48%	26%	26%
Modelled portfolio	54%	28%	18%
W1			
Traditional portfolio	32%	38%	30%
Modelled portfolio	32%	36%	32%
M1			
Traditional portfolio	30%	48%	22%
Modelled portfolio	30%	48%	22%

As we can see the structures of optimal portfolios against their types differ in the American market but in the Latvian equity market the difference between the structures is significantly decreased on W1 and M1 data. In M1 data the expected return of one stock (F) is negative. It means that this asset must be sold, i.e. an investor should open the short position.

To determine the significance of this difference it is necessary to compare the cumulative returns of two portfolios on test period. The results are shown in the Table 6.

Table 6: Cumulative Portfolio Returns

	Markowitz's portfolio	Modelled portfolio
Portfolio of 3 American stocks		
D1	106.13%	106.82%
W1	91.31%	92.88%
M1	86.21%	92.30%
Portfolio of 3 Latvian stocks		
D1	144.73%	145.97%
W1	157.64%	155.77%
M1	167.22%	167.22%

The Table 6 shows some interesting effects, i.e.

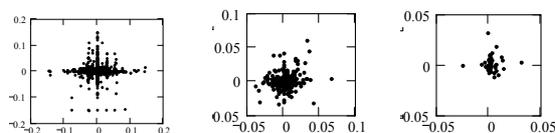
- 1) the quality of the modelled portfolio increases as used data time format increases in the American market;
- 2) the quality of the modelled portfolio decreases as used data time format increases in the Latvian market.

## CONCLUSIONS

Taking into account the given results the authors have come to the conclusions:

1. In most cases traditional Markowitz's portfolio is more optimistic than a portfolio formed with copula theory in the mature markets but practically does not differ in the emerging markets like the Latvian one.
2. The implementation of copula theory allows an investor to estimate portfolio risk more precisely in comparison with traditional Markowitz's approach in the mature markets. However, portfolio building based on copula theory is not rational in the emerging markets represented by the Latvian equity market because of complexity of computing and minimal deviation in the results (see the Table 5).
3. The difference between the cumulative returns in the portfolios formed utilizing copula theory and traditional Markowitz's approach is increasing function of time format in case of the mature markets and decreasing function in case of the emerging markets. It means that the difference is more evident in monthly data, less evident in weekly data and almost similar in daily data in the mature markets but vice versa in the Latvian equity market.
4. The features of non-linear dependences between Latvian stocks require to use more specific copulas for quality increasing of the corresponding portfolio

(see Figures 3). It means that it is necessary to find such copulas which could describe these unusual non-linear dependences.



Figures 3: Examples of Non-linear Dependences Between Two Latvian stocks (Daily, Weekly and Monthly)

scientific conferences. She is involved in elaborating of scientific research projects (grant) sponsored by Latvian Government and Scientific Council of Latvia (2001 – 2003; 2004 – 2007). Her e-mail is: natalja.lace@rtu.lv.

## REFERENCES

- Frees, E. W., and Valdez, E. A., “Understanding Relationships Using Copulas”, North American Actuarial Journal, V 2, No 1: p. 1-25.
- Harry M. Markowitz (The Rand Corporation), Portfolio Selection, The Journal of Finance, Vol. VII, No. 1, March 1952, p. 77-91
- Harry M. Markowitz, Foundations of Portfolio Theory, Journal of Finance, Vol. 46, Issue 2, June 1991, p. 469-477.
- Lindskog, F., “Modelling Dependence with Copulas”, Master Theses-MS-2000-06, Department of Mathematics, Royal Institute of Technology, Stockholm, Sweden, 2000
- Nelsen, R. B., “An Introduction to Copulas”, Springer Verlag, 1999

## AUTHOR BIOGRAPHIES

**VLADIMIR JANSON** (Dr.math) was born in Daugavpils, Latvia and went to the Latvian University of Latvia, where he studied mathematical science and obtained his doctoral degree in 1970. He worked eight years for Latvian University Computational Centre. From 1978 he worked in Latvian Technical University where he obtained doctor degree in mathematical science in 1983. The main field of research is simulation and optimization of economical systems. His e-mail address is: vladjans@latnet.lv.

**KONSTANTINS KOZLOVSKIS** (Mg.oec.) was born in Daugavpils, Latvia and went to the Riga Technical University, where he studied economics. He has been working for two years in the Latvian equity market and three years in FOREX as an independent trader. The main field of research is financial market analysis in the ultra-short time periods, building of mechanical trade systems, short-time investments. His e-mail address is: kosta@itunbizness.lv.

**NATALJA LACE** (Dr.oec.) graduated from Riga Technical University, Faculty of Engineering Economics. She has the degree in Economic sciences (1990). She is the Head of branch of study program “E-Business”. She has published more than 40 scientific articles and several textbooks, participated in different

# SIMULATION OF DAILY RUNOFF AND WATER LEVEL FOR THE LAKE BUTRNIIEKS

Ansis Ziverts  
Faculty of Rural Engineering  
Latvian University of Agriculture  
Akademijas 19, Jelgava LV-3001, Latvia  
E-mail: [aziverts@apollo.lv](mailto:aziverts@apollo.lv)

Elga Apsite  
Faculty of Geographical and Earth Sciences  
University of Latvia  
Raina bulv. 19, Riga LV-1586, Latvia  
E-mail: [elga.apsite@lu.lv](mailto:elga.apsite@lu.lv)

## KEY WORDS

Hydrological models, mathematical modelling of hydrological processes, climate changes, reservoir routing, the Lake Burtnieks.

## ABSTRACT

The Lake Burtnieks with around lying areas is one of the unique objects of the nature in Latvia. In this paper the analysis of mathematic modelling results for the rivers' runoff in the Lake Burtnieks watershed and water level of the lake are presented. First time for the Lake Burtnieks is demonstrated possibility to utilise regular observations of meteorological elements and to use mathematical model, which is adapted to natural conditions for the simulation of daily runoff and water level with a high statistical significance. The efficiency criterion  $R^2$  differs from 0.57 to 0.80 but the correlation coefficient  $r$  is from 0.8 to 0.9. Changing meteorological data according to the given scenario of climate changes we can obtain different parameters of the predictable hydrological regime in the future. Results of the study are widely applicable including the calculation of nutrient loading from the catchment area as well.

## INTRODUCTION

The Lake Burtnieks is the fourth largest lake in Latvia and locates in North-Easter part of Latvia. The lake is the source of the River Salaca, in which valley one of the largest complex of nature reserves in Latvia is located. The surface area of lake is 40.06 km<sup>2</sup>, and total drainage area - 2215 km<sup>2</sup> (which occupies 62% of the River Salaca basin). The Lake Burtnieks is shallow with average depth - 2.2 m. The climate is temperate, cool and humid. The average temperature of a year ranges from +5.0 to +5.5 °C. The mean temperature is - 6.5 °C in January and +17 °C in July. The average amount of precipitation ranges from 650 to 760 mm per year.

During last centuries different management actions have been carried out in the Lake Burtnieks watershed. However, not always all information about lakes and hydrological regime of lakes' hydrological networks

are possible used for the analysis of ecological situation or necessary conclusions. One of the explanations is that not all parameters of hydrological regime have been observed. Therefore, mathematical model as one of more accepted tools in hydrology could be use (Bergström 1991; Bergström 1992; Brandt and Arheimer 1998; Podstchine et al. 1999; Ziverts and Jauja 1999).

In this paper the analysis of mathematic modelling results for the rivers' daily runoff in the Lake Burtnieks watershed and water level of the lake are presented. Also, first time for the Lake Burtnieks is demonstrated possibility to utilise regular observations of meteorological elements and to use mathematical model, which is adapted to natural conditions. As a catchment example of the relationship between the conceptual model and the hydraulic reservoir routing model is presented as well. The simulated hydrological data are with a high statistical significance and have been applied in the project Water Protection Project of the Lake Burtnieks for the further calculations of nutrient loading and the different scenarios of climate changes (Bilaletdin et al. 2004).

## MATERIALS AND METHODS

The watershed model of the Lake Burtnieks has developed for the simulation of daily runoff and water level. This model is based on the specific hydraulic routing model of the Lake Burtnieks and the conceptual models METUL and METQ98 (Krams and Ziverts 1993; Ziverts and Jauja 1999) which are successfully applied to a relatively large river basin in Latvia as the River Daugava.

In present study the Lake Burtnieks watershed (part of the Salaca River basin) is divided in 7 sub-basins and one additional sub-basin between an outlet of the Lake Burtnieks and a gauging station Mazsalaca at the River Salaca (Fig. 1). To consider the runoff heterogeneity in runoff processes the catchment and its river basins and sub-basins are divided in hydrological response units (HRU) characterised by a relative homogeneity with respect to the most important parameters, which include slope, vegetation and soil characteristics. As in the model METQ98 each sub-basin is divided in 5 HRUs: agricultural lowlands, hilly agricultural lands,

forests, swamps and lakes. The total number of HRUs

for the investigated drainage basin is 40.

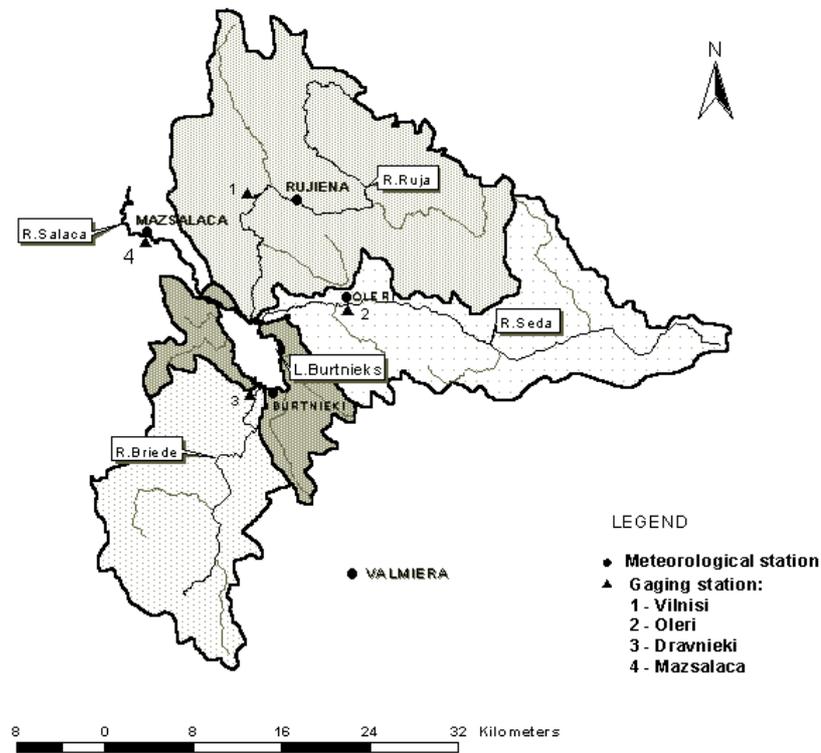


Figure 1: The Sub-basins, the Gaging and Meteorological Stations in the Watershed of the Lake Burtnieks

The model METQ98 applied is a mathematical model for the simulation of the daily runoff and evaporation for the rivers with different catchment areas. Input data for the model are daily mean values of air temperature, precipitation and vapour pressure deficit. The model can be classified as conceptual model and has 22 parameters. However, most of the parameters are physically based and the rest of parameters could be estimated by the calibration. The analysis of the model parameters is based on observed hydrological and meteorological data of the Vienziemite Brook Basin (Ziverts and Jauja 1999).

The water balance and runoff of each HRU has been simulated in three storages: snow (water content in snow cover), soil moisture (water in the root zone) and groundwater.

The total runoff from each of HRU consists of three runoff components:  $Q_1$  - surface runoff,  $Q_2$  - subsurface runoff (runoff from the groundwater upper zone) and  $Q_3$  - base flow (runoff from the groundwater lower zone). The snow accumulation and melting routine in the model is similar to the one used in the HBV model (Bergström 1992). The main difference between the METQ and HBV models is that the degree-day ration in METQ does not have a constant value, but it has a temporal difference depending on the daily potential insolation of each particular day.

In the watershed model of the Lake Burtnieks the runoff routing of river channel has been simulated by modified method of the unit hydrograph (a sum of the runoff components  $Q=Q_1+Q_2+Q_3$  has been calculated for one day intervals in the each gaging station and then transferred to the downstream at mouth of each sub-basin).

A principally different approach for the hydraulic routing of the Lake Burtnieks was used. The approach is based on common hydraulic methods of open channels. There was a lacking of channel measurements to obtain discharge rating curve  $Q=f(H)$  at the outlet of the Lake Burtnieks. However,  $Q=f(H)$  on the bases of the typical parameters of Latvian river channels (Golubovskis 1993) has been calculated. More about used parameters, structure and calibration of the watershed model of the Lake Burtnieks you could find out in the project report (Bilaletdin et al. 2004).

## RESULTS AND DISCUSSION

The observed meteorological data at station Rujiena and the precipitation measurements at stations Burtnieki, Mazsalaca, Oleri and Valmiera for the

simulation of the daily runoff in the mathematical modelling of Lake Burtnieks watershed have been used. The time series at least 5-year period of four river discharge (River Briede-Dravnieki, the River Rujavilnisi, the River Salaca-Mazsalaca, the River Seda-Oleri) and one water level of the lake (Burtnieki) stations have sufficiently good data for a successful calibration of the distributed rainfall-runoff model for the watershed of the Lake Burtnieks. Also the used rainfall-runoff model METQ98 simulates total runoff using two approaches: (1) separately for the HRU: hilly agricultural land, agricultural lowlands, forests and swamps, and (2) separately for surface runoff and subsurface runoff in each HRU. However, in this study for the water quality model, the first approach was used (Bilaletdin et al. 2004).

The Figure 2 shows good coincidence between the measured and simulated daily discharges. To analyse the results a statistical criterion  $R^2$  (Nash and Sutcliffe 1970) and a correlation coefficient  $r$  are used. The efficiency criterion  $R^2$  varies from 0.57 to 0.80 but the correlation coefficient  $r$  is from 0.80 to 0.90 for the tributaries of rivers in the Lake Burtnieks watershed. Also water level calibration of the Lake Burtnieks shows good results: the statistical criterion  $R^2$  is 0.58 and the correlation coefficient  $r = 0.83$  (Fig.3).

The main source of difference between the simulated and observed runoff values is the quality of

precipitation input data, as well as the location of the available meteorological stations to characterise the spatial and temporal distribution of precipitation in the drainage basin of Lake Burtnieks. The lowest statistical criterion  $R^2$  0.57 was found for the River Seda. It could be explained by a flat and broad flood plain and a high percentage of the wetlands in the river drainage basin. These reasons determine a specific hydrological regime of the River Seda which differs from others.

As a result of modelling of the investigated drainage area discharge at 8 points for the 10-year period (from 1990 to 1999) were simulated. The results of the model calibration show that the developed mathematical model for the Lake Burtnieks watershed is widely applicable including the calculation of nutrient loading from the catchment areas as well (Bilaletdin et al. 2004).

Also in this watershed modelling the scenario of climate changes for the next 10-, 30- and 60-years were used. Possible hydrological regimes of the Lake Burtnieks and its basin in the future have been obtained. The results are obviously presented in Figure 4. Having analysed the scenario of climate changes, we could conclude that main tendency in runoff changes are following: decreased in spring flood peaks and as opposite - increased in winter thaw-period.

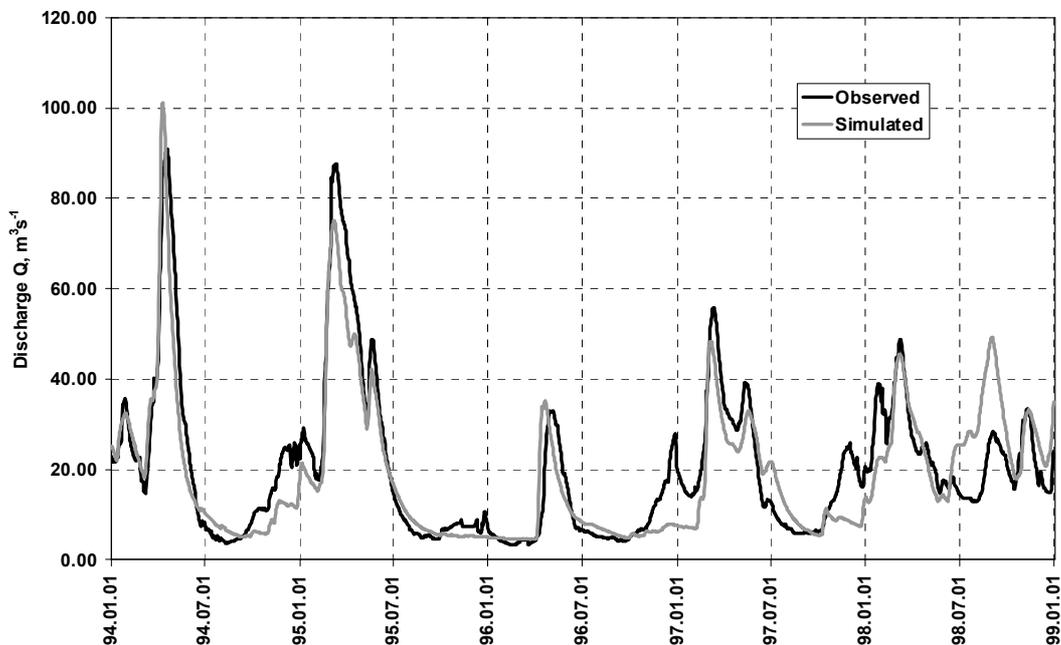


Figure 2: Observed and Simulated Hydrographs at River Salaca – Mazsalaca ( $R^2=0.8$  and  $r=0.9$ )

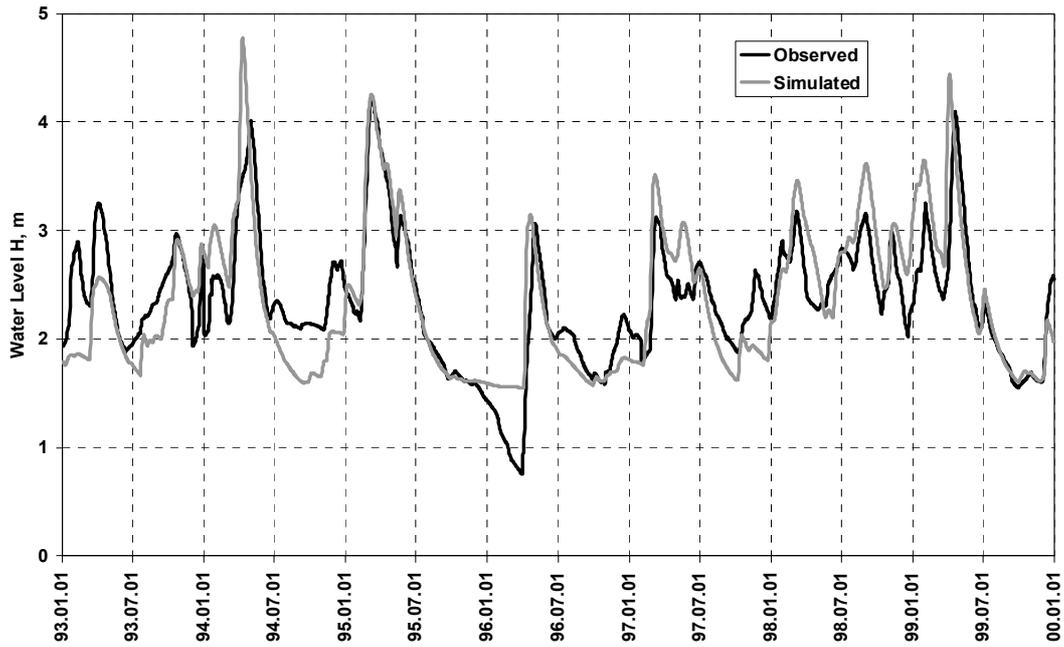


Figure 3: Water Level Calibration of the Lake Burtnieks ( $R^2=0.58$  and  $r=0.83$ )

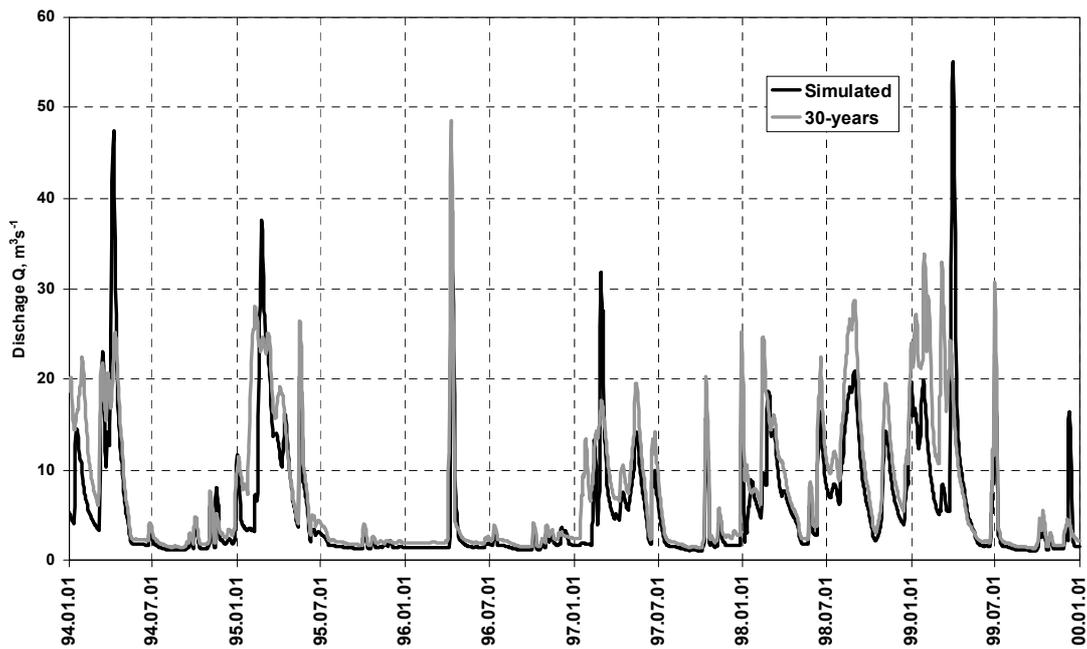


Figure 4: Simulated Hydrographs under the Scenario of Climate Changes at Present and After 30 Years at the River Ruja – Vilnisi

## CONCLUSIONS

The obtained results of modelling have revealed that the developed mathematical watershed model of the Lake Burtnieks are widely applicable including the simulation of different hydrological processes in

watershed - daily runoff and lake water level fluctuations to the given scenario of climate changes. Also calculation of nutrient loading from the catchment area and to create preconditions for successful forecasting of hydrological and hydrochemical regime in the basin are possible.

## REFERENCES

- Bergström, S. 1992. "The HBV Model - Its Structure and Applications." SMHI Reports Hydrology, No 4 (April)
- Bergström, S. 1991. "Principles and Confidence in Hydrological Modelling." *Nordic Hydrology*, No.22, 123-136.
- Golubovskis, E. 1993. "Noteku un upju reguleto gultnu hidraulikas un noturibas problemas (Problems of Hydraulics and Stability on Small Regulated Riverbeds ). LLU, Jelgava, 30.
- Brandt, M. and B. Arheimer. 1998. "Nitrogen Transport to the Sea from Southern Sweden – Catchment Modelling, Retention and Source Apportionment." In *Proceedings of the XX Nordic Hydrological Conferenc* (Helsinki, Finland, Aug.10-13, 1998) Kajander J., 13-20.
- Podsetchine, V.; T. Huttula; and H. Savijärvi. 1999. "A Three Dimensional-Circulation Model of Lake Tanganyika." *Hydrobiologia*, No.407, 25-35.
- Krams, M. and A. Ziverts. 1993. "Experiments of Conceptual Mathematical Groundwater Dynamics and Runoff Modelling in Latvia." *Nordic Hydrology*, No.24, 243-262.
- Nash, J.E. and J.V. Sutcliffe. 1970. "River Flow Forecasting Through Conceptual Models. Part I-A discussion of principles, *Jornal of Hydrology*, No.10, 282-290.
- Ziverts, A. and I. Jauja. 1999. "Mathematical Model of Hydrological Processes METQ98 and its Applications." *Nordic Hydrology*, No.30 (2), 109-128.
- Bilaletdin, Ä.; T. Frisk; H. Kaipainen; A. Paananen; H. Perttula; M. Klavins; E. Apsite; and A. Ziverts. 2004. "Water Protection Project of Lake Burtnieks." The Finnish Environment No.670, Pirkanmaa Regional Environment Centre, Tampere.
- Ziverts, A. and E. Apsite. 2001. "Watershed Modelling of the Lake Burtnieks in the Present Conditions and Considering the Effect of Climate Change." In *Proceedings of the 9<sup>th</sup> International Conference on the Conservation and Management of Lakes* (Conference Proceeding Sessions 5.5 B-P-31 Biwako, Japan), 292-295.

## AUTHOR BIOGRAPHIES

Ansis Ziverts was born in Balvi, Latvia. He went to Latvia Academy of Agriculture, where studied water resources engineering and obtained his degree of engineer in 1959. He worked over 30 years at the Latvian Land Reclamation Design Institute. Hi obtained doctor of hydrology at the Moscow Hidrometeocentere in 1966. Since 1991 he is Professor of Water Resources Engineering at the Latvia University of Agriculture, Faculty of Rural Engineering. Also hi is corresponding member of

Latvia Academy of Science. His research interest is connect with development of the methods for determining of hydrologic design quantities, unsteady water flow in river beds and flood plains, mathematical modeling of the hydrological processes and water management systems. His e-mail address is: aziverts@apollo.lv

Elga Apsite was born in Madona, Latvia. In Riga she studied geography and environmental sciences at the University of Latvia and obtained doctor of geography in 1999. She worked for a couple of years for the Institute of Geodesy and Geoinformation and the Latvian Environment Agency. Since 1991 she works at the University of Latvia, Faculty of Geography and Earth Sciences, now she is Associate Professor. Her research interest is connect wit long term hydrological regime changes, diffuse pollution of nutrients, environmental modelling and water resources management. Her e-mail address is: elga.apsite@lu.lv

# SIMULATION OF RADIOWAVE PROPAGATION USING PROPAGATION MODELS

Yelena Chaiko

Transport Communications and Information Systems professor group  
Riga Technical University Institute of Railway Transport  
Indrika, LV-1004, Riga, Latvia  
E-mail: krivcha@inbox.lv

## KEYWORDS

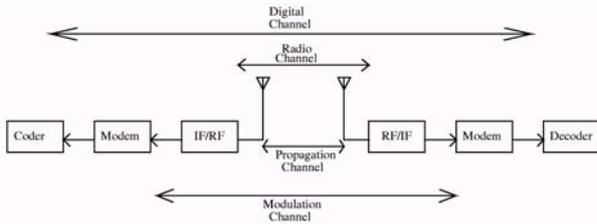
Model, propagation channel, radio communication system, pathloss, base station, simulation.

## ABSTRACT

In this article we review the fundamental issues involving the simulation of the radio channel and discuss some of the more challenging simulation topics that are pertinent to the radio channel.

## Radio and propagation channels

The propagation channel, together with the transmit and receive antennas, constitute the *radio channel*. Figure 1 depicts the propagation system and radio channel's place in a radio communication system.



Figures 1: Various channels in a communications system.

The net effect of reflection, diffraction and scattering on the transmitted signal is attenuation, delay and phase change. Formally, this can be seen by starting from examination of the familiar wave equation

$$\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} + \frac{\partial^2 \phi}{\partial z^2} = \frac{1}{c^2} \frac{\partial^2 \phi}{\partial t^2} \quad (1)$$

which describes the propagation of waves. The wave equation has the well-known plane-wave solution which in one dimension (corresponding to horizontal propagation of vertically polarized field) is given by:

$$\phi(x, t) = S(x, t) e^{j\omega_c(t - x/c)} \quad (2)$$

where  $s(x, t)$  is the information bearing (or complex envelope) of the wave propagating in the  $x$  direction and  $\omega_c$  the carrier frequency in radian/sec. Letting the delay

$\tau = x/c$  and making the spatial dependency  $x$  implicit we have

$$\phi(t, \tau) = [S(t - \tau) e^{-j\omega_c \tau}] e^{j\omega_c t} \quad (3)$$

In a multipath environment,  $r(t)$ , the complex low-pass representation of the received signal is the contribution of many rays:

$$r(t) = \sum_n \alpha_n(t) S(t - \tau_n(t)) e^{-j\omega_c \tau_n(t)} \quad (4)$$

where  $\alpha_n(t)$  denotes the time varying complex amplitude of the  $n$ th ray. Note that in addition to the time varying amplitudes  $\alpha_n(t)$ , the delay of each path is also a function of time. Equivalently, the RF equivalent counterparts of  $r(t)$  and  $s(t)$  denoted by  $S(t)$  and  $R(t)$  are

$$S(t) = \text{Re} \left\{ S(t) e^{j\omega_c t} \right\}$$

$$R(t) = \Re \left\{ \sum_n \alpha_n(t) S(t - \tau_n(t)) e^{j\omega_c(t - \tau_n(t))} \right\} \quad (5)$$

## Signal impairments

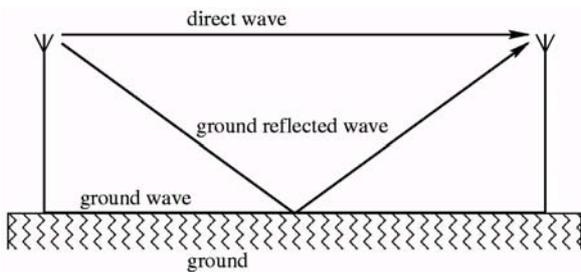
Communication quality between a base station and mobile or portable transceiver depends, among other factors, on the type and degree of impairments the signal undergoes in the radio channel. These impairments are now reasonably well understood and hence need to be included as essential parts of a realistic radio channel model. Some of the more important impairments are:

### 1. Pathloss

The transmitted signal suffers a loss proportional to  $1/R^n$  where  $R$  is the distance between transmit and receive antennas and  $n$  is a positive number typically between 2 and 6. For free space transmission,  $n = 2$  and the free space pathloss is given (in dB) by

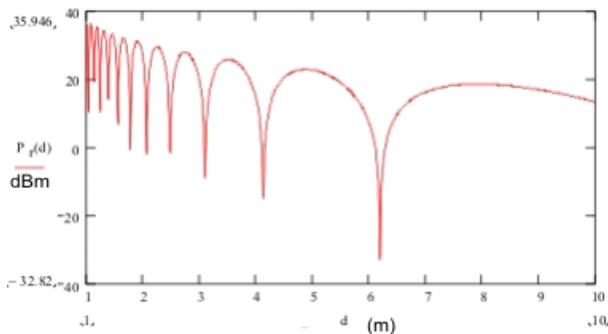
$$L_{FS} = +20 \log_{10} f_c + 20 \log_{10} R + 32.4 \quad (6)$$

where R is distance between transmit and receive antennas (in km) and  $f_c$  is the carrier frequency in MHz. Although free-space assumption is quite optimistic, it provides a useful figure of reference for pathloss. There are a number of more useful pathloss models for the propagation channel that are based on either theory, extensive measurements, or both. One such model is the analytical "flat earth" two-ray model that assumes a direct and a reflected ray coming from the ground as illustrated in Figure 2. This model takes into account the antenna heights, polarization of waves and the complex reflection coefficient of the earth surface.



Figures 2: The two ray, flat earth model.

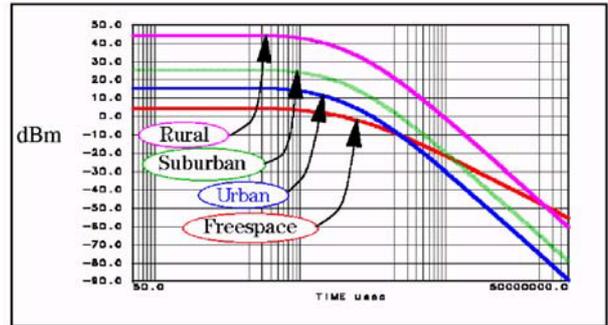
In this model, there are two regions with different slopes separated by a breakpoint, beyond which the pathloss follows a  $1/R^4$  law or 6 dB per octave. In the first region, however, the received signal oscillates due to destructive and constructive addition of the two rays. This phenomena can be described by the Fresnel zone clearance, where the breakpoint can be viewed as the distance for which the ground just begins to obstruct the first Fresnel zone. Figure 3 depicts the pathloss based on the flat-earth model as the mobile drives away from the base station with a constant velocity. The x-axis can be viewed as distance between MS and BTS.



Figures 3: Losses of capacity of a signal in a point of reception  $P_r(d)$  (dBm) as function from distance  $d$  (m).

A measurement-based model for pathloss in the mobile environment has been developed by Hata [1]. This model was developed from extensive data gathered by Okumura [2]. Hata's model predicts pathloss in urban, suburban and rural areas and have antenna heights and

frequency as parameters. Figure 4 shows the comparison of Hata's pathloss models for different environments as a function of distance (or time).



Figures 4: Comparison of Hata's pathloss models for different environments.

### Simulation of radio channel

Today, simulation is increasingly replacing extensive and costly field measurements. The use of productivity tools, such as OmniSys, has merits over hardware channel simulators particularly during the design stages because it allows the user to study the radio channel interaction with RF/IF front end and modems without need for a costly and difficult hardware interface.

There are numerous models of propagation medium based on analytical and empirical studies. In general, these models can be classified as deterministic and statistical. Deterministic models are useful for predicting the signal strength at different locations and in presence of obstacles. These techniques have also the added advantage of being able to incorporate the effect of antenna radiation patterns as well as polarization of the fields. Deterministic approaches like Geometric Theory of Diffraction (GTD), Uniform Theory of Diffraction (UTD) or Ray Tracing, are computationally intensive and require physical environment data. In spite of promising results, particularly for indoor environments, none of the deterministic modeling approaches have gained the flexibility or computational speed to be incorporated into an overall simulation of the digital channel. In addition, both GTD and UTD have serious limitations for predicting signal level in outdoor macrocell environments.

Statistical models treat the physical attributes of the medium as processes with certain distributions generally derived from measurements, deduction or both. Consequently, statistical models do not provide the user with accurate quantitative measure of the signal. Rather, they provide qualitative description of signals such as fluctuations, fade margins and the rate and duration of signal impairments. These models are ideal when a system perspective is sought. Common examples would be the performance evaluation of a given equalizer under frequency selective fading conditions or that of an AGC or power control loop as the signal degrades due to a flat fade or interference.

### Propagation models

The (low-pass) impulse response of the propagation channel  $h(\tau, t)$ , characterized by several discrete paths, each having a specific delay and attenuation, can be deduced from equation (4) as

$$h(\tau, t) = \sum_n \alpha_n(t) e^{j\theta_n(t)} \delta(\tau - \tau_n(t)) \quad (6)$$

If  $h(\tau, t)$  is modeled as a zero mean Gaussian process, the envelope  $|h(\tau, t)|$  at any time is Rayleigh-distributed. The transform of  $h(\tau, t)$  with respect to time, gives the spectrum of time variation  $S(\tau, \nu)$ , generally referred to as delay-Doppler-spread function [1].

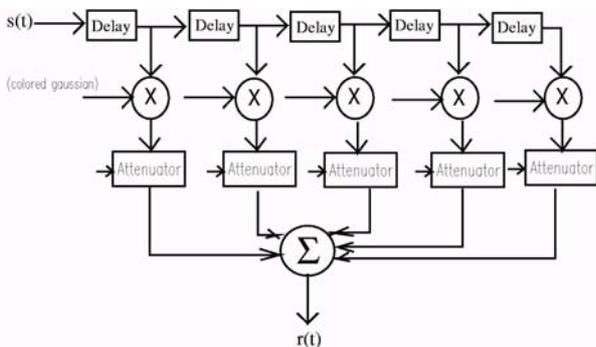
It has been shown [5] that for the case of two vertically polarized transmit and receive antennas and horizontal propagation of plane waves,  $S(\tau, \nu)$  for a fixed delay (Doppler spectrum) is given by

$$S(\nu) = \frac{1}{2\pi\nu m \sqrt{1 - \left(\frac{\nu}{\nu_m}\right)^2}} \quad \nu \leq |\nu_m| \quad (7)$$

where  $\nu_m = \frac{V}{c} f_c$  is the maximum Doppler shift due to vehicle speed  $V$ . When a direct path exists such that the total multipath contribution is equal to that of the direct path the spectrum is Rician and is given by

$$S(\nu) = \frac{k_1}{2\pi\nu m \sqrt{1 - \left(\frac{\nu}{\nu_m}\right)^2}} + k_2 \delta(\nu - k_3\nu_m) \quad \nu \leq |\nu_m| \quad (8)$$

with  $k_1, k_2, k_3$  constants related to proportion of direct and scattered signal and the direct wave angle of arrival.



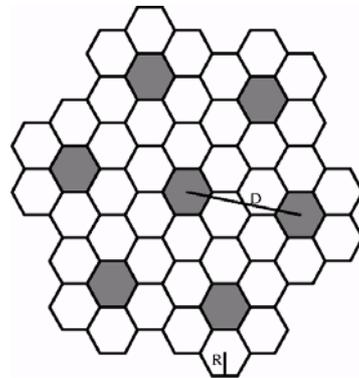
Figures 5: Tapped delay line channel model with frequency selective and flat fading.

A multipath fading model can therefore be constructed using a tapped-delay line filter. The typical tapped-

delay line filter model for simulation is represented in Figure 5. To generate a Rayleigh fading profile for each path, independent added white Gaussian noise (AWGN) sources, in cascade with a filter representing the effects of Doppler spread can be used.

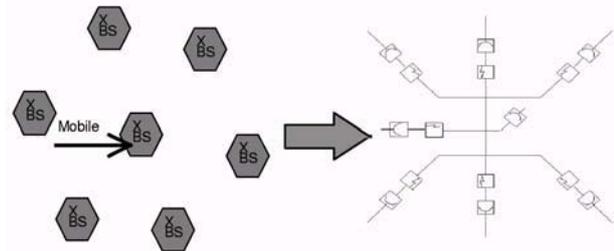
### Cellular layout

Cellular layout typically involves the frequency reuse factor which is inversely proportional to  $K$  (number of cells per cluster). Figure 6a depicts the cellular layout with  $K=7$  for a TDMA system. The shaded cells use the same frequency plan. The co-channel interference (CCI) is the most serious problem in this scheme while adjacent-channel interference is usually not a problem.  $D$  and  $R$  decide the interference levels where  $R$  is the radius of a cell and  $D$  the distance from center of a cell to its adjacent cell with the same frequency plan, as shown in the figure.  $D$  and  $R$  and number of cells in a cluster are related by  $D/R = \sqrt{3K}$ .



Figures 6a: Cellular layout for frequency reuse.

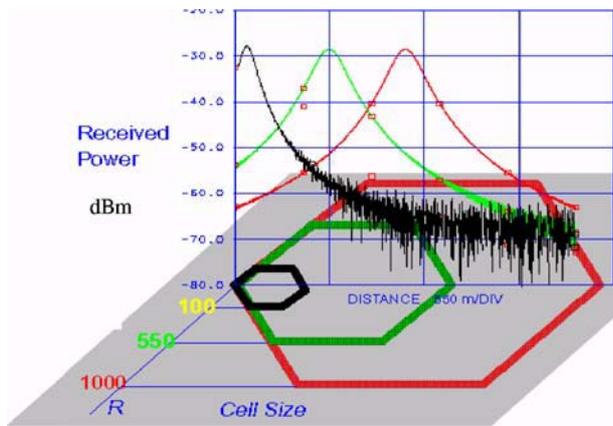
In spread spectrum systems (such as IS-95 CDMA) there is no frequency reuse and all cells use the same frequency band. This is possible because of the processing gain (21 dB) obtained by the use of quasi-orthogonal codes. In CDMA, mobiles are power controlled to equal power at cell site.



Figures 6b: Co-channel interference setup with equivalent schematic in OmniSys software.

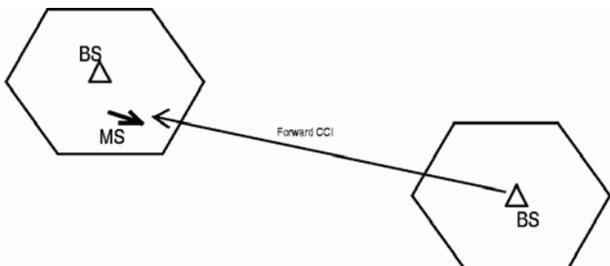
In TDMA cellular applications, the designer should also make sure that the power level at the cell boundaries is weak enough not to spill over into the adjacent cells. By placing a home base station at the center and the interfering stations on the perimeter of a circle of radius  $D$ , one can simulate the received signal plus co-channel

interference at the mobile antenna coming from its own and interfering base stations or what is known as forward CCI. Figure 6b depicts this cell arrangements and its schematic representation. Forward CCI becomes significant when the path between the mobile and its own base is obstructed such that the desired signal is very weak. This scenario occurs particularly around the boundaries of cells. Figure 6c shows the received signal power for three different values of R as the mobile moves from the left edge of the cell with a velocity of 100 km/hr towards the right. These simulation results show that the interference level grows near the cell edges and beyond the cell boundaries. The simulation also shows that the interference level changes with the cell radius.

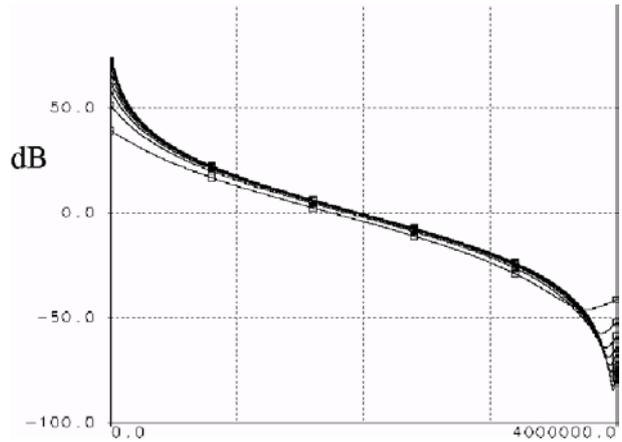


Figures 6c: Received signal power for three different values of D.

The change in signal to interference ratio (C/I) as a function of time as a mobile travels is a parameter of interest. Figure 7a depicts the setup, where a mobile is traveling from a home cell towards a co-channel cell in an urban environment. The C/I ratio is shown in Figure 7b as a function of time for various values of R. As shown, the C/I ratio is about 50 dB near the home base station and drops drastically as mobile moves toward the interfering base station.



Figures 7a: A mobile traveling from a home cell towards a co-channel cell in an urban environment.

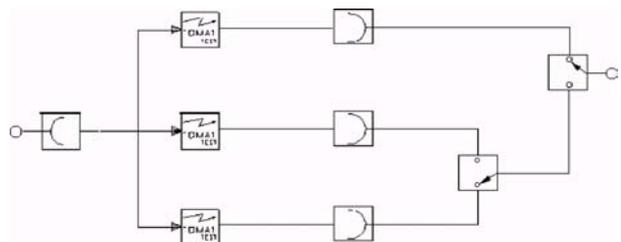


Figures 7b: C/I ratio as a function of time or distance.

### Diversity and Combining

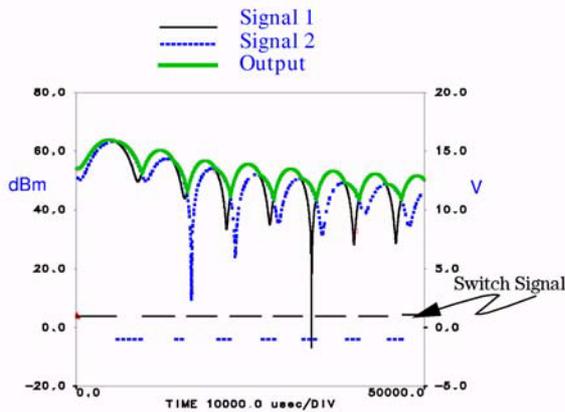
Antenna diversity is one of the methods for mitigating the fading signals. Space diversity relies on the fact that spatial fades occur about half a wavelength apart. Diversity reception relies on receiving a signal by combining two or more signals having very little correlation. The antennas are placed far enough such that the fade experienced by one antenna is not experienced by the other. Equivalently, since the rate, location and depth of fades is a function of carrier frequency, the transmission of two or more frequencies spaced far enough apart so that they are resolvable beyond duration of a fade constitutes frequency diversity.

There are basically two types of diversity reception, namely pre-detection and post-detection combining. Pre-detection combining is done via co-phasing of the randomly faded received signals or by picking the strongest signal among branches. Figure 8a depicts a reverse link design for reducing the fade margin by using a switched combining scheme. The mobile transmitter consists of a data source, GMSK modulator and the mobile antenna transmitting into the GSM propagation channel. The base station antenna system includes three omnidirectional antennas with a three-branch selection-combining scheme that includes two switches. The switch design is not described here, however it suffices to say that its function is to pick the signal with the stronger signal.

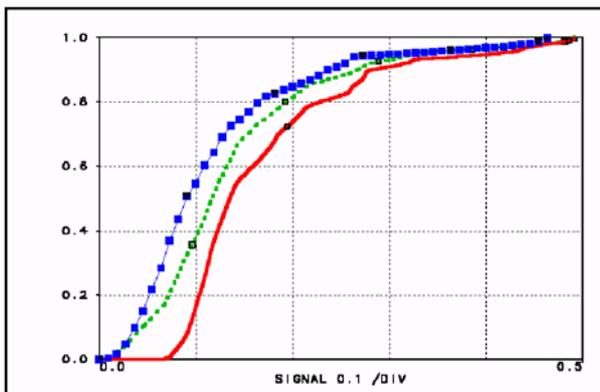


Figures 8a: Switched combining scheme for reducing the fade margin.

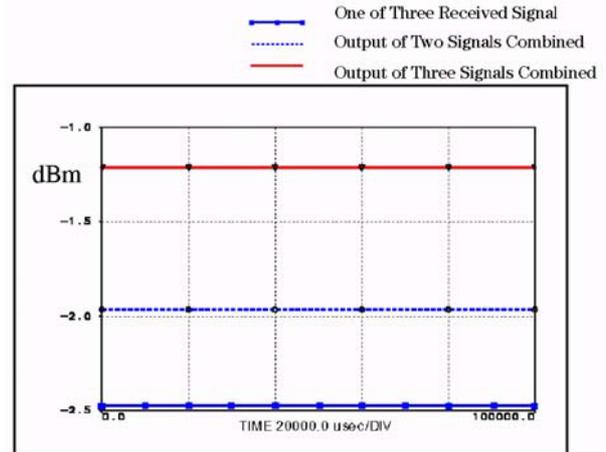
The upper trace in Figure 8b depicts the output of two antennas as well as the switch output. The lower trace shows the switching action for selecting either antenna. The output of the switch follows the stronger signal and hence avoids the deep fades of a single antenna. Figure 8c compares the cumulative probability density function (CDF) of one, two and three branch combining schemes, and Figure 8d depicts the average power from the two and three branch combiners, which shows a gain of up to three dB.



Figures 8b: Envelope of the switch input and output signals.



Figures 8c: Cumulative probability density function.

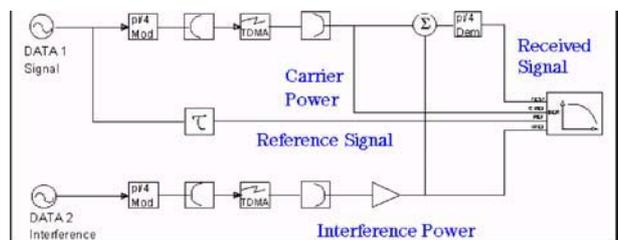


Figures 8d: Average power.

### BER Simulations

BER simulation for AWGN channels are well studied and documented while less effort has been spent on the study of BER for fading channels. Such simulations enable the wireless designer to test the performance of various modulation schemes under harsh fading conditions. In the following examples we use a number of BER simulation focusing on  $\pi/4$  DQPSK and GMSK modulation schemes used in IS-54 and GSM systems for various fading conditions.

Figure 9a depicts the design setup for simulating an interference-limited channel with  $\pi/4$  DQPSK modem. The interference limited assumption allows us to ignore AWGN and thereby concentrate on modeling the multipath fading in the presence of CCI. The simulation blocks include two identical but uncorrelated paths: one for the signal and the other for the interference. The signal and interference paths include a binary pseudo-random source, a  $\pi/4$  DQPSK modulator, a mobile antenna moving with a fixed velocity as a parameter, an IS-54 (TDMA) propagation channel model with flat and two-ray frequency-selective options, base station antenna, and a coherent  $\pi/4$  DQPSK demodulator including a carrier recovery. Note that two receive antennas are used for simulation to gauge the signal and interference separately, while in reality one antenna is used.



Figures 9a: An OmniSys simulation of an IS-54 system including fading radio channel with CCI.

The inputs to the modulator are the I and Q data streams in nonreturn to zero (NRZ) format. The NRZ data

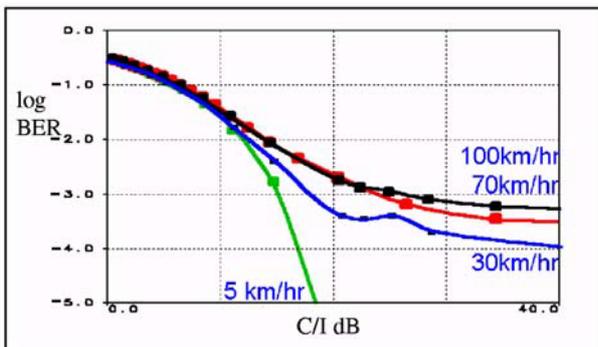
symbols are differentially encoded for the Pi/4 DQPSK format, filtered with raised cosine filters, and then modulated to the carrier frequency with a QAM modulator.

The IS-54-based two-ray propagation channel impulse response is a special case of equation (7) described by

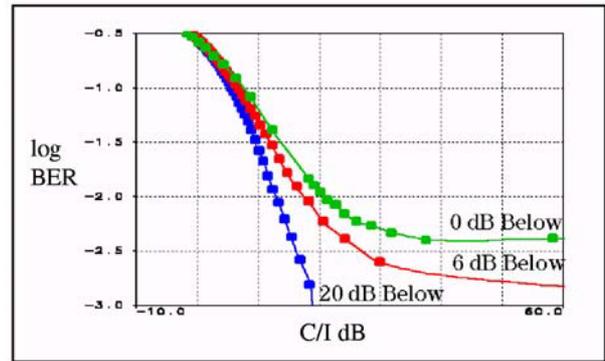
$$h(t, \tau) = \alpha_1(t)e^{j\theta_1} \delta(\tau - \tau_1) + \alpha_2(t)e^{j\theta_2} \delta(\tau - \tau_2) \quad (9)$$

Note that this model in general can be a flat fade ( $\alpha_2 = 0$ ) or frequency selective depending on the way amplitudes and delays are specified. We use this model to simulate the BER for both flat and frequency selective fading conditions. The coherent Pi/4 DQPSK demodulator uses a carrier recovery scheme to recover a reference carrier signal. The modulated signal and the reference carrier are fed into a QPSK demodulation scheme that recovers I and Q components of the signal, pass it through a square root raised cosine filter and finally a differential decoder to produce the binary data signals.

A BER measurement compares the transmitted bit stream with the received signal after it is demodulated. First simulation uses a data stream (48.6 Kbits/s) over a flat fading channel. The output of this channel is added to that of the interfering signal, which is a different pseudo-random stream modulated and transmitted over a statistically independent channel. The same data rate is used for the interfering signal but the interference power is changed, resulting in different values of C/I. Figure 9b depicts the BER performance at four different mobile velocities (5, 30, 70 and 100 km/hr) corresponding to Doppler frequencies of 4.58, 27.52, 64.21, and 91.73 Hz at 990 MHz. The BER simulation was performed with a time step of a tenth of a bit or roughly 2 microseconds and a total of 20,000 bits were processed in the simulation.

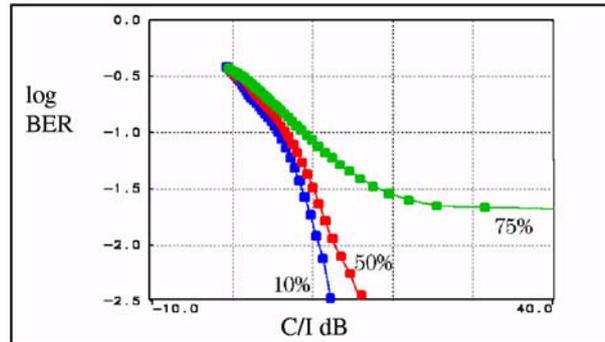


Figures 9b: OmniSys simulation of BER vs. C/I in NADC (IS-54) flat fading channel for different mobile speeds.



Figures 9c: OmniSys simulation of BER vs. C/I for different ratios of 1st over 2nd ray powers.

By fixing the mobile speed to 30 km/hr and using two-ray frequency-selective fading, the effect of power level in the second path can be simulated. The delay of the second path is set to a tenth of bit time. Figure 9c shows the BER simulation for various power levels of the second path with respect to the first. Figure 9d depicts BER as a function of C/I for various values of second path delay. Here, the power level on the second path is kept at 9 dB below the first, while the mobile travels at a very low speed (1 km/hr). Note that % delay refers to delay values as a bit-time percentile.

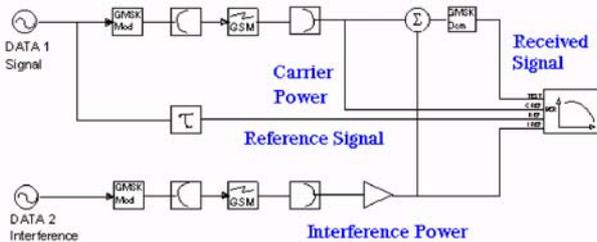


Figures 9d: BER vs. C/I for different 2nd ray delay values.

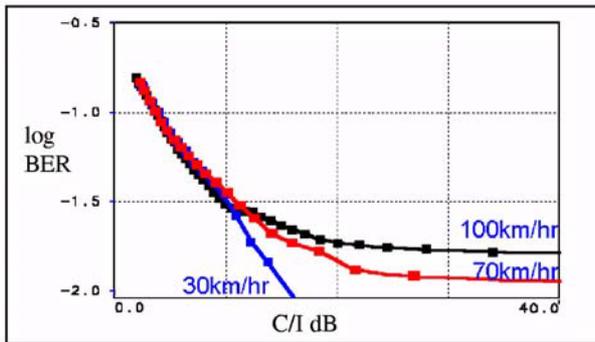
The second BER example simulates a GSM system including the OMSK modulator and a coherent GMSK demodulator. The GMSK modulator consists of a Gaussian filter, and an FM modulator with a sensitivity of  $1/(4STIME)$  Hz/volt where STIME is the input data bit time. The input bits are first hard limited into an NRZ format. The 3-dB bandwidth of the Gaussian filter is set to  $0.3/STIME$  (0.3 GMSK). The demodulator is a suboptimal coherent GMSK with built in carrier and clock recovery scheme. The propagation channel is one of the GSM options (rural area) with six taps and a delay profile with a maximum delay of 0.6 microseconds. This corresponds to about 16% of the

symbol duration (3.7 microseconds). Longer delay profiles (like hilly terrain in GSM) would require equalization. The propagation channel includes both flat and frequency-selective fading.

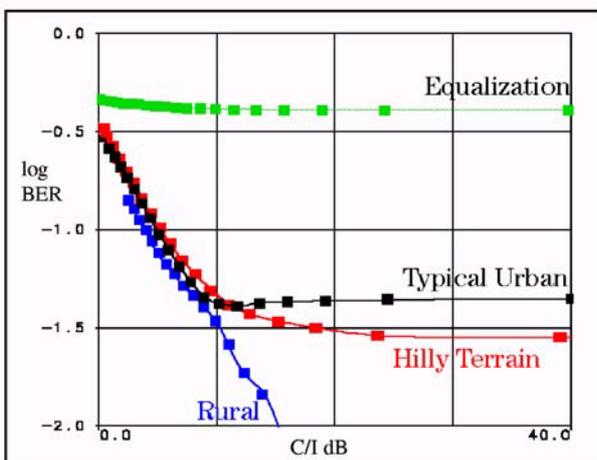
Figure 10a shows the schematic representation of the design including fading radio channel with CCI channel as well as the desired signal. Figure 10b depicts the BER vs. C/I for three different mobile velocities (10, 25 and 50 km/hr) corresponding to Doppler frequencies 8.3, 20.8 and 41.7 Hz at 900 MHz. Finally, Figure 10c shows the BER results for different propagation environments as specified in GSM standards. This result shows that in the absence of equalization, poor BER results are obtained for urban and hilly terrain (where delays are much longer than bit duration).



Figures 10a: GSM system including a fading radio channel with CCI.



Figures 10b: BER vs. C/I in GSM propagation channel for different mobile speeds.



Figures 10c: BER in different environments.

## Conclusion

System simulation is a powerful tool for the analysis and design of communication systems. The simulation of the radio channel is particularly worthwhile due to the otherwise extremely laborious and costly measurements. A simulation environment where the radio channel blocks are integrated with circuit, behavioral, and DSP models, allows the designer to study the interaction of radio, modulation, and digital channels simultaneously. Simulation of realistic mobile/portable scenarios where propagation takes place in uncoordinated, overlapping networks of interfering transmitters is clearly a desirable feature. Antenna and propagation modeling should be compatible with such realistic scenarios when due consideration is given to effects such as mobile travel and distance-dependent pathloss. Compatibility with industry standard such as GSM, allowance for fading, time and frequency spread and various faded signal distributions are also advantageous.

In this article, radio channel models consistent with above requirements were used, and a number of applications, including the impact of changing C/I on signal quality and the degradation of BER due to interference proximity were simulated. Also, using these radio channel models, a diversity combining scheme was shown to improve fading margins.

## REFERENCES

- Steele, Raymond, 1992.-“Mobile Radio Communications”, Pentech Press.
- Hata, M., 1980. “Empirical Formula for Propagation Loss in Land Mobile Radio”. IEEE Trans. VT-29, pp. 317-325.
- Okumura, Y., 1968 “Field Strength and its Variability in VHF and UHF Land Mobile Service.” Review of Electrical Communication Laboratory, Vol 16, pp. 825-873.
- GSM 05.05 1974.Recommendation, Radio Transmission and Reception.
- Jakes, W. C. (Editor),1974 Microwave Mobile Communications, John Wiley & Sons



**YELENA CHAIKO** was born in Riga, Latvia and she studied at Riga Technical University, Transport Communications and Information Systems professor group and obtained her degree in 2004. She works at Riga Technical University Her e-mail address is : krivcha@inbox.lv.

# MODELING AND SIMULATION OF PROCESSES IN THE SOIL AND GROUNDWATER ZONE

Peter-Wolfgang Graeber  
Department of Hydro Sciences  
Dresden University of Technology  
D-01062 Dresden  
Germany  
E-mail: graeber@rcs.urz.tu-dresden.de

## KEYWORDS

Decision support, Environmental science, Combined simulation, Hierarchical models

## ABSTRACT

The simulation of processes in the soil and groundwater zone is one of the prerequisites for controlling and monitoring of such applications like pumping wells in water works or drainage systems in mines and building pits as well as the remediation of contaminated sites and land fills. The storms, extreme heavy rain falls and floods of the last time require water management more than ever for extreme conditions and long-term climatic changes. Hence new strategies are needed to secure requirements of society and environment. On an international level comparable results and intergovernmental water management is required which means to learn from each other experience. To ensure sustainable water management strategies a holistic approach is emphasized. The bases of a modern water management system are computer based Decision Support Systems (DSS) including computer simulation tools with active graphical user interface. By using of modern mathematical solutions (e.g. Fuzzy-systems, GIS, Neuronal Networks) the prognosis of water level and flood prevention and control is possible. A DSS requires a coupled analysis of all kinds of data to describe the long-term scenario of global and local changes. To get measurable results benchmarks are defined to evaluate performance of the developed simulation tool. Water management needs will be set into a socio-economic context.

## INTRODUCTION

A deficit of water resources, unfavorable distributions of water in time and space, increasing water use, polluted water resources, variability and instability of climate and change of socio-economic activities demand rational ways of water management.

The goal of the DSS OSMOW computer program system is to develop a methodology for effective operational water management under normal and extreme conditions of the water cycle.

On December 22<sup>nd</sup> 2000 the General European Guidelines for Water was enforced. This guideline forms the basement for an inter-European cooperation for water protection, water management and catastrophe prevention and ensures a controlled management of surface and ground water. The General Guideline for Water requests harmonization of actions for water protection and contributes to the further decrease of water pollution. But much needs to be done to fulfill this goal. The General European Guidelines can be used not only inside the European Union, but also she is valid worldwide.

Consequent installation of a circumspect classification of watercourses especially under hydrological, economical and ecological aspects is an excellent chance to improve water management in Europe. This does not neglect the unique aspects of single river or lake system. Both must be enabled through:

- a consequent, aerial and river-specific approach
- a specific approaches for every water course
- the monitoring of sufficient data and its compilation in data bases,
- a combined consideration of pollutants (emission and imission) and
- a single pollutant or complex pollutant cocktail specific approach.

Paragraph 3 of the guidelines is a central theme of the guideline and every national water management authority will have to cope with its demands. Until now the water management is under supervision of national or even local authorities with no interaction to each other. Water management was enabled within political or geographical borders. But 50% of the residential areas are situated aside international water course systems and all of the water authorities responsible for the supply of water and waste water management have one common interest: to nationalize the advantages of water and to internationalize the disadvantages of water.

Due to the introduction of the New guideline water authorities will have to co-ordinate their actions on a broader and international scale. Along international river courses international river water management and construction plans must be introduced. International

river catchment areas must be handled adequately. In order to establish such water management and construction plans a huge amount of data for the evaluation, determination of management goals as well as action plans must be compiled. The interaction of climatic, social, agricultural, industrial, environmental and economical aspects have to be considered and controlled. An interactive decision for an action must be taken according to all that information. Water pollution and the measures for water protection will have to be revised continually. The management plan therefore must be revised every 6 years (prescribed by paragraph 4 of the guideline) and effects caused by previous actions, improvements and damages have to be evaluated. The status of the necessity of exceptional regulations has to be evaluated every time and, if the status has expired, the regulation must be withdrawn. All information must be stored and forms input data for an interactive decision support system for European water management.

The co-ordination of information exchange and decision consulting is in the first place a management task, which challenges experts and is the only way to ensure stable supply of water and to protect the environment. The living standard, agricultural production and modern industry depend on a stable supply of water and wastewater treatment.

Catastrophes caused due to absence or strong presence of water occurs with the highest probabilities. Natural catastrophes due to flooding or catastrophes caused by water such as slope sliding, dam failure or even the failure of concrete reservoir constructions due to flood impact as well as hygienic impacts, which come in the course of such events, are strongly burned into our mind. But also sneaking quiet catastrophes have a terrible and ugly face such as dry seasons. The more extreme a catastrophic event becomes the less probable becomes its occurrence. But this actually forms the danger of natural disasters. The danger that comes along with these events is difficult to predict and measures against such events are difficult to master. Prediction of such events, measures against floods and the planning of residential areas and industrial areas away from areas prone to floods is one effective measure. The construction of reservoirs for both flood protection and dry-out prevention becomes an international and intercultural task especially on river courses, which are situated in several countries. Water management for prevention of natural catastrophes must deal with

- the protection against catastrophes of high probability, but under economically considerations and with a calculable remaining risk,
- the remaining risk must be at a scale that the national economy is able to carry the damage and is not turned into long-term disaster.

Concerning the General European Guideline for Water much research demand exists for a variety of topics. The

research proposed deals with the classification of water courses using a circumspect approach, deals with compilation and evaluation of ecological aspects and in the end prepares a tools for fast and efficient water management for both flood prevention, dry-out prevention and pollutant impact under economical, ecological and sustainable aspects.

The main problems to be interpreted in the proposal are delineated in connection with: Water Resources, Water Users and Operational Water Management. The scientific/ technological objectives and the solution of these problems are outlined in general objectives and measurable targets are described in the specific problems, respectively.

The current research proposal deals with the adaptation of models and the development of management scenarios for catastrophe protection.

The addressed general objectives of the to developed project include:

1. evaluate the danger of catastrophes,
2. evaluate the vulnerability for disasters of an area,
3. manage the catastrophe prevention and catastrophe protection,
4. improve the model output of interpolation and prognosis for managing tasks for water management based on circumspect data and its handling in an interactive data base.
5. consider hydrological, meteorological, biological, biochemical and ecological influences for decision-making.
6. Furthermore the need for water supply and protection against dry seasons must be handled. Water resources therefore must be qualified regarding quantity and quality of the water reservoir. Decision support systems will be needed and therefore the development deals with:
  - a. coupled analysis of water supply safety and the safety of food supply. This involves the use of information and an integrated analysis method of water availability, water usage, food demand, and economical aspects for agriculture, and
  - b. improvement of respective development for decision support for the management of catchment areas with help of input data describing the long-term scenario of global climatic changes.
  - c. The decision support system must supply tools to help to improve the co-operation and communication of involved parties.
7. The decision support system also must help the governments and the private industries to decide and take action.
8. The decision support systems developed within the presented approach will help by the economical controlling and steering of water supply systems, waste water treatment plants and help to evaluate the environmental impact of actions taken.

## STATE OF ART

The motivation to submit the program system OSMOW is the necessity to develop tools - methods, programs, techniques, etc. - which will enable a decision support system for water management in Europe as requested by previously published guidelines. Present practices of many European countries may need considerable alterations. Water economy systems ensuring in general needs of different water users, are methodically and technically unfit in particular cases and cannot be operated in synchrony with user requirements, for example - during unexpected droughts and floods. The operation of systems managing groundwater flow for instance is in particular exposed to retardation. The response to user needs in large open channel systems and an inter-European approach is retarded as well.

The aim of the to be developed OSMOW is to develop a methodology, contributing Decision Support System for the sustainable water resources management. The methodology for water management under normal and extreme conditions of water cycle builds up on management of water resources for normal cases.

The approach to the solution of considered management problems combines the best available scientific knowledge and methodologies. On basis of existing and own-developed numerical and balance models of surface and ground water flows and other dynamic processes, a new simulation and controlling system on a regional scale is constructed. The controlling system is based on a hierarchical online closed-loop concept with two simulation levels. Water needs of different water users are reflected in the management methodology, thus achieving an integrated water management. Under a user-friendly interface a decision support system is formed that allows to control and predict possible actions for water management regarding environmental, agricultural, hydrological and socio-economic aspects. The management process is realized by an integrated information system, performing cyclic processing and analysis of the information and transferring the commands for the next time step.

The selection of several water systems/subsystems is envisaged to be the starting point and the criterion for the development of different problems included in the DSS - OSMOW. These territories (catchments) will be equipped with information devices and monitoring will be organized for the detailed observation of water flow processes, as well as for other processes in the course of experiments with the developed methods.

The introduction of strategic methods for sustainable water resources management in the practice of European countries will be in direct connection with the recently adopted European Water Framework Directive. The introduction of management methods leads to a decrease of water losses, thus affecting favorably economic efficiency. The regulation of the water/air regime in the unsaturated soil zone influences environmental conditions. Vadose zone is functioning

as buffer system for the mitigation of extreme conditions.

Water resources management methods are a technological progress for the population living and working in the corresponding territory. In this way significant social effects are achieved.

## NEW ASPECTS OF WATER MANAGEMENT SYSTEMS

Traditionally water resources management till now has been practically performed for surface water flows. It has been realized mainly by reservoir operation for both ordinary water use and flood prevention. The management task is expressed in an optimized hierarchical distribution of water volumes between different water users for an annual or seasonal time interval. The operative management of an open channel system is connected with substantial problems concerning the water distribution in single nodes of the system. The solution of these problems leads to accurate supply of water volumes to end points of the system.

During the last two decades the problem of integrated water management has reached a certain level of development. Groundwater management for instance is very important because of observed trends of water supply deficiencies and of possibilities to use water for irrigation purposes more economically. Regulating groundwater table contributes to the solution of diverse ecological problems. The general concept of groundwater management is based on controlled drainage according to available water resources in the region. The classical drainage facilities are ditches, open channels, drainage pipes, mole drainage and vertical drainage wells, providing mainly subsurface drainage. Water management systems are set up in various configurations. The drainage facilities should be operated alternately for the management purposes.

Investigations on water management systems useable for subsurface irrigation have been carried out in different countries: Slovakia, Russia, the Netherlands, Poland, Bulgaria, Germany, the USA, the Ukraine and elsewhere. In most cases the attention has been focused on drainage equipment operation and on possibilities of maintaining a certain groundwater level, or on determining the drainage water amount under constant operation conditions. The results of investigations and practical applications clearly confirm the possibility of regulating the groundwater level by means of drainage equipment. These results are only the basis for new research work on a complex of problems depending on requirements for control systems operating under dynamically changing conditions and still meeting the needs of water users.

The computer program system OSMOW comprises new investigations in following research topics:

1. harmonized modeling for strategic planning
2. hydraulic processes connected with the structures for regulation of surface and groundwater flows under changing conditions,
3. regional and inter-regional performance of management systems under the effect of natural conditions and the needs of different competitive water users,
4. influence of groundwater table management on mitigation of extreme environmental conditions (floods and droughts) by water retention or water release in the aquifer and in the vadose zone as well
5. technical-economic, social, hydrological, meteorological, biological, chemical and other problems related to development and operation of systems for water regime management.
6. evaluate the vulnerability of an area regarding disasters.
7. prevent or if it fails manage a catastrophic event
8. coupled analysis of water supply, safety, food supply and environmental impact
9. possible pollutant impact is involved for decision making

The strategy under consideration (item 1) is set into a holistic approach, including definition of tasks and needs (visions, guidelines), water controlling (transformation of strategy into practice), use of selected or developed tools (workmanship) and interpretation of results. The loop is completed through the feedback of results with respect to the defined tasks (Fig. 1).

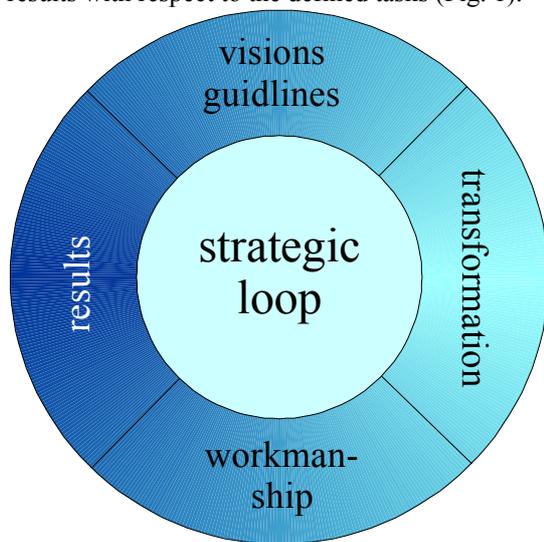


Figure 1: Strategic water resources planning approach

Investigations of item 2 are designated to formulate direct relationships between two types of parameters, describing the processes: "controlling" and "controlled" ones. These relationships are necessary for the quantitative process modeling for both surface and groundwater flows. The experience of the partners of the project consortium shows that it is possible to obtain these relationships by means of numerical modeling. A new simulation and controlling system to achieve the transformation of strategies for the regional water-resources management will be developed in connection

with the investigations in item 3. This system will be based on a hierarchical online closed-loop controlling system with two simulation levels, a detailed complex simulation tool and reduced block-models. The interaction between different processes in soil and groundwater zone is described by coupling different models to one detailed complex simulation package. Non-steady-state simulations can be performed. Substantial error reduction is achieved by introducing fine-mesh discretisation models. Detailed simulation requires the use of high-performance Computation (HPC) technique and parallel computers. The to developed project OSMOW emphasizes on the application of recently developed HPC tools in parallel codes for groundwater flow simulation to the Danube lowland and other groundwater dominated areas. On-line closed-loop control tools must be created for practical field application. These reduced models, the so-called block-models, are calibrated by the complex model. New methods will be used for calibration and sensitive analyses. In this way a hierarchical controlling system will be created. Elements of knowledge based system and mathematical methods for computation with inaccurate measurement values will be also incorporated in the block-models.

Research on groundwater pollution is included in the sphere of the regional management problems considered in this project OSMOW. In particular, the "pure zones" in the region will be defined by solving the so-called "inverse problems", as well as places for discharge and disposal of polluted water will be determined.

Investigations in test fields related to item 4 would contribute to the proof or adaptation of theoretical concepts. Data for all observed water, soil and plant parameters with respect to the regulated groundwater regime will be analyzed.

There are no data available for holistic investigations related to item 5 on technical-economic and social problems of water systems with controlled operation. Such an analysis will be performed on basis of quantitative characteristics both existing and new planned water systems.

OSMOW will advance the present situation by a holistic methodology for water resource planning under dynamic changing conditions.

Results of investigations envisaged in the DSS - OSMOW would represent a sound basis for further development and practical application of methods for water-resources management.

As a final conclusion the performance of these investigations will represent a stage in the acknowledgement of the general trend for rational use of water resources.

The OSMOW includes the development of a new method of modeling the diffuse pollution. The basis of the new method is the solution of "inverse problems". In this way the possibility is created to determine (manage) the flow path of disposed pollutant without aggravation of water quality in the rest part of the catchment.

## THE CONCEPT OF OSMOW

The work planned is following the red line conception phase → implementation → application.

The conception phase comprises the harmonization of the modeling concepts within the consortium to achieve comparable results and data process identification for extreme conditions. All partners need to be actively involved in this part, because it is essential for the success of the proposal to establish a common starting point and a mutual understanding of problems each partner contributes to the research investigations. The consortium is formed by partners comprising knowledge in water resources management, groundwater flow simulations, drainage and irrigation, unsaturated flow, data base establishment as well as socio-economic and environmental issues. In this way most synergy effects of the different expertises may be gained for the project. The research work will be performed with a strong link to endusers. Members of the consortium include research institutes and water management institutions.

The implementation is a key action of the project. Hence first the selection, adaptation and when needed the development of new mathematical/numerical concepts will be performed. The participating partners have experiences with all kind of models related to all parts of the hydrological cycle. With these models a pool of mathematical concepts is formed and cross models are established. The pool of models includes in particular:

- Regional models for description of surface water flow:  
HBV and HBV-light (rainfall-runoff model), NLC, NONLIN, MODI (river water quantity and quality)
- Models for water balance in cross-sections of a main river
- Regional models for description of groundwater flows:  
MODFLOW, MOC, THEIS
- Local models for description of drainage system operation:  
DRAINMOD (simulation of shallow groundwater table), SEEP-CANAL (operation of an open drainage channel), SEEP-DRAIN (operation of pipe drainage), SEEP-WELL (drainage wells)
- Solute transport models:  
MODFLOW-MT3D, SWMS-3D, RPMOD (3D-model for reverse filtration problems), HYDRUS, COMBESICK, GLOBAL (1D-, 2D- and 3D-models for unsaturated flow in rigid soils), MACRO (2D- water and solute transport in macro porous soils), FLOCR (water flow in shrinking and swelling soils)
- Models for determining the relationships between “controlling” and “controlled” parameters
- Algorithms for inverse problems:  
MINPACK (LEVENBERG-MARQUARDT-Code), NL2SOL (Secant method for nonlinear least squares)

- Block models for simplification of the detailed models
- Models for water-economy balance in a catchment

The pool is not defined to be final during this phase; it may be extended with other models whenever needed.

The mathematical/numerical concepts are made useable by developing simulation models. The simulation tools are structured in two levels. Out of a detailed complex simulation model so-called block models are generated (Fig. 2). These models are related to a given task.

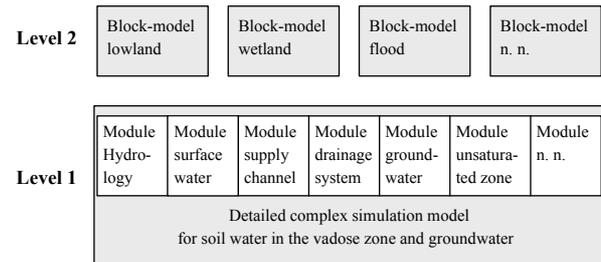


Figure 2: Hierarchical controlling system

Block models are:

- simplifications of a detailed model. There are different methods to achieve simplifications, e.g. condition viewer or transfer functions
- designed for special regions and users
- designed for different hydrological conditions, that means for a region more than one block model is required, e.g. for spring or summer conditions, flood or low discharge
- calibrated by the complex model
- integrating a part of knowledge based rules. In this case a lot of expert experience exists

The block-model level is a simplification, but enables the user to overcome the disadvantage that not all influences of the groundwater level can be described by mathematical equations. Block-models run not only on mainframes, but also on control computers, e.g. on a PC. Another aspect is inaccurate measurement values. In the state of art simulation tools take for granted accurate values, which are not available for practical working conditions. The innovative idea is the combination of mathematics of inaccurate values (e.g. Fuzzy set manipulation) with the block-models to achieve knowledge based Fuzzy controlled system.

The following figure shows the online closed-loop control system for a part of the regional water balance: surface water – vadose zone - groundwater.

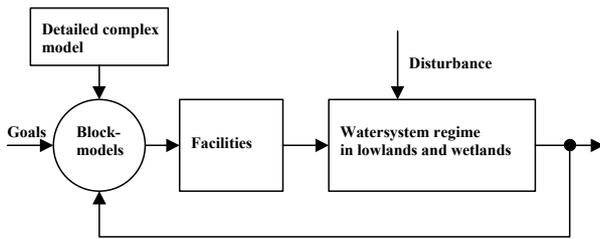


Figure 3: Loop for controlling components of water balance:  
surface water, vadose zone and groundwater

The block model is online coupled with the control-loop comparing results with actual values. The goal function will be created in the computer or put in by the operator. Actual measurement values are made available online or with datalogger systems. The block-model calculates the input of regulation units. The transmission from the computer to regulation units is online. In this way a feedback between actual values and the regulation unit will be guaranteed. For the improvement of controlling effects multi-stage control loops will be used (Fig. 4). Such control loops take into consideration e.g. disturbance values (disturbance feed forward) and early warning condition values (controller feedback).

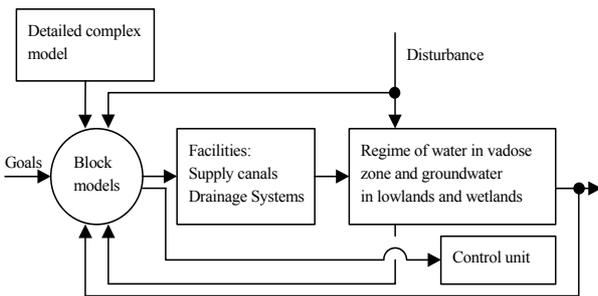


Figure 4: Multi-stage control system

The best mathematical concepts will not be used if they are not set into a user-friendly simulation environment. Therefore it is intended to embed the simulation tools into a graphical user-interface.

The application phase starts with the establishment of benchmarks. Test regions are selected to cover most foreseen extreme conditions in Europe. Simulations in test regions serve to proof the applicability of the proposed simulation tool and to perform a feasibility study with the established benchmarks.

A diversity of European test sites, covering the extreme conditions as previously described before, including the East Slovakia Lowland

- Estimation of the characteristics of:
  - Water-level regime in a river network and channels,
  - Water-level regime of groundwater,
  - Water-content regime of the unsaturated zone of soil

and their both spatial and temporal interaction.

- Estimation of the components of water balance in the system of atmosphere –plant canopy – unsaturated zone of soil - groundwater
- Comparison of the characteristics monitored in the test region with the results from the harmonised model

test field in Poland

- the catchment of the Gasawka river with drainage-subirrigation systems as main users of water, with other users as fish ponds, sugar plants, small towns and villages, environment (biological flows), wetlands, recreation and tourism
- the polder area in the north Poland with flood risk and mainly agriculture activity

test field in Bulgaria

- With subirrigation as characteristic component of water balance
- description of Iskar river basin up to Novi Iskar town, detail description of water resources system structure
- methodology for short-term operation of water resources system

Upper Tisza Region, Hungary

- Especially with respect to social, economic and environmental effects of integrated water resource management

Marchfeld region in Lower Austria

- This region is supplied with water from Danube for irrigation purposes. The main channel serves also as recipient for some small wastewater treatment plants.

should be representative for Europe.

After the establishment of the harmonized modeling concepts the following workpackages are embedded into a socio-economic context. Environmental issues, when not already included in water related topics, are attributed accordingly.

## INTERNATIONAL ASPECTS

The European dimension of dynamic changes of the water cycle is reflected in the European Water Framework Directive. Global change effects are investigated in various research programs. The changing of climate is not only a European problem, but also more a worldwide challenge. OSMOW will take advantage of these results and is aimed to develop strategies to react to extreme conditions, like floods and water shortages. Water as a transboundary resource needs common attention of neighboring countries or of those who share river catchments. A harmonized modeling concept for strategic water resource planning will contribute to improvements of guidelines, standardization and regulations. It also ensures that the guidelines are applied accordingly.

Declining availability of water is a profound problem, which is not only a result of reduced water supply, but also exaggerated by the continuing increase of water use for agriculture, industry, municipal demand, nature conservation, etc. These problems become evermore significant in many regions of Europe and worldwide, especially in CEE countries (Central and Eastern Europe).

On the other side of the spectrum extreme conditions related to an excess of water due to an increase of annual return periods are observed in the western part of Europe. Most practices are related to surface water. An important aspect of the proposed research project is to investigate the controlling possibilities of groundwater and the storage capacities of the unsaturated zone.

Due to human activities water quality (pollution) became a key topic for water provision and environmental conservation.

The problems stated above (floods - draughts - pollution) occur in different regions and under different hydrological and environmental conditions, but to cope with the effects for all an efficient management strategy is needed. Hence exchange of information and expertise between the regions in Europe has to be enforced. In particular this will emphasize the role of integrated water measurements and efficient water use strategies.

- By developing a harmonized modeling tool results are comparable and synergy effects are easily passed on.
- The establishment of benchmarks should contribute to unbiased analyses of the performance of the proposed strategies.
- The proposed strategies are set into operation by simulation for test regions, which are representative for Europe.

The project will utilize existing experience whenever available and will establish a knowledge resource that will be continuing beneficial after the present research program is completed. For the dissemination of results CD-ROM and other electronic means are utilized. The information collected in the created knowledge base will be accessible through Internet.

The scale of the project impact is potentially very large. Regional development in water scarce areas is highly dependent on available water resources. The efficient allocation of water is important as it maximizes the potential returns on water use economically and environmentally.

The research partners also see particular promise for the commercial exploitation of European expertise around the World. Solving the problems of rational water use in Europe will enable European organizations to utilize or adapt the techniques for other parts of the world – particularly developing countries where water-shortage is steadily increasing and represents a profound threat to the stability of the local environment and society.

## SOCIAL ASPECTS

The natural, economic and social conditions of most countries in Europe and worldwide confront the society with the challenge to solve the problem of sustainable water resources management under dynamic changing conditions. These changing conditions may be a result of global change effects or due to human impacts. Strategic water resource planning utilized by operational methods contributes to mitigate negative effects of droughts (water deficiencies) or an excess of water (floods) in other periods of the year. In general, according to quantity and quality, water is considered to be a deficient natural product.

Water resources management in modern practice represents already an instrument for solving water economic and ecological problems in a rational way. Management tasks are related to surface and groundwater. Emphasis has to be paid between both resources in a catchment. An important role plays the movement of water in the unsaturated zone. The resource water is under continuous pressure of pollutants, due to various points and diffuses sources.

Sufficient amount of water of good quality is essential for the quality of life and health.

This will only be achieved if the resource water is treated as an European concern.

In a broader sense “operational management” includes all activities leading to a sustainable water regime. More specific, operational water management has to deal with predictions of water resources use in a catchment, monitoring and data processing, analyses of water resources changes and identification of response actions.

This has to be seen in a socio/economic context and interactions with water policy statements are foreseen.

The results of OSMOW will directly effect European and worldwide international social objectives by:

- maintaining rural livelihoods. By providing mentoring advice to water and agricultural organizations, the project output will help to improve food and water availability; preserve/enhance the environment and to provide employment. Improvement of Quality of Life and Health and Safety can be expected. It has been recognized in the Fifth Environmental Action Program that the abandonment of farmlands can have negative effects on the environment.
- expansion of EU by promoting and developing links between leading organizations in the existing EU that are interested in efficient water resource management (especially in regions with water shortage) and similar organizations in Central and Eastern Europe. The network will encourage better intra-European co-operation. This will ensure that relevant results and experiences within the

community can be utilized - hopefully to the advantage of the pre-accession states.

- improvements directly developed in the CEE, which will be utilizable in many regions of the EU. This is a step towards equality of information access and opportunity in all countries of EU. The compilation of all the information from the various sub-topic networks into one integrated system will provide specialized knowledge to be readily accessible by professional colleagues in partner countries, thereby adding to the EC's net overall knowledge.
- Improving agricultural production and maintaining the agricultural nature of regions in Europe. Better availability of technical information across the range of irrigation and drainage topics will help participants to better contribute to European efforts to improve yields and to limit effect of external impositions of drought and associated difficulties.

The project results will enable researchers and practitioners across Europe to share experience and skills, and then apply/adapt the benefits to their own respective countries or regions.

From an indirect point of view, the wasteful use of water has financial and economic consequences including the loss of rural livelihoods. In water scarce regions farm incomes are often very marginal and livelihoods are delicately balanced.

## CONCLUSION

A deficit of water resources, unfavorable distributions of water in time and space, increasing water use, polluted water resources, variability and instability of climate and change of socio-economic activities demand rational ways of water management. The main objective is a methodology for effective operational water management under normal and extreme conditions of the water cycle. The research supplies solutions and tools to: evaluate the danger and vulnerability of an area; prevent damage; handle water supply and ensure quantity and quality of water resources; consider hydrological, meteorological, biological, biochemical and ecological influences; generate sufficient data; handle an interactive data base for interpretation and prognosis. A DSS requires a coupled analysis of all kinds of data to describe the long-term scenario of global and local changes. DSS improves co-operation and communication of involved parties in order to take action for controlling and steering.

Main milestones and expected results are: harmonized modeling concepts and data base for simulation of extreme conditions, simulation models including a graphical user interface, benchmarks to ensure comparable results and to provide a check of simulation performance, socio-economic and environmental

restrictions. The result is a DSS as required by the new General European Guidelines regarding the controlling and steering of the water cycle to provide help to evaluate possible impact of actions.

## REFERENCES

- Graeber, P.-W. 1999. "Modelling and Simulation of Ground Water Processes." *Processes Modelling*, Springer-Verlag, Berlin, 446-457
- Graeber, P.-W.; Puehl, St. 1999 "Rule-based Spatial Query and Production as Part of the WHPA." *Processes Modelling*, Springer-Verlag, Berlin, 545-558
- Schaefer, E.; Loiskandl, 1996. "Application of Simulation in the Field of Water Resources and Water Treatment, Modelling and Simulation." *ESM96*: 1053-1055
- Loiskandl, W. 1999 "Modelling approaches for water flow and salute transport through a porous media, *Processes Modelling*, Springer-Verlag, Berlin, 457 - 470

## AUTHOR BIOGRAPHIES



Peter-Wolfgang Graeber was born in Leipzig, Germany and went to the Dresden University of Technology, where he studied electronic and control engineering and obtained his degree in 1970. He worked for the research institute of water management before moving in 1980 to the Dresden University of Technology where he is now leading the research group of modeling and simulation in the water engineering. Since 1996 he works as professor in the Faculty of Forest-, Geo and Hydro Sciences in the field of system analysis and computer science in the environmental background. His email address is: [graeber@rzs.urz.tu-dresden.de](mailto:graeber@rzs.urz.tu-dresden.de) and his Web-page can be found at <http://www.tu-dresden.de/fghh/aa/graeber>.

# Texture Classification applied on aerial imagery in Forestry

Dietmar P.F. Moeller, Christian Koerber  
Department Computer Science, AB TIS  
University of Hamburg  
Vogt-Koelln-Straße 30, 22527 Hamburg,  
Germany  
E-Mail: dietmar.moeller@informatik.uni-hamburg.de

Christoph Kaetsch  
Department of Forest and Wood Science  
University of Stellenbosch  
Privaatsak X1, 7602 Matieland,  
South Africa

## KEYWORDS

multi-channel Image Analysis, Digital Signal Processing, Texture Classification, Object Extraction, Clustering

## ABSTRACT

Feature extraction from aerial images is an important research topic with a wide area of applications, like traffic or agriculture monitoring, natural disaster early warning system, etc. Because, to deal successfully with, information gained by remote sensing is by several factors much more cost effective compared to manually accomplished measurements. Certain Information is actually only available by remote sensing since, as the area under investigation is as huge, that any other means would be infeasible. The extraction of certain objects from aerial images has proven to be a very difficult problem especially if the investigated objects do not have sharply bounded lines. This is common the regular case when dense forests are concerned during image analysis. Here, trees usually occlude each other and are hard to differentiate from epiginous vegetation, that make geometrical approaches of object identification, as well as direct representations, hard to apply. On the other hand, counting's performed in the frequency domain offer the advantage of transformation invariance and suffer lesser from diffuse object boundaries. The determination of a clear signature is difficult, if the objects in question are quite similar, though. Hence the paper suggests an approach derived from texture classification that achieves better results in the forestry.

## INTRODUCTION

Monitoring of forests for several purposes has proven itself as being a demanding and costly task. Gathering data on this diverse natural resource providing information for sustainable forest management means identifying single trees as well as their most important properties like species, age, health-status, height, crown and trunk diameter. In a normal forest inventory covering 8000 ha to 10.000 ha the number of trees to be measured easily exceeds 10.000 or even 12.000. To minimize costs, current forest inventory systems are generally based on ground measurements taken on sample plots distributed regularly over the whole forest. In addition, terrestrial measurements are supported by using aerial

photographs. The synoptic overview provided by the images can help to stratify the forest and allows to optimize the terrestrial work. In fact, remote sensing is used for nearly 100 years (Hildebrandt 1996) in the forestry domain resulting in a widely used set of tools for analogue image analysis in forest inventory.

Despite all efforts towards optimized forest inventory concepts measurements in forests remain costly. Expensive equipment required for image interpretation and time consuming manual Photogrammetry have not met the high expectations.

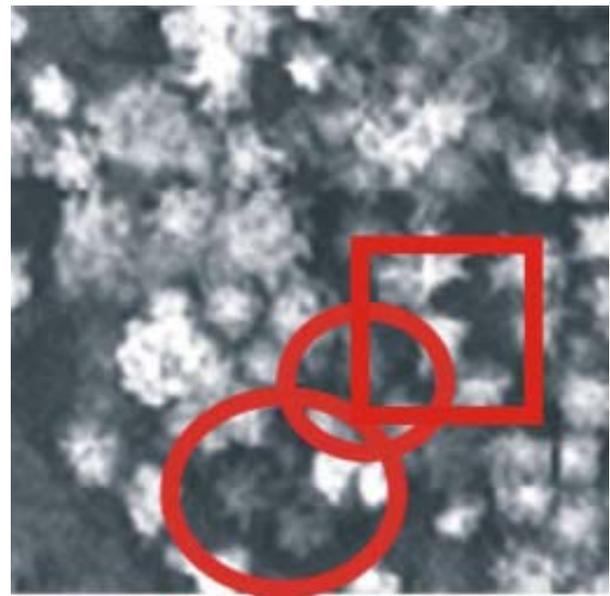


Figure 1. Aerial<sup>1</sup> photograph showing an uneven-aged mixed forest stand formed by Norwegian Spruce, White Oak and European Beech. Spruces with low intensity are marked by circles, the square corresponds to Fig. 3.

Growing and improved availability of digital images may change this situation dramatically. While digital images are easy to handle and allow sophisticated image processing, reliable methods capable of addressing a

---

<sup>1</sup> IRC chromatic aerial image courtesy of Institut für Forsteinrichtung und Ertragskunde, University of Göttingen. Scale approximately 1:6000, scanned with a photogrammetric scanner, resolution 12  $\mu$ m, GSD approximately 8 cm.

broader range of stock enumeration issues may pave the road towards automatic forest inventory systems.

Since the early 1990's several efforts to extract data on trees and forest stands from digital images in a widely automated manner have been published (a.o. Pinz, 1988, 1992; POLLOCK, 1994, 1996, STRAUB & HEIPKE 2001, KÄTSCH 2002). Despite very promising results presented from different authors none of the methods proposed so far can be seen as an universal tool ready to meet all challenges related to different stand structures, tree species or age-classes found in forests around the globe.

This paper presents first results of study carried out in uneven aged Central European forest stands using multi-channel texture classification, which in combination with existing methods may lead to more general approaches of automated stock taking in forestry.

Fig. 1 shows a cutout of a grayscale version of a chromatic aerial image, displaying the Hills area in south western Lower Saxony, Germany. The (1) mixed forest shown is characterized by a tree population of (2) very varying age (40 to 120 years), consisting mainly of oaks, beeches and spruces. The (3) differing height structure of the trees leading to shadowing effects is clearly visible. All three mentioned factors result in an increased complexity regarding automated image analysis. In this paper we restrict ourselves to issues (1) and (2), which are, among others, characterized by the difference between high intensity broad-leaved trees and low intensity conifers, the latter marked by a circle in Fig. 1, a characteristic which classical image processing algorithms tend to miss, as exemplified in Fig. 2, showing the results of an edge detection filter using a Sobel kernel as in (Paulus and Hornegger 2003).

## THE MULTI-CHANNEL APPROACH

This paper suggests an approach, based on the two pass filter algorithm, introduced by (Zell 1994), which itself was derived from (Hatzigeorgiou 1993). The first pass comprises a windowed multi-band signature analysis, the second pass consists of an artificial neural network classifier. Our contribution introduces a multi-resolution<sup>2</sup> representation of the first pass and replaces the neural network classifier by *k*-means clustering, supersede the automatic algorithm by a semi-automatically one, i.e. one, that needs currently at least some manual interaction, and in such manner gaining increased accuracy for an extremely narrow feature set.

<sup>2</sup> (Zell 1994) refers to the multi-band approach as multi-resolution which is appropriate in the sense that lower frequencies contribute more to the signal than higher frequencies and that thus a low pass filtered image has a different resolution than a high pass filtered one. We refer to the term multi-resolution in the sense of different window sizes.

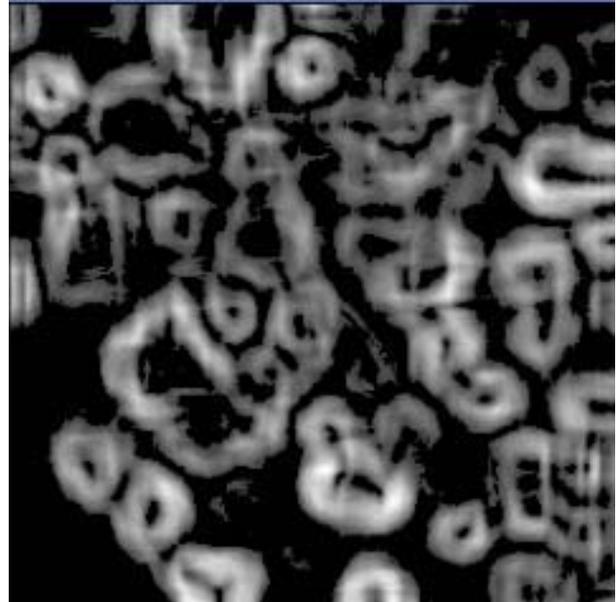


Figure 2. Edge detection applied to Fig. 1 after low pass filtering. The conifers, marked by circles in Fig. 1, tend to vanish

The **first pass** consists of five steps:

1. Window extraction  
A square window of size  $N \times N$  is extracted for each pixel of the image, centred at the pixel, where  $w$  is an odd number, see Fig. 3 (left). Windows not fitting completely into the image are disregarded so that, if the image has  $M \times M$  pixel, only the inner  $(M-N+1) \times (M-N+1)$  pixel can be considered. Typical window sizes which had been proven to be the most effective are rather smaller than those reported by (Zell 1996) and range from 7 to 25 pixel. The example shown in Fig. 3 has a size of  $64 \times 64$  pixel only for demonstration purposes.
2. Window smoothing  
The extracted sub image becomes smoothed to zero at the edges during the next step in order to avoid undesired edge effects. Smoothing is performed by multiplying the extraction window pixel wise with a filter that shows good stop-band attenuation if transformed to frequency space, as shown in Fig.3 (middle and right).

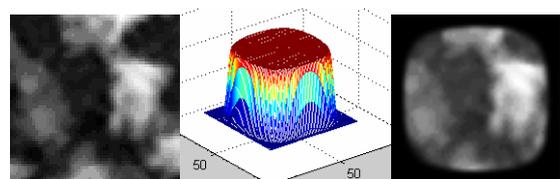


Figure 3. Extraction window corresponding to the square region of Fig. 1, multiplication filter, resulting extraction window (respective extracted sub-image, from left to right)

### 3. 2D-FFT

Window smoothing is followed by calculating the magnitude of the two-dimensional discrete Fourier transform<sup>3</sup>,

$$X[k_1, k_2] = \frac{1}{N^2} \sum_{n_1=0}^{N-1} \sum_{n_2=0}^{N-1} x[n_1, n_2] e^{-i2\pi \left( \frac{k_1 n_1 + k_2 n_2}{N} \right)}$$

whose locality in pixel space obviously depends on the size of the extraction window. Too large windows result in failing to detect texture boundaries, while smaller windows cannot detect larger patterns. Therefore the window size depends on the type of application and on the scale of the aerial photographs or satellite images. Images, window sizes between 7 and 25 rendered, show the best results. The extracted sub-image had to be zero-padded to achieve at least a magnitude spectrum of  $128 \times 128$  pixel in order to avoid numerical complications in subsequent steps of the algorithm.

### 4. Multi-band pass filtering

Step 4 consists of 8 successive band pass filters completely partitioning the spectrum. The radius of the succeeding filter is  $\sqrt{2}$  times the radius of its predecessor, which is the reason for the greater size of the spectrum than the original image's size. Hence the first filter, the low pass, must thus be relatively small, and would be smaller than a single pixel if the extracted image wouldn't be zero padded. The adopted filter was a perfect band pass, obtained by setting all values of the spectrum to zero, which don't fall into the circle of the respective band. Since no inverse transform is intended no disadvantages are expected from avoiding a more sophisticated filter as e.g. the gradients of a Gauss function.

### 5. Signature generation

Finally, the signature of the pixel which lies at the centre of the extraction window, is generated by computing the arithmetic mean value per band pass. This results in eight scalar values, as eight band passes are used, representing an eight-dimensional vector, that is referred to as signature of the pixel.

The **second pass** completely deviates from (Zell 1996) and consists of three steps:

#### 1. $k$ -means clustering

$k$ -means clustering is a statistical method of vector quantization where a "codebook"  $K$  of  $k$  vectors is build up by the algorithm with the objective to heuristically determine its  $k$  vectors so, that the average minimum Euclidean distance for the  $m$  vectors of a set  $M$  (here the signatures) to one of the  $k$  codebook

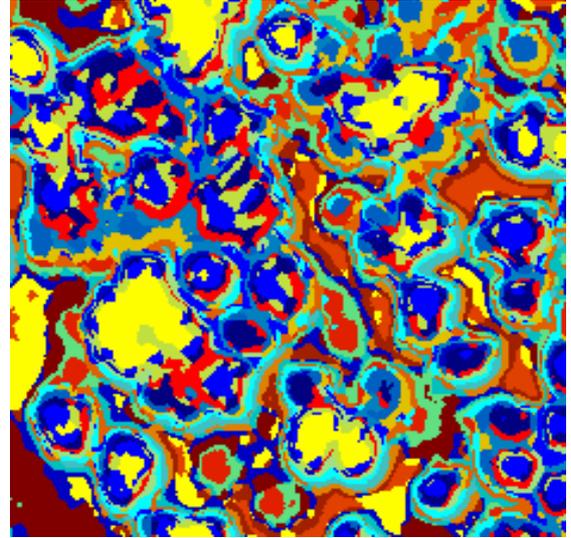


Figure 4. Merging the signatures into  $k$  cluster obtained from a 16-digit signature

vectors becomes minimized (with  $k < m$ ). Applying this kind of clustering results in good outcomes. Fig. 4. gives an indication though it refers already to the improved version of the algorithm, described in the next section. It has to be noticed that the clustering was not able to distinct noise from low intensity conifers remarkably better than other methods. It has also been observed that increasing the number of clusters didn't increase the quality if the distinction, an indication that the band which separates the spectrum of the conifers from noise, must be vary narrow. An improved version of clustering, which lead to significantly better results, is described in the following section.

#### 2. Merging clusters to classes

Once the clusters are obtained, they need to be merged down to a more concise number of 3 or 4 clusters which are referred to as classes, see Fig. 5. This step currently requires manual user interaction and this is the sense where some kind of "intelligence" needs to be applied. Since this is limited to merging about 32 color regions to larger ones, we expect this to be achievable in about 5 to 10 minutes, based on our observations on manually merging clusters within an imaging application and assuming an appropriate user interface is provided, which we consider tolerable. In the conclusion a thorough discussion of this subject will be given.

#### 3. Morphological operations

Morphological operations are not necessarily an integral part of the algorithm but are required by the application in order to utilize the gained information. These operations include dilation and erosion in order to get a better object separation without losing too much object area, binarization,

<sup>3</sup> Following the notation for the complex DFT of (Smith 1999) and using the FFT for the implementation

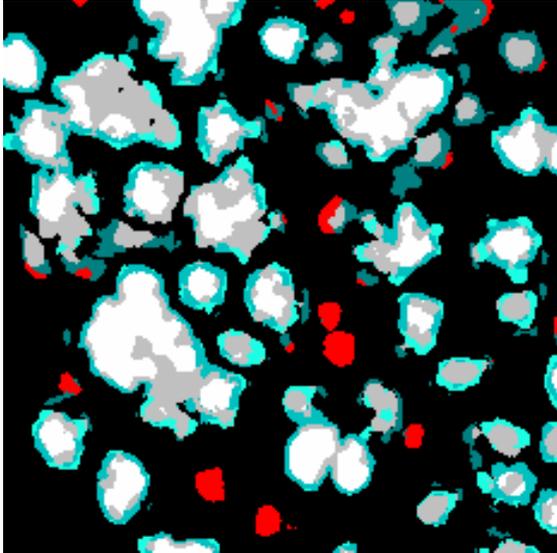


Figure 5. Merging clusters (as in Fig. 6) into classes, red regions refer to regions identified as conifers

required for object identification and ellipse embedding (unaccounted by us yet), used for further formalization and subsequent automated processing. Fig. 6 shows the identified objects after dilation, erosion, binarization and contouring overlaid on the original image.

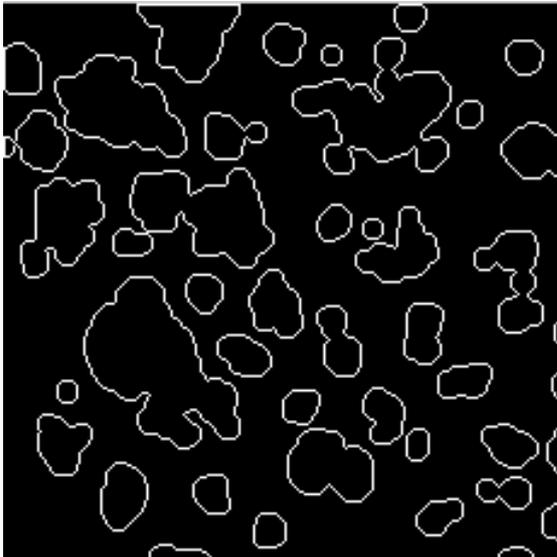


Figure 6. Identified objects after dilation, erosion, binarization and contouring

## IMPROVEMENTS

The originally obtained results showed similar disadvantages as filter-based algorithms. The algorithm had difficulties to distinguish low-intensity conifers from noisy, non-vegetable parts of the signal as it can be seen in the lower left corner of the investigation area in Fig. 1, that represents a forest track. This could not be remedied by increasing the number of clusters, which had only led to the same finer subdivision in both areas, the

one containing the conifers and the one containing the noise.

One obvious solution was to use information gathered by varying image sizes, especially the tendency of low intensity objects to vanish with increasing window size, because noise remains noise. We applied two different algorithms, both utilized a 7 pixel wide and a 25 pixel wide extraction window. The first algorithm combined two clustered images by generating  $k1 * k2$  (basically by appending  $k2$  to  $k1$ ) new clusters from the  $k1$  respective  $k2$  clusters, computed by the method suggested above. This not only generates numerous meaningless clusters respectively needlessly separated well bounded clusters and thus hampered the successive classification step, it also introduced unwanted artefacts, though it indeed result in a better object distinction. Much better results are gained by appending one signature to the other, thus gaining a 16-digit<sup>4</sup> instead of a 8-digit signature vector, and performing the k-means clustering in a 16-dimensional instead of a 8-dimensional vector space, see Figs. 4 to 6.

## CONCLUSION AND PERSPECTIVE

- Our work so far emphasized the detection of low intensity objects (conifers) which obviously had the greatest prospect of success for our application. More work is required with respect to object separation, see Fig. 6. We expect to gain better results in this area either with improved morphological operations (i.e. testing with successive erosion operations whether objects are separable) or with geometrical operations (ellipse embedding).
- We anticipate further improvements through utilizing other spectra, especially the infrared channel. Hyper-spectral Images which had been introduced to remote sensing in the context of forestry may give this approach a hole new direction as the up to 220 channels could be utilized for direct clustering.
- The elimination of the semiautomatic classification step could probably be reached by e.g. artificial neural networks (perceptron, similar to character recognition) But since the requirements for manual interaction are already condensed to a very small effort, which can easily be further streamlined by providing a simple interface, we would rather improve object separation which will improve the overall usefulness of the algorithm to a much higher degree.
- Some investigations may be appropriate whether at least partial results of this algorithm could be ported to operations in the spatial domain. The solution to work with different window sizes for the 2D-FFT corresponds to working with different

<sup>4</sup> with each digit being a floating point number

kernel sizes for convolution (low pass filtering, edge detection).

## REFERENCES

- Hatzigeorgiou, A. "Texturerkennung mit neuronalen Netzen" In Zell, A (Ed.): *Workshop SNNS-93: Simulation neuronaler Netze mit SNNS, Universität Stuttgart, Fakultät Informatik, Bericht Nr. 10/93*. pp. 50-58, 1993
- Hildebrandt, G. "Fernerkundung und Luftbildmessung für Forstwirtschaft, Landschaftsökologie und Vegetationskartierung. Wichmann Verlag. 1996
- Kätsch, Chr. Towards an automatic forest inventory system based on remotely sensed data and digital image processing. Proceedings of the 29<sup>th</sup> ISRE-Conference, Argentina 2002
- Paulus, D.W.R., Hornegger, J. "Applied Pattern Recognition", Friedr. Vieweg & Sohn Verlagsgesellschaft mbH, Braunschweig/Wiebaden, pp. 146-156, 4<sup>th</sup> Ed. 2003
- Zell, A. "Simulation Neuronaler Netze". Addison-Wesley, Bonn; Paris; Reading, Mass., pp. 519-531, 1994
- Pollock, Richard J., The Automatic recognition of Individual trees in Aerial Images of Forests Based on a Synthetic Tree Crown Image Model, Vancouver, Canada, June 1996: The University of British Columbia, 1996.
- Pinz, A.: Ein Bildverstehendes Expertensystem zur Erkennung von Bäumen auf Farb-Infrarot-Luftbildern. PhD thesis, Technische Universität Wien, 1988
- Smith, S.W., "The Scientist and Engineers's Guide to Digital Signal Processing", California Technical Publishing, San Diego, California, p. 570, 2<sup>nd</sup>. Ed. 1999
- Straub, Bernd-M. and Heipke, , "Automatic Extraction of Trees for 3D-City Models from Images and Height Data," (Automatic Extraction of Man-Made Objects from Aerial and Space Images (III), Band 3 hg. von A. G. L. v. G. E. Baltsavias) A.A.Balkema Publishers: Lisse/Abingdon/Exton(PA)/Tokio, 2001, 267-277.

## AUTHORS



**DIETMAR P. F. MÖLLER** was born in Preetz, Germany. He enrolled at the Universities of Lübeck, Bremen Mainz and Bonn, where he studied Electrical Engineering and Human Medicine. He obtained his doctoral degree in 1980. From 1985 to 1991 Dr. Möller lead the anaesthesia division of Dräger AG in Lübeck, Germany. 1991 he has been elected as Full Professor for Computer Engineering (TI) at the Technical University of Clausthal. In 1998 he has been elected as Full Professor for Computer Engineering Systems (AB TIS) at the

University of Hamburg where he holds the chair for Computer Engineering. Since 1998 he also is Adjunct Professor at the California State University Chico. His E-Mail address is [dmoeller@informatik.uni-hamburg.de](mailto:dmoeller@informatik.uni-hamburg.de) and his Web-page can be found at <http://www.informatik.uni-hamburg.de/TIS/>



**CHRISTIAN KÖRBER** was born in Hamburg, Germany. He enrolled at the University of Hamburg, where he studied Computer Science and obtained his diploma degree in 2002. He joined the Computer Engineering Department (AB TIS) as a research assistant where he lead research work on Geo Information Systems, <http://www.verve-gis.com> with specific emphasis on computer graphics and geometric modeling. E-mail address/web page is [koerber@informatik.uni-hamburg.de](mailto:koerber@informatik.uni-hamburg.de); <http://www.informatik.uni-hamburg.de/TIS/>



**CHRISTOPH KÄTSCH** was born in Göttingen, Germany. He studied Mechanical Engineering at the Technical University of Clausthal and Forestry Sciences at the University of Göttingen. He finished his Phd in 1990 and received the *venia legendi* (Dr. habil.) in 1997. From 1982 to 1987 he was a Forest Officer at the State Forest Service of Lower Saxony where he became head of the department of Informatics and Accounting in 1988. 1992 he was appointed as Professor for Geoinformatics & Remote Sensing at the University for applied Science & Art in Hildesheim. Since 2004 he is holding a chair on Geoinformatics & Remote Sensing at the Faculty of Agriculture and Forestry Sciences, Stellenbosch University, South Africa. Email: [christop@sun.ac.za](mailto:christop@sun.ac.za), Web: <http://www.academic.sun.ac.za/forestry>

# A PHILOSOPHY OF MODELLING AND SIMULATION AS APPLIED TO DYNAMIC SYSTEMS

Richard N. Zobel  
Department of Computer Engineering  
Faculty of Engineering  
Prince of Songkla University, Phuket Campus  
80 Vichit Songkram Road, Kathu  
Phuket, Thailand 83210  
Email: [rzobel@phuket.psu.ac.th](mailto:rzobel@phuket.psu.ac.th)  
[r.zobel@ntlworld.com](mailto:r.zobel@ntlworld.com)

## KEYWORDS

Modelling, level of modelling, simulation, simulation studies, parametric and statistical studies, V,V & C, testing, results, analysis, reports

## ABSTRACT

Undertaking a simulation study, except when trivial, is always a complex process. It is viewed in very different ways by the various parties involved. This paper looks at the processes, stages and sequences from several different perspectives, in order to expose the philosophy and arguments involved in the decision making leading to the successful and profitable use of simulation or to its rejection.

## INTRODUCTION

Simulation can be a very powerful tool for system design support and validation, for providing performance data, for investigating operational regions beyond the anticipated safety limits, and provide a valuable aid for training. Although many complex systems may be amenable to analytical techniques, simulation is required to ascertain characteristics and performance for systems represented by many equations, which are also non-linear and time-varying.

In the following discussion the author uses his own experience and knowledge of simulation gained over the past 45 years in industry and academia to expose some of the problems, dilemmas, views and solutions for a variety of different simulated systems.

## AIMS AND OBJECTIVES OF A SIMULATION STUDY

Simulations and simulation studies should be developed in a similar way to a piece of software or a product, that is, through a sequence of cycles of requirements, specification and outline design, until the aims and objectives are reasonably clear. At this stage it is essential to involve the management team, project engineers and others, in order to provide a properly argued case for the need of such a simulation study on

a need and cost/benefit basis (Korn and Korn 2000, Melsa and D. Cohn 1978).

Costs should include items for modelling, development of simulation including verification and validation of the models and simulation, modelling and simulation tools, computers and software, and suitably qualified staff, all associated with time estimates for each task. These and the benefits must be achievable and clear to the management. The benefits should include the verification and validation of product or service design decisions at all stages. Where persons are included in the loop, appropriate human factors must be considered. Where there is hardware-in-the-loop to be considered and/or humans, the real-time and interface aspects must be included in the evaluation. For the situation where there are both person(s) and hardware-in-the-loop all of these factors must be considered.

The principles of concurrent engineering should be used in the product design and manufacturing processes, and also used for parallel modelling and simulation developments, see figure 1 (Habibi and Zobel 1996).

The level of modelling employed overall and for each of the sub models should reflect the aims and objectives of the study in relation to the requirements for sufficient accuracy, resolution and validity. Higher levels cost money and time and should be seriously questioned by the management. Of course, re-use of existing models is highly desirable, even if they are more complex than required for the new study. However, this must be subject to adequate documentation including a complete specification, interface definitions, evidence of verification and validation, and a complete list and specification of modelling assumptions and limitations (Gass 1978, Han 2000). A simulation model library is desirable, but this must be used with care (Lee and Zobel 1997).

After sufficient iterations, the final aims and objectives must be completely specified, documented and agreed by all interested parties. These should include the customer, whether internal or external to the organisation proposing the simulation study.

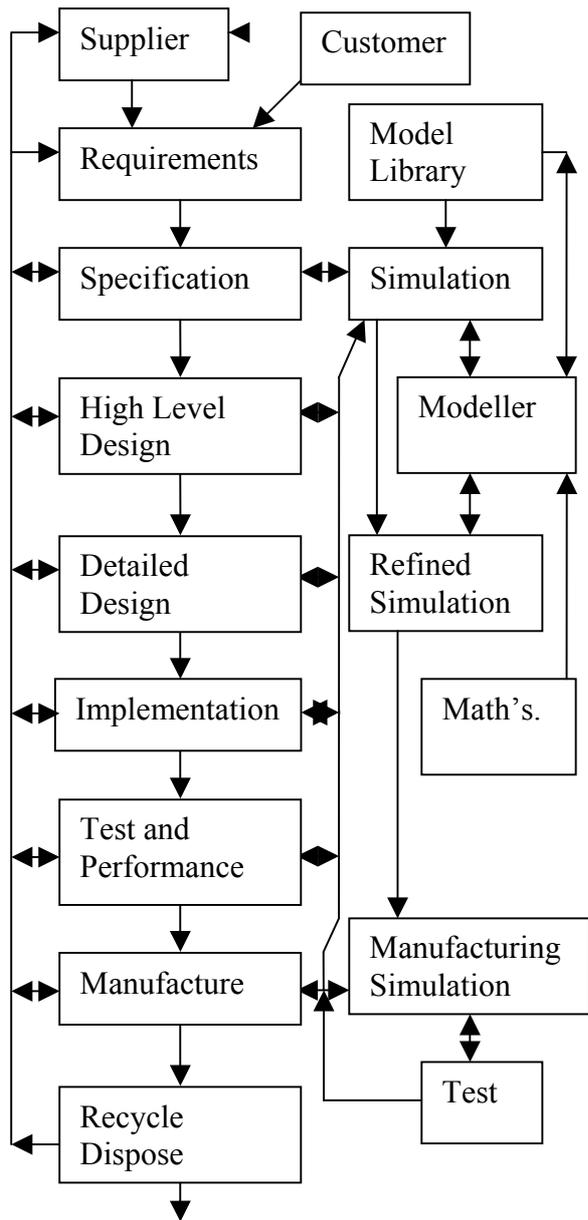


Figure 1: Parallel Concurrent Engineering and Simulation for Manufacturing

**STRUCTURE OF A SIMULATION STUDY**

There are many aspects, facets and views of a simulation study which must be considered, implemented and documented, before such a study can be justified and carried out.

**System Block Diagram**

The first requirement is a for a top level system block diagram, with details of each block (figure 2). Lower level block diagrams are also needed, subject to the need for restricting the level of detail to those required by the aims and objectives of the study, and other technical details (figure 3). The latter may involve the

need for the real-time solution to a partial differential equation, for which the available hardware does not have the performance. Alternative methods may have to be sought. It may be necessary to look at the assumptions, justification, and limitations of proposed methods and the acceptability of alternatives.

Some systems are purely discrete event and others are continuous systems. In this day and age, many systems are mixed continuous and discrete event. This aspect will have a strong influence on the simulation package choice.

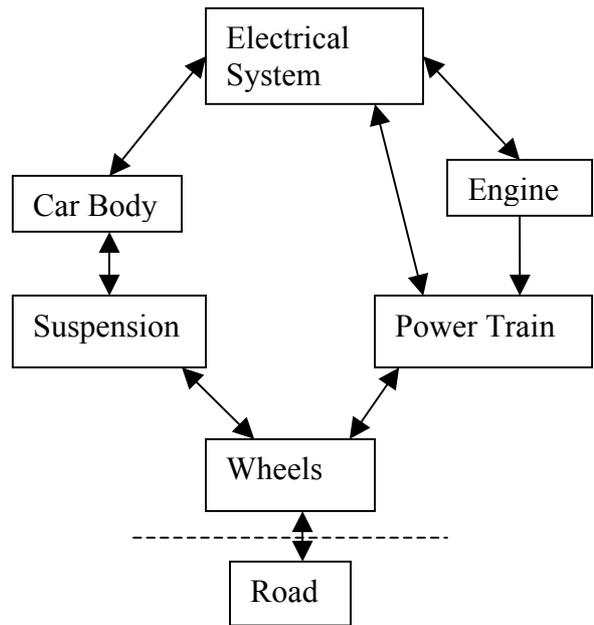


Figure 2: Car Top Level System Block Diagram

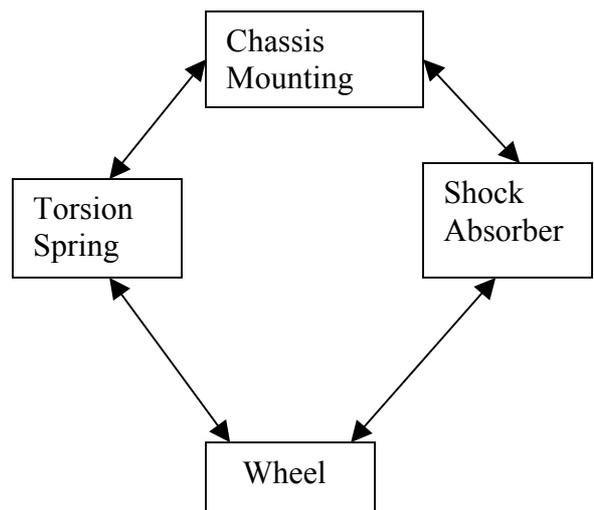


Figure 3: Lower Level Suspension Sub-System Block Diagram

## Mathematical Modelling

Second, the mathematical modelling is required (Korn and Korn 2000), if not already completely or partly already available from previous studies, or is partly available through adaptation of earlier models. However, all of this may be suspect, due to incomplete model specification or lack of detail, especially of assumptions and limitations associated with the model to be re-used. On completion, the mathematical models must be verified to determine if they are complete and correct, otherwise there is no point in proceeding to build a simulation from the models (Balci and Saadi 2002, Balci 2003).

## Simulation Construction

The simulation is then constructed, preferably by using a standard simulation package. This is often one that the organisation already has, but may not be the most suitable or appropriate. However, the simulationist may well already be familiar with the package, thus speeding up development of the simulation. A better or more suitable package may well require much learning before it can be used efficiently, properly or even reliably, and its cost and delivery time may be prohibitive, even if the package is desirable. It is important, if not essential, that the simulation package has a good graphical user interface and has, or is linked to, an animation package for visualisation of the simulation results. Such results may be in tabular, graphical or full 3D animation as in DIS/HLA simulations [Tandayya and Zobel 1998].

## Simulation Testing and Verification

After a simulation is built, it must be tested and its operation verified. The process is, of course, complex. It is similar to the testing of any complex piece of software (and hardware). This is best carried out by testing modules and then the interaction between modules, until the entire system has been tested. The process is the verification that the simulation operates as its designer intended. This does not mean that it is correct.

## Validation

For reasonably correct operation, the simulation must then be validated against the real world, or as near as possible to the real world as is available or feasible. It is important to remember that for new products or systems, there will be no real system to validate against. In this case the operation and performance of similar, perhaps earlier products or versions, together with the experience of engineers and simulationists has to suffice. The concept of *reasonableness* can be useful here. A degree of fuzziness is also required here. Only experience can be used to judge whether resulting simulation is sufficiently validated. It is useful to

introduce the concept of “this can not happen”, so do not test for this. Many system failures can be attributed to making this erroneous assumption.

## Certification

Certification of the validity of a simulation or simulator is required for systems which can be categorised as safety critical. This particularly applies to training simulators for civil and military aircraft, ships, land vehicles, weapons systems, air traffic control, nuclear power plant, etc. The certification authority in each area requires mandatory compliance and certification before use.

## SIMULATION STUDIES

The specification, design and execution of a simulation study is a complex matter. In the discussion below, many of the issues and topics are discussed. The list is not guaranteed to be exhaustive.

### Parameters

Parameters, or “constants”, of a system need to be specified before any study may be carried out. Although defaults can be useful, one incorrect default can make the result of an entire study useless (if discovered!).

### Initial values

A system has to start somewhere, often at time zero. However, all dynamic systems require initial values for all the system variables and for all of their derivatives, except the highest, which are algebraically determined from the relevant differential equation(s).

For example:

$$\text{If } a d^2x/dt^2 + b dx/dt + cx = f(t) \dots\dots\dots(1)$$

For equation 1, at  $t = 0$ , it is necessary to specify  $dx/dt(0)$ ,  $x(0)$  and  $f(0)$ .

Then  $d^2x/dt^2(0)$  is automatically specified by the equation after the values of the parameters  $a$ ,  $b$  and  $c$  have been set.

### Inputs

Inputs to a simulation are driving functions, forcing functions, outside influences, etc. For example, the vehicle block diagram illustrated in figure 2 shows a wheel with an external connection to a road. The deviation of the level of the road surface from a reference value is an input to the tyre on the wheel. The relative speed of the vehicle to the road determines the required rate of generation of the road surface function. This function may also contain a roughness noise

element. Other inputs may be weather, including effects of wind, rain, fog, and snow on the road and vehicle, other vehicles, road signals. In general terms, inputs are any external events and/or forces or conditions that, in some way, affect the simulation.

### **Noise components**

Many real systems include noise arising from a variety of sources. Examples are noise in a vehicle from wind arising from the motion of the vehicle and also meteorological wind. There is also tyre noise arising from the granularity of the road surface. There are many other naturally occurring sources of noise which affect the operation of systems designed by humans. Some of these may significantly affect system performance under certain conditions and thus noise signals may need to be included in a simulation.

Noise is generally considered to random with a distribution such Gaussian, Poisson or binomial. Noise occurring in real systems may have different distribution due to additions of specific components such power supply hum, sideband interference, or to modification due to filtering effects, resulting in what is loosely described as pink noise.

Noise frequency components may also vary with system parameters and current variable values. Variations may occur in both amplitude and frequency.

Noise sources include analogue generators of various types. However, digitally generated sources, using a variety of random number generation techniques have the advantage of repeatability, useful for replays and for studies with parameter variation requiring identical noise for each parameter setting.

Mathematicians are deeply suspicious about the use of digitally generated noise and other "random" number generators and their statistical relevance.

### **Parameter Studies, Statistics and Sensitivity**

Parameter studies are commonly carried out using system simulation to gather more information about a system. One of the most useful types of parameter study is optimization. This concerns finding the operational area, in relation to the major system parameters, in which the system operates in the most optimal way. Optimum here can mean many things, but usually it is related to the major performance criteria, such as speed, cost, availability, time, and customer expectations. Adjustment of key parameters can make an important difference to the performance of a system.

Usually and optimisation is carried out using a specific strategy such as hill climbing or steepest descent methods for a specific set of parameters. Of course, complex systems have many parameters, and

determining which of these are important is not an easy task. However, using statistical variation of parameters based on random selection can be quite useful.

A more directed approach is to start with a study of parameter sensitivity. This concerns the observation of the performance in terms of variation of each parameter to determine its sensitivity to variation.

An example of this might be a system in which there are two cascaded time constants (equation 2). One is large and the other is small. The system response time is normally dominated by the larger time constant, which implies that the smaller one is not particularly sensitive to variation in terms of the overall response time of the system, but may affect closed loop stability.

$$V_o/V_i(s) = K/(1 + sT_1)(1 + sT_2); T_1 \ll T_2 \dots\dots\dots (2)$$

The result of this may be, that in a parameter optimisation study, it is safe to disregard  $T_1$  as being not relevant for the study. This reduces the size and complexity of the parameter optimisation study.

### **Tolerance Studies**

Another type of parameter study which is commonly used is that concerning tolerances in order to determine the probability that a product will operate correctly with a given set of parameter tolerances. A typical example of this occurs in electronic circuits or in mechanical systems, where components are often used, having a tolerance of  $\pm n\%$ , where  $n$  should be as large as possible to minimize cost, but without compromising performance.

### **Systems with Several Modes of Operation**

A further complexity arises where a system has more than one mode of operation. A car, train or plane may have more than one state. For example, a car may be cruising on the autobahn, driven in town, stuck in a jam, stopped at traffic lights, parked or garaged. An aeroplane has other modes, such a taxi, take-off, climb, cruise, change course, descend, land, taxi, engines off with local services, parked, etc.

Such systems require a more complex simulation with manual or automatic sequence of mode changes. A study of the transients associated with mode change can be important.

### **Model Limitations**

Models are rarely fully general purpose. They are usually restricted to specific regions of parameter space, and are further often simplified or linearised to save on memory and/or processor time.

To achieve this, it is necessary to specify the limits of operation and use of such a model. It is essential to document, not only the limits, but also the reasons and assumptions which form the basis for the model simplification and the limits to its use. It is essential that when the limits are exceeded, even by a small amount, the validity of the simulation results must be questioned.

### **Exceptional Conditions**

Some years ago an aeroplane was flying at a height of 10km and at a speed of around 850km/h, i.e. normal cruising conditions. Suddenly, one of the two engines went into reverse thrust, a condition which should not happen. Although there are interlocks to prevent this from happening, it did happen. Pilot instructions offer little if anything to give a procedure to overcome this problem. The result was a tragedy.

Simulation was used to establish if this was a condition that could be recovered from. It is unlikely that the simulation model was valid for this exceptional condition.

Any simulation that is operated outside the anticipated parameter envelope should be re-validated as a minimum requirement.

### **Research**

Research continues into computer simulation as a useful, and in many cases essential tool for understanding and evaluating the complex systems that we design, build and use every day.

However, it is vital that we continue to live in the real world and not in a virtual world. We should remember that computers and simulators are tools. Producing a working simulation is *not* the object of the exercise. It is the verification, validation and intelligent use of the simulation which is its most important aspect.

There is a well known cynical expression “lies, damn lies and statistics”. One might insert “simulation” and/or “virtual reality” into the list. We commonly rely on simulation to add to our knowledge about a system, or to interpolate where we can only afford a few live system tests. A well known UK politician as heard to remark recently about the performance of a certain military system, but then added that this was only a simulation. He was clearly unaware that most complex military systems rely heavily on simulation throughout their design, manufacture, testing and use. The latter item is concerned with training simulators, which are now heavily employed to save, time, money, wear, and environmental damage. This now also applies to many non-military systems as is obvious from even a casual study of the proceedings of simulation conferences.

### **Understanding the Model Behaviour**

Many modern systems are complex or very complex. It is essential for all of those concerned with the engineering of such systems to be able to understand the function and operation of the overall system, its subsystems and the communications between the subsystems and the external world. Whilst the mathematical model is important, it is not always easy for those who have not been involved in the design to fully understand the significance of the details of the mathematical model.

Experience shows that simulationists can gain a lot of practical feel of the operation of a complex system during a simulation study. This is particularly true of simulationists, such as the author, who have much experience of a variety of different systems and how they operate. It is generally called having a *feel* for the system. It helps to identify faults or faulty operation when the system behaves in an unusual way. It also leads to a better understanding of the system. Of course, some new system advances do lead to unfamiliar behaviour. This is particularly true of digital filters, for example, which can do things that analogue filters cannot do.

### **Distributed Simulations**

Over the last 10 years or so, the linking of simulations and/or simulators together over the network has led to a whole new set of applications and problems. Not the least of these is the requirement for real-time operation with limited bandwidth and good virtual reality, especially for training simulators. This has resulted in the development of prediction techniques and minimisation or elimination of the effects of the actions of distant objects.

Further, for non-military applications, the requirement for security on the internet has led to the development and use of more secure systems for authentication and encryption (Roberts and Zobel 2004).

### **RESULTS**

Simulations can produce copious quantities of data in the form of results and the reasons for and conditions under which they have been obtained. It is important to have proper management of this in a usable form.

### **Recording of Simulation Results**

All sets of data from simulation runs and full details of the scenarios in which they were obtained must be recorded in a standard machine readable form. Casual recording of results really is a waste of time and can even be dangerous. Of course it is acceptable to play with a simulation to get a feel for a system, the concept of reasonableness comes into play here. Subsequent

simulation studies must be dealt with in a more formal manner if a truly professional approach is used, an essential for all serious simulationists.

### Analysis and Report Writing

With large quantities of data available some systematic method must be applied to reduce the important results to graphical and/or tabular form. There are, of course, tools available specifically for this purpose. Consequent upon this, reports need to be written for project meetings, records, management, etc, each in an appropriate style.

### CONCLUSIONS

It is clear from the above discussion, that simulation is a serious and complex activity. As with all such activities, experience is highly desirable. However, few people have a career in simulation. Many are involved in simulation only when their current project requires it, and then they move on. Consequently, the simulation community world wide is relatively small. For this reason it is desirable to have regular simulation conferences. This leads one to make contact with others who are working on similar activities in totally different topics or even areas. This in turn results in simulationists acquiring a very wide knowledge base. Long may it last.

### REFERENCES

- Balci O. and Saadi S.D. 2002, "Proposed Standard Processes for Certification of Modelling and Simulation Applications," In *Proceedings of the 2002 Winter Simulation Conference* (San Diego, CA, Dec. 8-11). IEEE, Piscataway, NJ, pp. 1621-1627.
- Balci O. 2003, "Verification, Validation, and Certification of Modelling and Simulation Applications," In *Proceedings of the 2003 Winter Simulation Conference* (New Orleans, LA, Dec. 7-10). IEEE, Piscataway, NJ, pp. 150-158.
- Gass, S. L. 1978. "Computer Model Documentation". In *Proceedings of the 1978 Winter Simulation Conference*. (Miami Beach, FL, Dec. 4-6). IEEE, Piscataway, N.J., 281-287.
- Habibi J and Zobel R.N. 1996. "Integration of Simulation and CAD Systems for a Virtual Factory within Synthetic Environments for Manufacturing." In *Proceedings of the ESS 96 Multiconference on Computer Simulation in Industry*, (Genoa, Italy, Oct.) SCS, Ghent, Belgium. pp. 86-90. ISBN 1-565-550-99-4
- Han. L. 2000. 'Development of Full Generic Verified and Validated Re-Usable Simulation Models'. MSc Thesis, Manchester University, UK.
- Korn, G.A., and T.M. Korn: 2000 *Mathematical Handbook for Scientists and Engineers*, 2nd Edition, Dover, New York.

Lee C-H, and Zobel R.N. 1997. "Model Re-Use Through the Use of a Simulation Model Library". In *Proceedings of the ESS 97 Multiconference on Computer Simulation in Industry*, (Passau, Germany, October). SCS, Ghent, Belgium. pp. 91-95. ISBN 1-565-550-99-4

Melsa, J. L, and D. Cohn 1978. *Decision and Estimation Theory*, McGraw-Hill, New York.

Roberts P.H. and Zobel R.N. 2004. "A Discussion of Elliptic Curve Cryptography and Configurable ECC System Design with Application to Distributed Simulation". *Invited paper in International Journal of Simulation, Systems, Science and Technology, Special Issue on Modelling and Simulation of Complex Systems. UKSim.Vol. 5, Nos 1-2*, pp. 47-57. 1-473-804X Online, ISSN 1-473-8031 Print

Tandayya P and Zobel R.N. 1998. "VISTA Real-Time Animation Software in a Distributed Spacecraft System". In *Proceedings of SESP'98, 5th Int. Workshop on Simulation for European Space Programs*, (ESTEC, Noordwijk, The Netherlands, Nov. 3-5).pp 183-190. ISSN 1022-6656.

### AUTHOR BIOGRAPHY



**RICHARD ZOBEL** graduated in Electrical Engineering from London University in 1963. His first experience of simulation was obtained during 1962-66 at Sperry Gyroscope whilst working on naval surface to air missiles, using mainly valve analog computers. His Ph.D., obtained in 1970 at Manchester University, concerned hybrid analog-digital computing. As Lecturer and Senior Lecturer he became involved in digital signal processing, instrumentation and design environments with special emphasis on the simulation aspects of real-time embedded systems. He is a Committee Member and former Chairman of the United Kingdom Simulation Society (UKSim), Former Secretary of the European Federation of Simulation Societies (EUROSIM), and was a European Director of SCS, the Society for Computer Simulation International. His current research interests concern distributed simulation for non-military applications, model re-use, distributed simulation model databases, issues of verification and validation of re-useable simulation models and security for distributed simulation under commercial network protocols. He is now an independent consultant, and still very active. He is currently teaching Computer Engineering during the winter months at the Prince of Songkla University, Phuket and Hat Yai Campuses in South Thailand.

# GRAPHING ZHUKOVSKI TRANSFORMATION IN DERIVE

Adam Marlewski  
 Institute of Mathematics, Poznań University of Technology  
 ul.Piotrowo 3a, 60-965 Poznań, Poland  
 e-mail: amarlew@math.put.poznan.pl

## KEYWORDS

airfoil, complex functions, computer algebra, Jowkowski, Zhoukovski

## ABSTRACT

Zhukovski (noted also as Joukovsky) transformation is a conformal mapping and generates families of orthogonal lines. It is famous for its application in aerodynamics where it reduces the flow around a wing profile to the flow around a circle. The paper gives a systematic presentation of Zhukovski map and transformations associated to it, with an emphasis on their visualisation and the use in the simulation. Taking into account that all mathematical manipulations and graphs are made within DERIVE 5.05 from Texas Instruments, the paper shows the potential use of this computer algebra system in lecturing, modelling and simulating plane flows.

1. The Russian physicist Nikolai Zhukovski (1847-1921) worked on fluid mechanics in both theoretical and experimental aspects, see e.g. Betyaev 2003, (Lazarev 1999). In particular, he showed that the investigation of the flow around an airfoil (the transversal section an aircraft wing or propeller) can be reduced to the flow around a circle. In particular, it lets

to simulate the flow around wing profiles via the observation of easier case which are phenomena around circles. This reduction bases on the function, which is called after his name a Zhukovski map. It is defined by the formula

$$f(z) := (z + a/z)/2,$$

where  $a$  is a positive number (and, usually,  $a = 1$ ).

Zhukovski transformation  $f$  is a conformal mapping, i.e. it map from the plane (which can be regarded as  $\mathbf{R}^2$  or  $\mathbf{C}$ ) to itself which preserves angles. That is, the angle between any two curves is the same as the angle between their images.

2. Using the Gauss representation  $z = x + y \cdot i$ , where both  $x$  and  $y$  are real and  $i := \sqrt{-1}$ , we have

$$f(x + y \cdot i) = u + v \cdot i,$$

where  $u := x \cdot (s + 1)/(2s)$ ,  $v := y \cdot (s - 1)/(2s)$

and  $s := x^2 + y^2$ .

Both the real part  $u$  and the imaginary part  $v$  of  $f$  satisfy Laplace equation of variables  $x$  and  $y$  (and, by the theory of analytic functions, it has to be), both they define lines in the real plane  $\mathbf{R}^2$ . These lines form

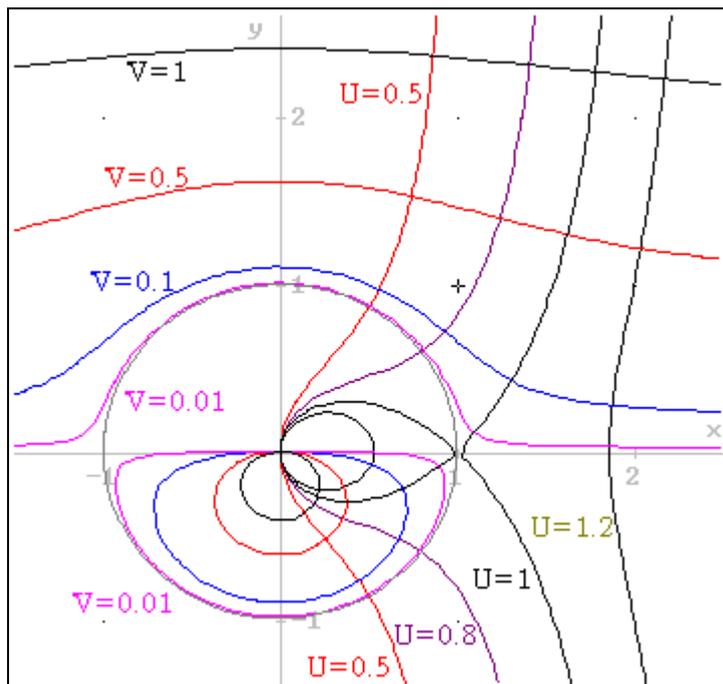


Figure 1: Lines governed by equations  $u = U$  with  $U = 0.5, 0.8, 1, 1.2$  and  $v = V$  with  $V = 0.01, 0.1, 0.5, 1$ . Moreover, there is drawn the unit circle  $x^2 + y^2 = 1$ .

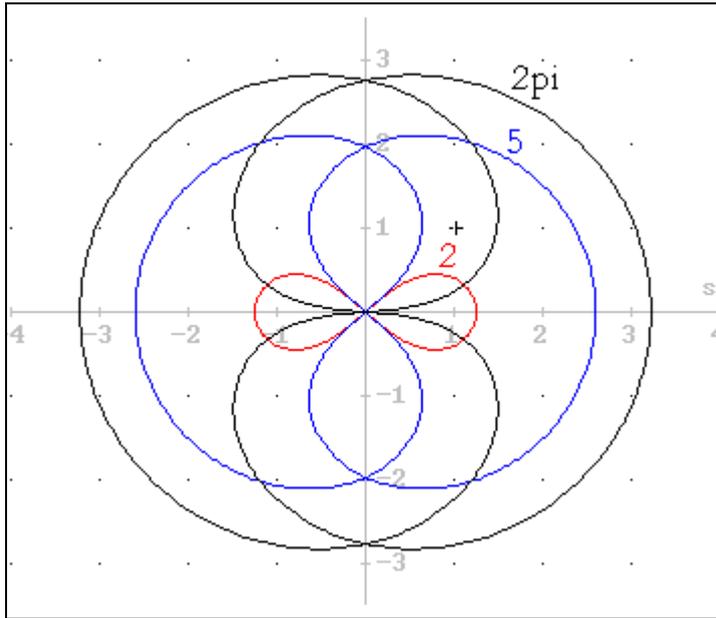


Figure 2: Lines in the polar coordinate system  $O_s\varphi$  governed by equations  $s = (r + a/r)/2 \cdot \cos(\theta)$ ,  $\varphi = (r - a/r)/2 \cdot \sin(\theta)$ , for  $a = 1$ ,  $r = 2, 5, 2\pi$ , and  $\theta$  running from  $0$  to  $2\pi$

orthogonal families, see also (Puel 1999). Some lines governed by equations  $u = U$  and  $v = V$  are shown in Fig.1. In fluid mechanics these lines are called that of equivelocity lines and streamlines, respectively. The last ones are also shown in Fig.10.

3. Obviously, the trigonometric representation  $z = r \cdot \exp(i \cdot \theta)$ , where both  $r, \theta$  are real (and interpreted as the distance to the origin  $O$  and the angle, respectively), gives the same lines, in the polar coordinate system  $O_r\theta$  they are governed by the equations  $u = U$  and  $v = V$ , where  $u$  and  $v$  are real and imaginary parts of  $f(r \cdot \exp(i \cdot \theta))$ , i.e.

$$u = (r + a/r)/2 \cdot \cos(\theta),$$

$$v = (r - a/r)/2 \cdot \sin(\theta).$$

4. Let's use above relations to define the function

$$g(r, \theta) := [(r + a/r)/2 \cdot \cos(\theta), (r - a/r)/2 \cdot \sin(\theta)]$$

depending on two real parameters  $r$  and  $\theta$ .

For any fixed  $r = R$  in the rectangular coordinate system  $Ouv$  these equations determine ellipses. Shapes traced in the polar coordinate system  $O_s\varphi$  are also finite and closed graphs, see Fig.2.

For example, for  $a=1$  and  $R=5$  we have  $s = 13/5 \cdot \cos(\theta)$ ,  $\varphi = 12/5 \cdot \sin(\theta)$ , see Fig.3, and its shape is close a petal of Grandi rose (see Xah Lee, "A Visual Dictionary of Famous Plane Curves", [www.xahlee.org/SpecialPlaneCurves\\_dir/Rose\\_dir/rose.html](http://www.xahlee.org/SpecialPlaneCurves_dir/Rose_dir/rose.html)) covered by the equation  $s = 13/5 \cdot \cos(\varphi)$ . Setting  $r = a = 1$  we get the interval  $\langle -1, 1 \rangle$  on the real axis.

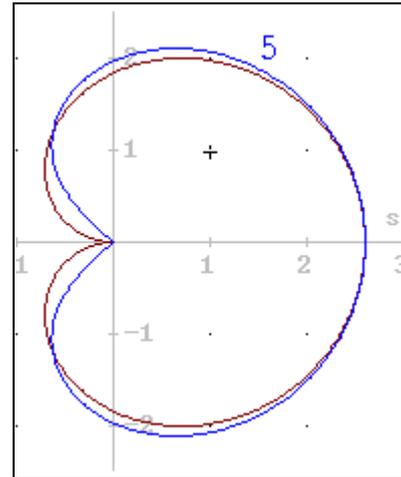


Figure 3: In the polar coordinate system  $O_s\varphi$ : the line governed by the parametric representation  $s = 13/2 \cdot \cos(\theta)$ ,  $\varphi = 12/2 \cdot \sin(\theta)$ , where  $\theta$  runs from  $-\pi/2$  to  $\pi/2$ , and the arc of Grandi rose  $s = 13/5 \cdot \cos(\varphi/2)$  traced when  $\varphi$  runs from  $0$  to  $\pi$

Fixing the parameter  $\theta$ , in the rectangular coordinate system  $Ouv$  we get the infinite curves which describe the stream lines of the fluid passing along the vertical axis  $Ov$  and hampered by an infinitesimally thin segment laying on the horizontal axis from the point  $(a, 0)$  to the right. For  $a = 1$  and some values of  $\theta$  see these lines in Fig.4. They include  $\theta = 0^\circ$  and  $\theta = 90^\circ$ , and then we have the obstacle and the vertical axis, respectively.

The parameter  $\theta$  fixed in the polar coordinate system  $O_s\varphi$ , we get rather complicated curves. An arc of one of them is traced in Fig.5.

5. Let's take the quotient of quantities forming the parametric representation  $g(r, \theta)$ . Denoting it by  $q$  we have  $q = (r^2 + a)/(r^2 - a) \cdot \cot(\theta)$ . With  $\theta$  fixed it defines the function of  $r$ . A graph of this function in the polar coordinate system  $Oqr$  (so here  $r$  is must be treated as the angle, not as the radius) for  $\theta = 30^\circ$  is shown at Fig.6. One can compare it to the graph plotted in Fig.7 besides, in J.Wassenaar's "Mathematical curves" (<http://www.2dcurves.com>) the graphs similar to that last are called atom spirals (first produced by Annie van Maldeghem in 2002).

Both relations,  $q = (r^2 + a)/(r^2 - a) \cdot \cot(\theta)$

$$\text{and } q = (r + a)/(r - a) \cdot \cot(\theta),$$

in the rectangular coordinate system  $Orq$  (so  $r$  marks the horizontal axis, and  $q$  is measured along the vertical axis) describe well-known hiperbola and an other simple curve, see Fig.8.

6. A flow around a circular cylinder is a fundamental fluid dynamics problem of practical importance. In general case there isn't known its

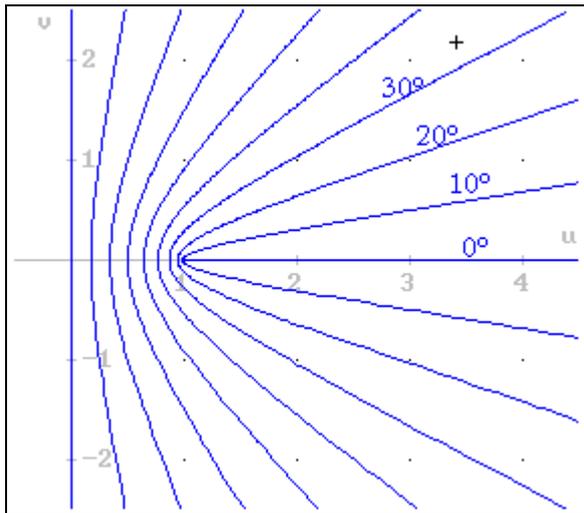


Figure 4: Lines in the rectangular coordinate system  $Uov$  governed by equations  $u = (r + a/r)/2 \cdot \cos(\theta)$ ,  $v = (r - a/r)/2 \cdot \sin(\theta)$ , for  $a = 1$ ,  $\theta = -0^\circ, 10^\circ, 20^\circ, \dots, 80^\circ, 90^\circ$ , and  $r$  running the interval  $(-\infty, +\infty)$

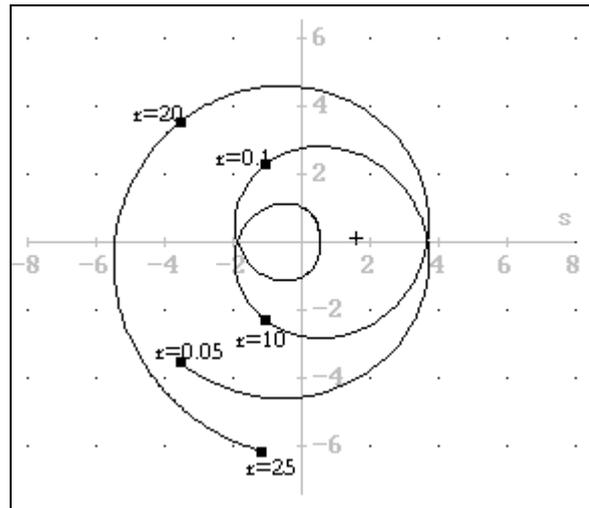


Figure 5: The graph in the polar coordinate system  $Os\phi$  governed by the parametric representation  $s = (r + a/r)/2 \cdot \cos(\theta)$ ,  $\phi = (r - a/r)/2 \cdot \sin(\theta)$ , for  $a = 1$ ,  $\theta = 30^\circ$ , and  $r$  running the interval  $(0.05, 25)$

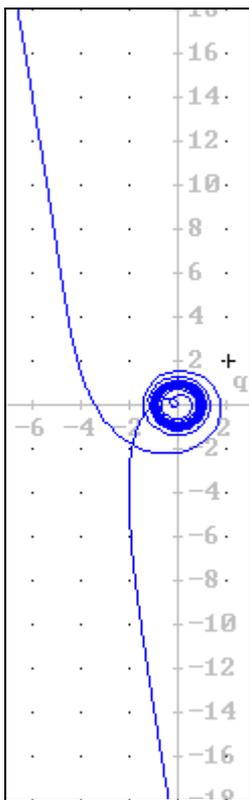


Figure 6:  $q = \sqrt{3} \cdot (r+1)/(r-1)$  in the polar coordinates  $(q,r)$

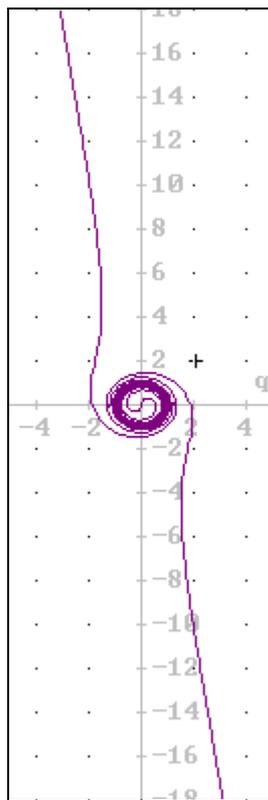


Figure 7:  $q = \sqrt{3} \cdot (r^2+1)/(r^2-1)$  in the polar coordinates  $(q,r)$

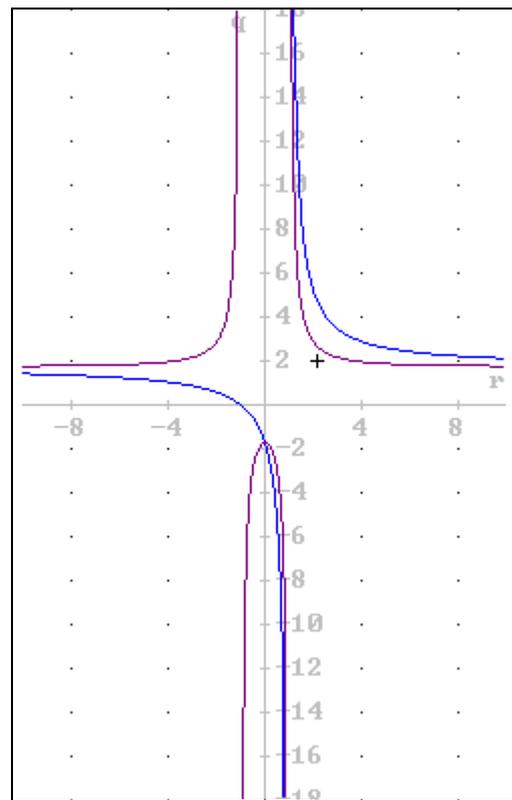


Figure 8: Hiperbola  $q = \sqrt{3} \cdot (r+1)/(r-1)$  and the line  $q = \sqrt{3} \cdot (r^2+1)/(r^2-1)$  in the rectangular coordinates  $(r,q)$

analytical solution, the experiments reveal its character essentially depends on the Reynolds number: at its low values, below circa 80, the flow field is symmetric, and the flow begins to separate (and the unsteady phenomenon, the vortex shedding, takes place) as the Reynolds number

increases. In case of an uniform flow given simply as the map  $z \rightarrow v \cdot z$ , where  $v$  stands for the velocity of the flow of the fluid, by the circle theorem (a.k.a. Milne-Thompson Theorem) we obtain that the complex potential of the flow past a disk of radius  $\sqrt{a}$  is  $f(z)$ , and its real and imaginary parts are called

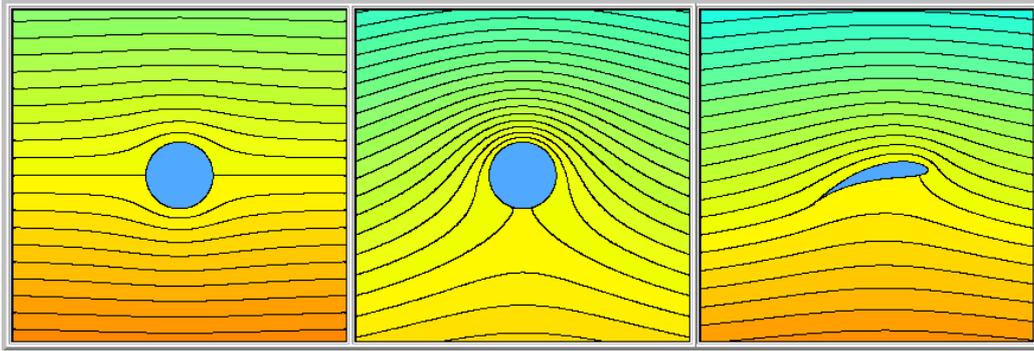


Figure 9: The streamlines of the uniform flow passing along the horizontal axis, from right to left, around a circle with zero circulation and with non-zero (two first captures) and around the airfoil (the right capture), where the attack angle is taken into account by the positioning of the airfoil (M. Colombini, “Moti irrotazionali piani di fluidi ideali”, [http://www.diam.unige.it/~irro/lecture\\_e.html](http://www.diam.unige.it/~irro/lecture_e.html)).

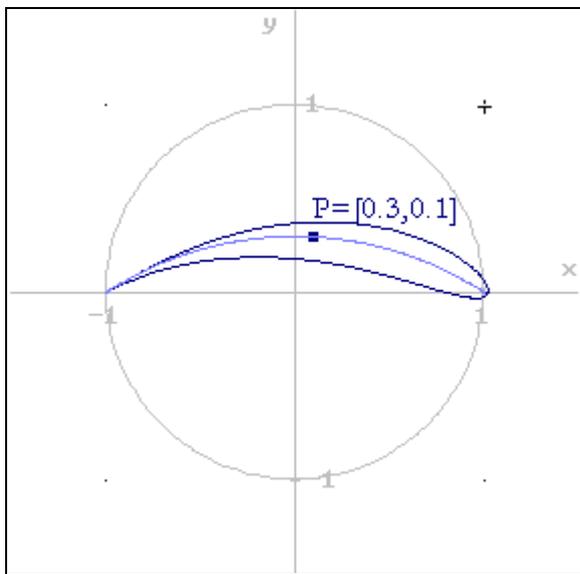


Figure 10: Zhukovskii airfoil determined by the point  $P = (0.1, 0.3)$  and its central line which is Zhukovskii profile determined by  $(0, 0.3)$

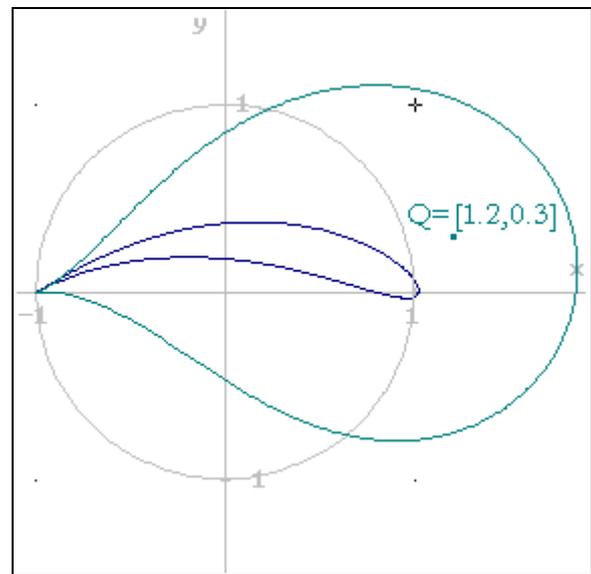


Figure 11: Zhukovskii airfoil copied from Fig.10 and Zhukovskii profile determined by  $Q = (1.2, 0.3)$

velocity and stream function, respectively. To take into account the attack angle  $\alpha$  and the circulation parameter  $\gamma$  we modify the Zhukovskii function to have the map

$$z \rightarrow f(z \cdot \exp(\alpha \cdot i)) + \gamma \cdot \ln(z \cdot \exp(\alpha \cdot i)) / (2\pi).$$

Analogical situation occurs in the aerodynamics, and it is even more difficult because there have to be considered the shapes which are more complex than a cylinder (see Fig.9). A typical shape is the airfoil (such as in Fig.10). The Zhukovskii function transform a circle (which, obviously, is a trace of a cylinder on a plane) into an airfoil, and its inverse,  $w \rightarrow -w \pm \sqrt{(w^2 - a)}$ , realises the back mapping.

7. The image of a circle under the Zhukovskii function is called (its) Zhukovskii profile. Let a circle be centered at the point  $c_0$  and passes through the

point  $(-a, 0)$ . Then its Zhukovskii profile reduces to the interval  $\langle -a, a \rangle$  if  $c_0 = (0, 0)$ , and to a symmetric arc if  $c_0 = (0, y_0) \neq (0, 0)$ .

8. A really wide applications of Zhukovskii function  $f$  is in aerodynamics, where it transforms a circle into the curve called Zhukovskii profile. If  $a = 1$ , this circle contains the point  $(-1, 0)$  of the Cartesian plane. Thus in the complex plane  $Oz$  it has the equation

$$z = s_0 + r_0 \cdot \exp(i \cdot \theta),$$

where  $s_0 = x_0 + y_0 \cdot I$ ,

$$r_0 = \sqrt{\{(x_0 + 1)^2 + y_0^2\}}.$$

Now the formulas  $x = \text{Re}(f(s_0 + r_0 \cdot \exp(i \cdot \theta)))$ ,

$$y = \text{Im}(f(s_0 + r_0 \cdot \exp(i \cdot \theta)))$$

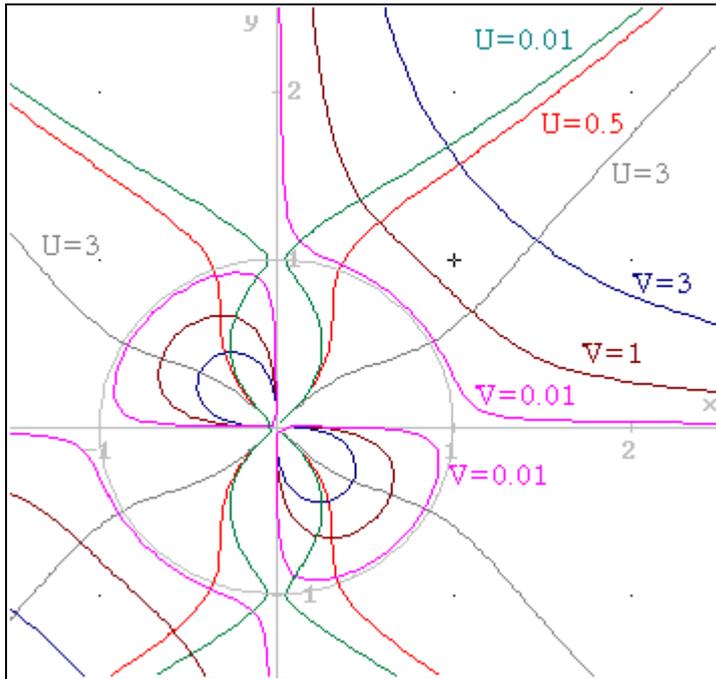


Fig.12. Lines governed by equations  $u = U$  ( $U = 0.01, 0.5, 1, 3$ ) and  $v = V$  ( $V = 0.01, 1, 3$ ), where  $u$  and  $v$  are real and imaginary parts of the expression  $f^2(x+y-i)$ . Identically as in Fig.1, it is also drawn the unit circle  $x^2 + y^2 = 1$ .

cover the Zhukovski profile with respect to the point  $s_0$ . For example for  $s_0 = (1+3-i)/10$  they are

$$x = (\sqrt{130}\cos(\theta)+1)\cdot(m+1/20)$$

$$y = (-\sqrt{130}\sin(\theta)+3)\cdot(m+3/20)$$

where  $m := \sqrt{10} / \{4 \cdot (\sqrt{13}\cos(\theta) + 3 \cdot \sqrt{13}\sin(\theta) + 7\sqrt{10})\}$ , see Fig.10.

9. It's clear, the Zhukovski transformation may be involved in other operations. An example is given in (Nersessian and Ter-Antoyan 1997) where Zhukovski function is used to parametrize the oscillator trajectories. It leads to the square of Zhukovski function, i.e. the  $z$  complex plane is mapped by the function

$$f^2 : z \rightarrow (z + 1/z)^2 = z^2 + 2 + z^{-2}.$$

Some of curves determined by this transformation are reproduced in Fig.12, and it is worth to compare them to that in Fig.1.

Another example dealing with Zhukovski and Karman-Trefftz mappings is (Czeńnik and Prosnak 2002).

A wide spectrum of web-pages with Zhukovski airfoils is offered by J.H.Matthews in "Joukowski transformation and airfoils" ([http://math.fullerton.edu/mathews/c2003/Joukowski Trans Bib.html](http://math.fullerton.edu/mathews/c2003/Joukowski%20Trans%20Bib.html)). In R.Ferréol's "Coubre de Joukovski, profile d'aile d'avion" (<http://www.mathcurve.com/courbes2d/joukowski/joukowski.shtml>) there is presented the animation showing that the Zhukovski airfoil is the locus of middle points of appropriate segments having ends on two circles.

10. In this paper all figures except Fig.9 are plotted by DERIVE 5.04, a computer algebra system from Texas Instruments, Inc.; for more information see producer's webpage as well as numerous papers, a.o. (Jankowski and Marlewski 2005), where there are widely discussed advances of the use of symbolic algebra in sciences and education. All mathematical transformations were also did within this system. This paper is completed with an Appendix, a collection of definitions to handle the subject in DERIVE. If this system is in use, in lessons in complex variable functions and fluid mechanics the functions listed in Appendix can help teachers and students to handle classical Zhukovski mapping as well as similar transformations, as Karman-Trefftz maps discussed in (Iollo and Zannetti 2000) and (Simakov et al. 2000).

11. **COCLUSIONS** Maple from Waterloo Maple Inc., Mathematica from Wolfram Research Inc., MatLab from MathWorks Inc. and other computer algebra systems offer additional packages or files to handle Zhukovski transformation. Till now in this family of computer programs for symbolic manipulations there is absent the system DERIVE, where a good quality and high achievements meet moderate price and user-friendly control, so it conquers still more and more students all over the world, and its educational are well stated, see e.g. (Koepef 2000). Its research utility is also well acclaimed, see e.g. (Marlewski and Kołodziej 1997). Surely, DERIVE may be applied to investigate Zhukovski transformation, too. This paper proposes the way at which it can be done and,

at the same time, wants to show potential areas to be covered by this approach.

## REFERENCES

- Betyaev, S.K. "On the history of fluid dynamics: Russian scientific schools in the 20th century", *Phys. Usp.*, 2003, 46 (4), 405-432
- Cześniak, P. and W. Prosnak, "Conformal mapping of the Gulf of Gdańsk onto a canonical domain", *Oceanografia*, 44 (2), 2002, 179-207
- Koepf, W., Mathematics with DERIVE as didactical tool, *DERIVE-Newsletter* 38, 2000, 23-35
- Lazarev, P.P., "Historical essay on the 200 years of the development of natural sciences in Russia", *Phys. Usp.*, 1999, 42 (12), 1247-1257
- Iollo, A. and L. Zannetti, "Optimal Control of a Vortex Trapped by an Airfoil with a Cavity", *Applied Scientific Research*, 2000, vol. 65, no. 3-4, pp. 417-430
- Jankowski, L. and A. Marlewski, "Troubles with a heart-shaped curve", *Pro Dialog* 19 (2005) in print
- Marlewski, A. and J. Kołodziej, "DERIVE assistance to the collocation method for solving potential flow problems", *Int. J. Math. Educ. in Sc. and Techn.* (1997), Vol.27, No.6, 875-882

- Nersessian, A., Ter-Antoyan, V.M., "Anyons, monopole and Coulomb problem", *Phys. Atom. Nucl.* 61 (1998) 1756-1761
- Simakov, S.T.; A.S. Dostovalova; and E.O. Tuck, "A GUI for computing flows past general airfoils", *Australian MATLAB User Conference 2000*, 1-18
- Puel, F., "Potentials having two orthogonal families of curves as trajectories", *Celestial Mechanics & Dynamical Astronomy* 74 (3): 199-210, 1999

All web-pages cited in the text were accessible on January 29, 2005



**ADAM MARLEWSKI**, born in Czempień, Poland, studied numerical methods at Adam Mickiewicz University in Poznań. In 1981 he got Ph.D. for the dissertation on the Bernstein polynomial approximation. Since 1973 he works for Poznań University of Technology, in 1992-94 he lectured in Universidade da Beira Interior in Covilhã, Portugal. He published books on numerical methods and DERIVE, he co- and authored papers in pure and applied mathematics, on computer algebra systems, induction machines, simulation in remote sensing and in civil engineering. His webpage is <http://www.math.put.poznan.pl/~amarlew>.

## APPENDIX

A collection of expression to handle Zhukovski map and its associates in DERIVE 5.04

$ZhukovskiFunction(a,z):=(z+a/z)/2$

$ZhukovskiFunction(a,x+y \cdot i)$  simplifies to  $a \cdot x / (2 \cdot (x^2 + y^2)) + x / 2 + i \cdot (y / 2 - a \cdot y / (2 \cdot (x^2 + y^2)))$ .

$u:=RE(ZhukovskiFunction(a,x+y \cdot i))$  and  $v:=IM(ZhukovskiFunction(a,x+y \cdot i))$

returns the real and imaginary parts of the above expression, respectively.

$u=U$  and  $v=V$  plot graphs as that shown in Fig.1, if earlier it is set  $a=1$ .

$flowAroundCylinder(\alpha,\gamma,z):=(z \cdot EXP(\alpha \cdot i) + 1/z \cdot EXP(-\alpha \cdot i))/2 + \gamma/(2 \cdot \pi) \cdot LN(z \cdot EXP(\alpha \cdot i))$

where  $\alpha$  and  $\gamma$  stand for the attack angle and the circulation parameter, respectively.

$flowAroundCylinder(\alpha,\gamma,x+y \cdot i)$  simplifies to Gauss form of the above complex expression.

$velAC(\alpha,\gamma,x,y):=RE(flowAroundCylinder(\alpha,\gamma,x+y \cdot i))$  returns the velocity.

$strAC(\alpha,\gamma,x,y):=IM(flowAroundCylinder(\alpha,\gamma,x+y \cdot i))$  returns the stream.

The simplification of  $velAC(\alpha,\gamma,x,y)=c$  is plotted as the line of the velocity equal to a constant  $c$ .

The simplification of  $strAC(\alpha,\gamma,x,y)=c$  is plotted as the streamline corresponding to a constant  $c$ .

$complexCircle(c,\theta):=c_1 + i \cdot c_2 + \sqrt{(c_1+1)^2 + c_2^2} \cdot (COS(\theta) + i \cdot SIN(\theta))$

defines a circle centered at a point  $c=(c_1,c_2)$  and passing through the point  $(-1,0)$ .

Given  $c$  and  $a=1$ , the simplification  $p:=ZhukovskiFunction(complexCircle(c,\theta))$

returns the complex expression, its real and imaginary parts compose the vector

$airfoil(c,\theta):=[RE(p),IM(p)]$

which parametrically describes a Zhukovski profile. For appropriate point  $c$  ( $c$  has to be a point within the unit circle, but not on the vertical axis) it returns the Zhukovski airfoil. For example,  $airfold([0.3,0.1],\theta)$

simplifies to  $[\sqrt{10} \cdot (\sqrt{130} \cdot COS(\theta) + 1) / (4 \cdot (\sqrt{13} \cdot COS(\theta) + 3 \cdot \sqrt{13} \cdot SIN(\theta) + 7 \cdot \sqrt{10})) + \sqrt{130} \cdot COS(\theta) / 20 + 1/20,$   
 $-\sqrt{10} \cdot (\sqrt{130} \cdot SIN(\theta) + 3) / (4 \cdot (\sqrt{13} \cdot COS(\theta) + 3 \cdot \sqrt{13} \cdot SIN(\theta) + 7 \cdot \sqrt{10})) + \sqrt{130} \cdot SIN(\theta) / 20 + 3/20],$

and its graph, traced for  $\theta$  running the interval  $\langle 0, 2\pi \rangle$ , is shown in Fig.10.



# **Sim-Serv-Session**



# MODELING AND COMPUTER SIMULATION FOR THE PREDICTION OF FORCES IN HIGH-SPEED MACHINING PROCESSES

Angel Alique  
Rodolfo Haber  
José R. Alique  
Salvador Ros

Instituto de Automática Industrial  
Consejo Superior de Investigaciones Científicas  
Ctra. de Campo Real km. 0.200. La Poveda  
28500, Madrid, Spain  
E-mail: a.aliq@iai.csic.es

## KEYWORDS

Forces, high-speed machining, modeling, simulation.

## ABSTRACT

This paper reports some results in modeling and simulation of the high-speed cutting process based on cutting force signal. The mathematical model has two fundamental, co-dependent parts. The first is a multiple-input system that defines the model's kinematics, where constants and variables describe the tool geometry, the material type and the cutting parameters. The second addresses the dynamics, represented by integral-differential equations. The results of the simulation for cylindrical mills show the suitability of forces signal in describing the high-speed machining processes.

## INTRODUCTION

One of the basic tasks manufacturing systems have to perform today is machining, especially high-speed machining (HSM) (Haber et al. 2004a). High-speed machining may be defined according to the available literature as the process of machining at the highest possible cutting speeds allowed by existing restrictions set by the workpiece and the material to be machined, the machine tools to be used and the control and operating devices available (e.g., CAD/CAM systems, numerical controls). In this paper, we focus on high-speed milling machines where the workpiece remains fixed and the tool is clamped to a rotating head. Using this high-performance rotating head and working with each of its axes, cutting speeds can be reached that are much higher than those found in conventional machining, leading to what is known nowadays as high-performance machining (HPM).

The conventional machining process may be regarded as a complex electromechanical process in which force and torque signals are representative of the physical processes taking place during cutting (Haber et al. 2004b). Precise knowledge of the dynamic behavior of such signals can be interpreted to evaluate the status of

the cutting process, so force and torque signals are extremely useful in monitoring and control systems. Despite the enormous progress made in HSM, it is unclear what usefulness the mathematical basis already developed for conventional machining may have for modeling and characterizing high-speed machining. In addition, more light needs to be thrown upon the role that variables such as cutting force play in HSM, where they may furnish relevant information on cutting-tool condition, vibration and surface finish.

The main goal of this paper is to derive a mathematical model from the characterization of the physical processes taking place during high-speed machining and to perform simulations. Modeling shall be done following classic patterns used in conventional machining, looking upon cutting force as the output variable.

At the present time, modeling high-speed machining processes, especially high-speed milling, is a very active area of investigation that is peppering the scientific community with challenges. High-speed machining has now been adopted and put into regular use at many companies, and yet certain points of how to monitor cutting-tool condition have been worked out only partially as yet, largely for lack of a mathematical model of the process that can feasibly be used in real-time applications. Even so, HSM today is a cutting technology whose solid bases open the doors to the possibility of machining materials of a hardness of over 50 HRc and narrow walls just 0.2 mm thick.

If we confine ourselves to HSM, and more particularly the modeling of HSM, and we analyze the literature available to us, we can see that there are few papers dealing with the relationship of cutting force as an output variable with the constants and input variables that define mill or cutting-tool geometry, the type of material to be machined and the cutting parameters themselves.

## GEOMETRIC MODEL OF THE TOOL

Geometrical modeling of the helical cutting edge includes the kinematic and dynamic analysis of the cutting process. Predicting cutting forces requires a system of coordinates, the helix angle and the angular distance of a point along the cutting edge (Yucesan and Altintas 1996). The mathematical expressions that define this geometry in a global coordinate system are presented below in the geometric model, using classic vector notation.

Vector  $\vec{r}(z)$  drawn from point  $O$  to a point  $P$  in cylindrical coordinates is expressed mathematically in equation 1.

$$\vec{r}_j = x_j \vec{i} + y_j \vec{j} + z_j \vec{k} = r(\phi_j)(\sin \phi_j \vec{i} + \cos \phi_j \vec{j}) + z(\phi_j) \vec{k} \quad (1)$$

where  $\phi_j$  is the radial rake angle of a point  $P$  at tooth  $j$ .

Point  $P$  lies at an axial depth of cut  $a_p$  in the direction of axis  $Z$ , at a radial distance  $r(z)$  on the  $XY$  plane, with an axial rake angle  $\kappa(z)$  and a radial lag angle of  $\psi(z)$ .

The geometry of the tool is represented mathematically, considering that the helical cutting edge wraps parametrically around a cylinder. The mathematical model dictated for the cutting edge considers that the edge is divided into small increments, where the cutting coefficients can vary for each location. The initial point of reference to the cutting edge of the tool ( $j = 1$ ) is considered to be the angle of rotation when  $z = 0$  is  $\phi$ . The radial rake angle for the cutting edge  $j$  in a certain axial position  $z$  is expressed as:

$$\phi_j(z) = \phi + \sum_{n=1}^j \phi_p - \psi(z) \quad (2)$$

The lag angle  $\psi(z)$  appears due to the helix angle  $\theta$ . This angle is constant in the case of a cylindrical mill, and it varies for a ball-end mill. In the generalized model for the geometry of a mill with helical teeth, the tool diameter may differ along the length of the tool, depending on the shape of the tool (cylindrical, ball-end, spherical, angular, etc.). An infinitesimal length of this cutting edge may be expressed as

$$\begin{aligned} dS &= |dr| = \sqrt{r^2(\phi) + (r'(\phi))^2 + (z'(\phi))^2} d\phi \\ r'(\phi) &= \frac{dr(\phi)}{d\phi} \\ z' &= \frac{dz(\phi)}{d\phi} \end{aligned} \quad (3)$$

Chip thickness changes as a function of the radial rake ( $\phi$ ) and axial rake ( $\kappa$ ):

$$h_j(\phi_j) = s_{ij} \sin \phi_j \cdot \sin \kappa \quad (4)$$

As mentioned above, this paper looks at cylindrical mills. Analyzing the geometry, for a cylindrical mill, the following conditions are defined for finding the general solution:

$$r(z) = \frac{D}{2} \quad (5)$$

$$\kappa = 90^\circ \quad (6)$$

$$\psi = k_\theta z \quad (7)$$

$$k_\theta = (2 \tan \theta) / D \quad (8)$$

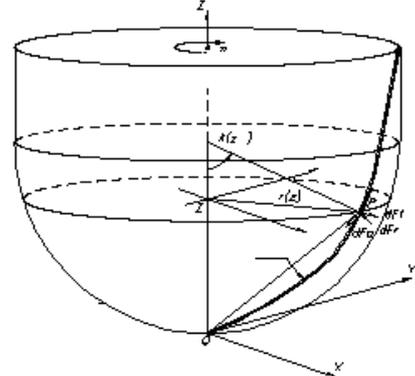


Figure 1: Tool geometry

## MODEL OF CUTTING FORCE

In order to find the cutting forces in high-speed machining, cutting is considered to occur in opposition. The force differentials ( $dF_t$ ), ( $dF_r$ ), ( $dF_a$ ) act on an infinitesimal element of the cutting edge of the tool (Altintas and Lee 1998):

$$\begin{aligned} dF_t &= K_{te} dS + K_{tc} h_j(\phi, \kappa) db \\ dF_r &= K_{re} dS + K_{rc} h_j(\phi, \kappa) db \\ dF_a &= K_{ae} dS + K_{ac} h_j(\phi, \kappa) db \end{aligned} \quad (9)$$

It is also considered that:

$$db = \frac{dz}{\sin \kappa} \quad (10)$$

In order to facilitate finding the mathematical relations inherent in this set-up, very small time increments are used. The positions of the points along the cutting edge are evaluated with the geometrical model presented herein above.

Furthermore, the characteristics of a point on the cutting surface are identified using the properties of kinematic rigidity and the displacements between the tool and the workpiece. The constants or cutting coefficients ( $K_{te}$ ,  $K_{re}$ ,  $K_{ae}$ ,  $K_{tc}$ ,  $K_{rc}$ ,  $K_{ac}$ ) can be found experimentally using cutting forces per tooth averaged for a specific type of tool and material (Fu et al. 1984, Budak et al. 1996). We might point out that these coefficients are highly dependent on the location (axial depth) of the cutting edge. How these coefficients are found shall not be addressed in this paper.

Cutting forces can be evaluated employing a system of Cartesian coordinates:

$$\begin{bmatrix} dF_x \\ dF_y \\ dF_z \end{bmatrix} = \begin{bmatrix} -\sin \phi \sin \kappa & -\cos \phi & -\sin \phi \cos \kappa \\ -\cos \phi \sin \kappa & \sin \kappa & -\cos \phi \cos \kappa \\ -\cos \kappa & 0 & -\sin \kappa \end{bmatrix} \cdot \begin{bmatrix} dF_r \\ dF_t \\ dF_a \end{bmatrix} \quad (11)$$

The total cutting forces as a function of  $\phi$  are found by integrating equation (11) along the axial depth of cut for all the lips of the mill that are in contact with the workpiece:

$$\begin{aligned} F_x(\phi) &= \sum_{j=1}^{Nf} \int_{z_1}^{z_2} [-dF_{rj} \sin \phi_j \sin \kappa_j - dF_{tj} \cos \phi_j - dF_{aj} \sin \phi_j \cos \kappa_j] dz \\ F_y(\phi) &= \sum_{j=1}^{Nf} \int_{z_1}^{z_2} [-dF_{rj} \cos \phi_j \sin \kappa_j + dF_{tj} \sin \phi_j - dF_{aj} \cos \phi_j \cos \kappa_j] dz \\ F_z(\phi) &= \sum_{j=1}^{Nf} \int_{z_1}^{z_2} [-dF_{rj} \cos \kappa_j \quad 0 \quad -dF_{aj} \sin \kappa_j] dz \end{aligned} \quad (12)$$

where  $z_1$  and  $z_2$  are the integration limits of the contact zone at each moment of cutting and can be calculated from the geometrical model described herein above. For the numerical calculation, the axial depth of cut is divided into disks having an infinitesimal height  $dz$ . The differentials of the cutting forces are calculated along the length of the cutting edge in contact, and they are summed to find the resulting forces for each axis  $F_x(\phi)$ ,  $F_y(\phi)$ ,  $F_z(\phi)$  in an angle of rotation defined by

$$\phi = \Omega \cdot dt \quad (13)$$

where  $\Omega$  is the speed of the head in ( $rad/s$ ) and  $dt$  is the time differential in the interval for the integration.

The left end of the tool is the initial point of reference for the radial rake angle ( $\phi$ ), designated by the distance  $a_e$  as the entry point and  $a_s$  as the exit point (figure 2). The points lying between  $a_e$  and  $a_s$  remain at the angles designated by:  $\phi_j(0) = \phi + j\phi_p$ ;  $j = 1, 2, \dots, (Nf)$ , where  $j$  indicates the lip of the tool.

The rake angle for lip  $j$  of the mill, due to the depth of cut, is defined along the  $Z$  axis by:

$$\phi_j(z) = \phi + j\phi_p - k_\theta z \quad (14)$$

in which chip thickness:

$$h_j(\phi, z) = s_{ij} \sin \phi_j(z) \cdot \sin \kappa(z) \quad (15)$$

The cutting constants can be calculated using equation (19) and the transformation of the orthogonal cut into an oblique cut, regarding the helix angle as an oblique cutting angle (i.e.,  $i = \theta$ ) (Engin and Altintas 2001).

$$\begin{aligned} K_{tc} &= \frac{\tau_s}{\sin \phi_n} \frac{\cos(\beta_n - \alpha_n) + \tan i \tan \eta \sin \beta_n}{\sqrt{\cos^2(\phi_n + \beta_n - \alpha_n) + \tan^2 \eta \sin^2 \beta_n}} \\ K_{ac} &= \frac{\tau_s}{\sin \phi_n} \frac{\sin(\beta_n - \alpha_n)}{\cos i \sqrt{\cos^2(\phi_n + \beta_n - \alpha_n) + \tan^2 \eta \sin^2 \beta_n}} \\ K_{rc} &= \frac{\tau_s}{\sin \phi_n} \frac{\cos(\beta_n - \alpha_n) \tan i - \tan \eta \sin \beta_n}{\sqrt{\cos^2(\phi_n + \beta_n - \alpha_n) + \tan^2 \eta \sin^2 \beta_n}} \end{aligned} \quad (16)$$

where  $\tau_s$  is the cutting or shearing force defined as the quotient between the shearing force and the flat cutting area,  $i, \alpha_n$  are the oblique angle and the normal rake angle, respectively,  $\phi_n$  is the normal shear or cutting angle,  $\eta$  is the chip-flow angle and  $\beta_n = \arctan(\tan \beta_A \cos \eta)$  where  $\beta_A$  is the friction angle.

Of course, the cutting coefficients described in (16) are considered constant for the tool/material as a block, and these values can be adjusted either empirically in milling operations or by using the oblique-cut transformation.

The primary forces are calculated in the direction of feed  $X$ , normal direction  $Y$  and axial direction  $Z$ , which are derived from the transformation indicated in equation (17) for the particular case of a cylindrical mill.

$$\begin{aligned} dF_{x,j}(\phi_j(z)) &= -dF_{t,j} \cos \phi_j(z) - dF_{r,j} \sin \phi_j(z) \\ dF_{y,j}(\phi_j(z)) &= +dF_{t,j} \sin \phi_j(z) - dF_{r,j} \cos \phi_j(z) \\ dF_{z,j}(\phi_j(z)) &= +dF_{a,j} \end{aligned} \quad (17)$$

Furthermore, the exact solution for cylindrical mill can be found. By substituting (14) and (15) into (12) and making  $\kappa = 90^\circ$ , we obtain:

$$\begin{aligned} F_{x,j}(\phi_j(z)) &= \left\{ \begin{array}{l} \frac{s_{ij}}{4k_\beta} [-K_{tc} \cos 2\phi_j(z) + K_{rc} [2\phi_j(z) - \sin 2\phi_j(z)]] \\ + \frac{1}{k_\beta} [K_{te} \sin \phi_j(z) - K_{re} \cos \phi_j(z)] \end{array} \right\}_{z_{j,1}(\phi_j(z))}^{z_{j,2}(\phi_j(z))} \\ F_{y,j}(\phi_j(z)) &= \left\{ \begin{array}{l} \frac{-s_{ij}}{4k_\beta} [K_{tc} (2\phi_j(z) - \sin 2\phi_j(z)) + K_{rc} \cos 2\phi_j(z)] \\ + \frac{1}{k_\beta} [K_{te} \cos \phi_j(z) - K_{re} \sin \phi_j(z)] \end{array} \right\}_{z_{j,1}(\phi_j(z))}^{z_{j,2}(\phi_j(z))} \\ F_{z,j}(\phi_j(z)) &= \frac{1}{k_\beta} [K_{ac} s_{ij} \cos \phi_j(z) - K_{ae} \phi_j(z)]_{z_{j,1}(\phi_j(z))}^{z_{j,2}(\phi_j(z))} \end{aligned} \quad (18)$$

where  $z_{j,1}(\phi_j(z))$  and  $z_{j,2}(\phi_j(z))$  are the lower and upper limits, respectively, that establish the axial depth of cut at lip  $j$  of the mill.

## SIMULATION AND MODEL VALIDATION

The algorithms were implemented in MATLAB, drawing upon the mathematical models. MATLAB is the abbreviated name for “MATrix LABoratory.” MATLAB is a software tool for doing numerical computations with matrices and vectors (Matlab 2003). It can also display information graphically and includes many toolboxes for several research and applications areas.

MATLAB was chosen for its simplicity of programming, the possibility it affords of running simulations and applications in real time and the portability of the programs that are developed with its use (i.e., it is possible to generate C/C++ programs from MATLAB files).

Through the simulation based on these models, studies may be conducted to gain an understanding of the influence the variables and parameters have. The main difficulties are choosing the cutting coefficients and the properties of the materials, which were taken from earlier work on the subject (Altintas 2000). As mentioned above, this paper is not an attempt to develop a methodology for finding such coefficients, but an attempt to use the information already available in the literature.

The tool type considered (cylindrical mill) is one of the tools most frequently used in machining molds and equipment in the construction of parts for the aerospace and automobile industry.

The workpiece-material properties that were used for the simulation in the MATLAB environment are the properties of GGG-70 cast iron with nodular graphite. In this study, in the simulation and in the real tests, the cutting condition for high-speed milling operations were regarded:  $V_c=640$  m/min,  $sp=17500$  rpm,  $f=2700$  mm/min,  $a_p = 0.5$  mm,  $a_e = 0$ ,  $a_s = 12$  mm,  $\theta=30^\circ$ ,  $H = 25.0$  mm,  $D = 12.0$  mm.

The constants used in the simulation and in the experimental validation were  $K_{tc} = 2172$  N/mm<sup>2</sup>,  $K_{rc} = 850$  N/mm<sup>2</sup>,  $K_{te}=17.3$  N/mm,  $K_{re} = 7.8$  N/mm,  $K_{ac} = 726$  N/mm<sup>2</sup>,  $K_{ae} = 6.7$  N/mm. These constants or cutting coefficients referring to the material and the tool were drawn from the available literature, due to their similarity to the characteristics of the tool/material set-up used in the study in question.

### Experimental validation

The validation tests were conducted at a KONDIA HS1000 high-speed machining center equipped with a Siemens 840D open CNC. Cutting force was measured using a Kistler 9257 dynamometric platform installed on the testbed. The main technical specifications of the Kistler platform are: a [-5, 5] kN measurement range

along each axis, a natural frequency of  $>4$  kHz, a full-scale linearity of  $\leq 2\%$  and a sensitivity of  $-7.5$  pC/N. Three 5011 load amplifiers were used as well, one each to measure the cutting forces  $F_x$ ,  $F_y$  and  $F_z$ . Measurement was done by means of a DAQBOARD-2005 data-acquisition card at a sampling frequency of 50 kHz.

A Karnasch 30.6472 cylindrical mill 12 mm in diameter was selected to be used in validating the model developed under the procedure described herein. The tool was 12% cobalt micrograin with a 1-3-micra coating of TiAlN-Al, microhardness 3300 (HV 0.05), withstanding a maximum temperature of 800°C. The chosen test piece, measuring 200x185x50 mm, was made of GGG-70 iron and was machined in a spiral pattern. The real cutting conditions chosen were the same as considered above for the simulation. A view of the tool used in the tests (a), the test piece and the configuration of the cut (b) and the laboratory (c) may be seen in figure 2.

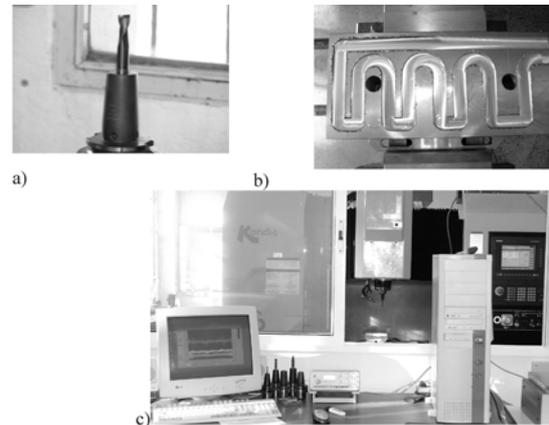


Figure 2: Cutting tool for experiments, b) experimental workpiece, c) partial view of the Laboratory for machine tool research

The objectives focused not only on validating the theoretical model for the cylindrical mill, but also on ascertaining the importance of cutting forces in HSM and their ability to provide relevant information on the condition of the cutting tool during high-speed cutting processes. Figures 3, 4, 5 and 6 show the real behavior of cutting forces  $F_x$ ,  $F_y$  and  $F_z$  and the resulting force  $F_{qT}$  for each of the four cases analyzed. The model's response is shown as a solid line.

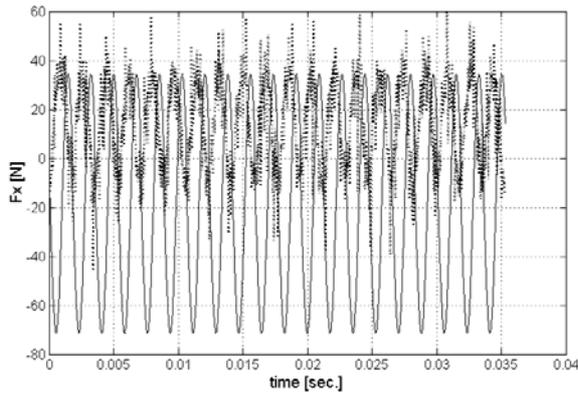


Figure 3: Measured (straight line) and predicted (dashed line) cutting force  $F_x$

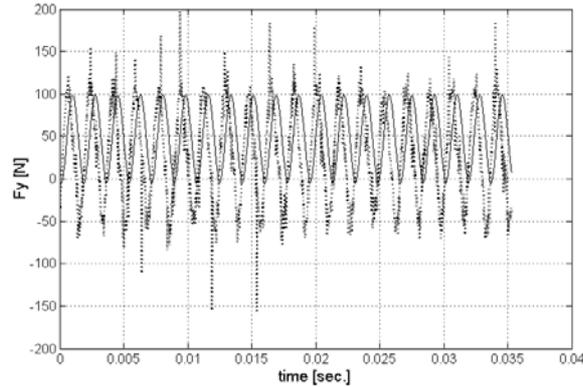


Figure 4: Measured (straight line) and predicted (dashed line) cutting force  $F_y$

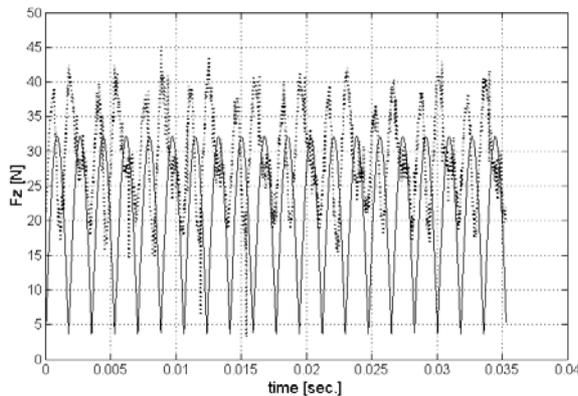


Figure 5: Measured (straight line) and predicted (dashed line) cutting force  $F_z$

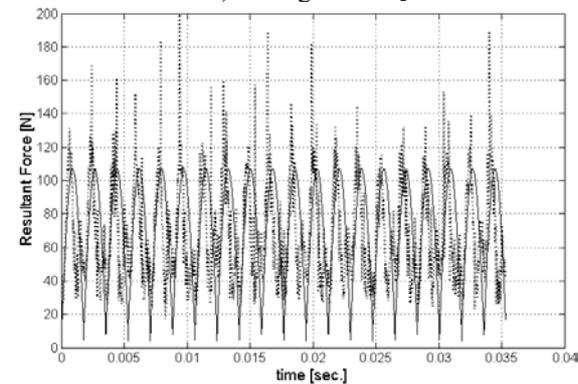


Figure 5: Measured (straight line) and predicted (dashed line) resultant cutting force

The average resulting cutting force  $\bar{F}_e$  estimated by the model is 70.6N and the average resulting cutting force  $\bar{F}_{qT}$  measured in a real high-speed cutting operation is 71.7N. The error criterion  $\bar{E} = \frac{(\bar{F}_{qT} - \bar{F}_e) \cdot 100}{\bar{F}_{qT}}$ , is 7.9%.

## CONCLUSIONS

This paper reports some initial results in modeling the high-speed cutting process and the validation of the model in question. The mathematical model has two fundamental, co-dependent parts. The first is a multiple-input system that defines the model's kinematics, where constants and variables describe the tool geometry, the material type and the cutting parameters. The second addresses the dynamics, represented by integral-differential equations. The exact analytical solution is found, inasmuch as the limits of integration and the boundary conditions along the tool geometry can be pre-established.

The flexibility of the model developed in this paper makes simulation studies valid for both conventional and high-speed machining, maintaining a consistent relationship among cutting parameters such as feed, spindle speed and cutting speed.

Furthermore, the importance of cutting force as a representative variable of high-speed machining processes has been proved. Thus, what cutting conditions will maximize the chip-removal rate or useful lifetime of a specific tool can be evaluated by comparing average cutting-force values. In this way, improvements in machining time, useful tool lifetime and waste-product reduction can be achieved. It is also possible to obtain information necessary for estimating tool wear. Whether cutting-force figures provide enough information about the condition of the cutting tool is a matter that should be analyzed in the future.

With this paper and the implementation conducted using MATLAB, the doors are thrown open onto the use of a model for predicting surface finish and stability lobes in high-speed machining processes.

## REFERENCES

- Altintas Y.; 2000. *Manufacturing Automation: metal cutting mechanics, machine tool vibrations, and CNC design*, Cambridge University Press, USA.
- Altintas Y. and P. Lee. 1998. "Mechanics and dynamics of ball end milling forces." *Journal of Manufacturing Science and Engineering- Trans. ASME* 120, 684-692.
- Budak E.; Y. Altintas; E.J.A. Armarego. 1996. "Prediction of milling force coefficients from orthogonal cutting data." *Journal of Engineering of Industry* 118, 216-224.

- Bukkapatnam S.T.S; A. Lakhtakia; S.R.T. Kumara. 1995. "Analysis of sensor signals shows turning on a lathe exhibits low-dimensional chaos." *Physics Review E* 52, No.3, 2375-2387.
- Engin S. and Y. Altintas. 2001. "Mechanics and dynamics of general milling cutters Part I: helical end mills." *Int. Journal of Machine Tools and Manufacturing* 41, 2195-2212.
- Fu H. J.; R.E. Devor; S.G. Kapoor. 1984. "A mechanistic model for the prediction of the force system in face milling operation." *Journal of Engineering of Industry* 106, No. 1, 81-88.
- Haber R. E.; J.E. Jiménez; J.L. Coronado; A. Jiménez. 2004a. "Modelo matemático para la predicción del esfuerzo de corte en el mecanizado a alta velocidad.", *Revista Metalurgia Madrid* 40, No. 4, 247-258.
- Haber R. E.; J.E. Jiménez; C.R. Peres; J.R. Alique. 2004b. "An investigation of tool wear monitoring in a high-speed machining process." *Sensors and Actuators A: Physics* 116, 539-545.
- The Math Works Inc. Users manual Matlab 6.5 Release 13. 2003. Natick, Massachusetts, USA.
- Yucesan G. and Y. Altintas. 1996. "Prediction of ball end milling forces." *Journal of Engineering of Industry* 1, No.1, 95-103.



**ANGEL ALIQUE** Angel Alique received the B.Sc. And Ph.D. degree in physics from University Complutense of Madrid in 1972 and 1979, respectively. He is currently researcher at Instituto de Automatica Industrial (CSIC). He has published several technical

papers on supervisory fuzzy control and modelling of machining processes. He is currently the main researcher of two projects funded by CICYT.

His research interests include fuzzy modelling and control, robotics systems and intelligent control



**RODOLFO E. HABER** was born in Santiago de Cuba, Cuba, in 1969. He received his bachelor's degree with first-class honors in automatic control engineering from Cuba's Oriente University in 1992. In 1995 he received a Ph.D. grant supported by the Spanish Office for Scientific Cooperation (AECI)

in Spain. He earned a Ph.D. cum laude in industrial engineering from Madrid Polytechnic University (UPM) in 1999. He joined the Spanish Council for Scientific Research (CSIC) in 1999, where he has worked on a number of research and development projects. Since 2002 he has belonged to IFAC's Technical Committee 3.1 *Computers&Control*. He has published a number of technical papers in specialized journals and several chapters of books. His research interests include complex electromechanical processes, sensing systems, control theory and applications,

hardware/software solutions, soft-computing techniques, classical and adaptive control and supervisory control.



**JOSE R. ALIQUE** José R. Alique received the B.Sc. degree in physics from the University Complutense of Madrid (UCM), Spain in 1969. He received the M.Sc. and Ph.D. degree of physics sciences from International Institute of Philips,

North Holland, in 1971, and University Complutense in 1973. From 1969 to 1970, he was Assistant Professor of Physics and Automation at UCM. He has been working in automation since 1972 in the Spanish Council for Scientific Research (CSIC). From 1991-96, he was the Director of the Technology Transfer Office in the CSIC. From 1996 to 2001, he was the vice director of R+D projects in the Ministry of Science and Technology. He is currently researcher of Instituto de Automatica Industrial (CSIC) and Head of the Computer Science Department.

# MODELLING, MONITORING AND CONTROLLING ELECTROLESS NICKEL PLATING PROCESS OF PLATED THROUGH HOLE BOARDS

Kalle Kantola  
Control Engineering Laboratory  
Helsinki University of Technology  
P.O. Box 5500, FIN-02015 Espoo, Finland  
E-mail: kalle.kantola@hut.fi

## KEYWORDS

Electroless, model, nickel, plating, control.

## ABSTRACT

In this paper the main points of electroless nickel immersion gold (ENIG) surface finish are discussed. To increase the final quality of the surface, a mathematical model for electroless nickel plating process – a part of ENIG-surface finish – is proposed. Based on the model an online monitoring and a new control strategy are proposed. The methods are evaluated by simulations and it is shown that the most important deposition parameters can be controlled and the effect of the process perturbation can be eliminated with the proposed methods. Finally the benefit of the proposed methods is discussed.

## INTRODUCTION

Electroless nickel plating is a widely used process in many microelectronic applications including plated through hole boards (PTH) manufacturing (Mallory 1990, Rohan et al. 2002). The manufacturing process of PTH is a complex operation of different chemical and mechanical processes (U.S. Environmental Protection Agency (EPA) 1995). High production volumes, tighter tolerances and the demand for better quality have forced producers to enhance and automate their production. Because of the complex processes and lack of research the automation level is still low and the main control is manual (Nuzzi 1983, Chen et al. 2002).

The base of the PCB is a glass fibre reinforced epoxy laminate, which is laminated on both sides with a thin copper layer. The laminated copper layer works as the circuit conductor and the circuit layout is transferred to the layer by using the print-and-etch –process. (Coombs 2001)

The final step in the manufacturing process is to protect the copper circuit from oxidation by electroless nickel immersion gold finish (ENIG) (Coombs 2001). During the process the copper circuitry is plated with a 3-6  $\mu\text{m}$  thick nickel-phosphor (Ni-P) layer followed with a thin (0,05-0,15  $\mu\text{m}$ ) gold (Au) layer. Gold is used because it doesn't oxidise but dissolves into the solder, which makes it an excellent surface finish. However,

gold cannot be plated directly on copper but an electroless nickel layer is required to work as an oxidation barrier between the copper and gold layers (Rohan et al. 2002).

The ENIG-process has a very critical effect on PTH final characteristics. Especially, the Ni-layer and its phosphorous content are very crucial (Mallory 1990, Kwok et al. 2004). However, these characteristics cannot be measured during plating and afterwards quality inspections are needed. Unfortunately at that time the possible errors cannot be corrected. Thus it is crucial to control the electroless plating reaction during the process.

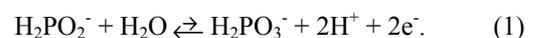
This paper introduces a sophisticated process control method for automatic control of electroless nickel plating process. The key point is a mathematical process model which estimates the critical unobservable process parameters - such as deposition rate, deposition thickness and the phosphorous content - online based on the measurable process. The aim of the control is to keep Ni-P alloy thickness and phosphorous content stable despite perturbations like changes in bath loading or effect of bath aging and prevent bath contamination.

## MODEL

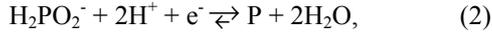
The process model was proposed and calibrated on a large sample of experimental data gathered from an industrial process in (Tenno et al. 2005a). The model was former validated using independent data in (Kantola et al. 2005) where it was shown that the model is capable to estimate the unobservable processes accurately from the online measurable processes.

Despite the intensive studies the reaction mechanism of the studied process is not known (Djokic 2002). This describes the complexity of the reaction. The developed model is based on the following electrochemical reaction mechanism proposed in (Riedel 1991, Mallory 1990, Kim and Sohn 1996).

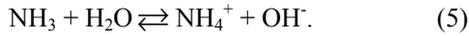
Anodic reaction - hypophosphite oxidation



Cathodic reactions - phosphorous deposition, hydrogen evolution, nickel deposition



In the reaction the hydrogen ion formation exceeds consumption and ammonia is used for pH-index balancing



### Electro-Chemical Cell Model

The model is formulated as a two-directional electrode model for reactions (1)-(4). The current densities of the reactions are calculated from the Butler-Volmer equation and concentration controlled through empirical formulas which accelerate or limit reaction rates according to reactant concentration.

$$i_n = i_{0n}\mu_n \{ \exp(v\alpha_{an}p_nk\eta_n) - \exp(-v\alpha_{cn}p_nk\eta_n) \}, \quad (6)$$

where

$n$  - reaction number: 1 - oxidation, 2 - P deposition, 3 - hydrogen evolution, 4 - Ni deposition,

$i_n$  - current density, A/cm<sup>2</sup>,

$i_{0n}$  - exchange current density, A/cm<sup>2</sup>,

$$i_{01} = 18, \quad i_{02} = 0.5, \quad i_{03} = 2.5, \quad i_{04} = 1.6 \text{ mA/cm}^2,$$

$\eta_n$  - overpotential,  $\eta_n = \phi - v_n$ ,

$\phi$  - mixed potential, V,

$v_n$  - thermodynamic equilibrium potential, V,

$\mu_n$  - dimensionless concentration of species,

$v$  - robustness coefficient, 0.1,

$k$  - temperature voltage,  $k = F/RT$ , 1/V,

$T$  - temperature, K,

$R$  - universal gas constant, 8.3145 J/mol-K,

$F$  - Faraday's constant, 96487 C/mol,

$p_n$  - number of exchanged electrons,  $p_1 = 2$ ,  $p_2 = 1$ ,

$\alpha_{an}$  - anodic apparent transfer coefficients,  $\alpha_{an} + \alpha_{cn} = 1$ ,

$\alpha_{cn}$  - cathodic apparent transfer coefficients,

$$\alpha_{a1} = 0.53, \quad \alpha_{c2} = 0.62, \quad \alpha_{c3} = 0.59, \quad \alpha_{c4} = 0.47.$$

Because of the current conservation the anodic current density is equal to the sum of cathodic current densities

$$i_1 + i_2 + i_3 + i_4 = 0. \quad (7)$$

The mixed potential is a measured parameter in (Tenno et al. 2005a); as it should be when estimating the model parameters. In online use the mixed potential can be adjusted to satisfy (7) as is the case in this paper.

The equilibrium potentials of the reactions are calculated from Nernst equation using unit activity for solid material (deposited nickel and phosphorous) and water, and unit partial pressure (1 atm) of hydrogen as follows

$$v_1 = U_1 + \kappa(\log c_2 - \log c_1 - 2\text{pH}), \quad (8)$$

$$v_2 = U_2 + 0.2\kappa(\log c_1 - 2\text{pH}), \quad (9)$$

$$v_3 = U_3 - 2\kappa \text{pH}, \quad (10)$$

$$v_4 = U_4 + \kappa \log c_4, \quad (11)$$

where  $\kappa^{-1} = 2k \log e$ ,

$c_i$  - ion concentration of species ( $c_1$  - hypophosphite,  $c_2$  - orthophosphite or  $c_4$  - nickel), mol/dm<sup>3</sup>.

### Alloy Deposition

The deposition rate of nickel and phosphorous are calculated according to current densities of respective reactions while the Ni-P-alloy film is composed by superposition of these  $x_a = x_{\text{Ni}} + x_{\text{P}}$ .

$$\frac{dx_{\text{Ni}}}{dt} = -i_4 \frac{M_{\text{Ni}}}{2F\rho_{\text{Ni}}}, \quad \frac{dx_{\text{P}}}{dt} = -i_2 \frac{M_{\text{P}}}{F\rho_{\text{P}}}, \quad (12)$$

where

$t$  - immersion time, sec,

$x_{\text{Ni}}$  - partial thickness of nickel, cm; physically, this is a ratio between volume of nickel and plate area,

$x_{\text{P}}$  - partial thickness of phosphorous, cm,

$x_a$  - thickness of the Ni-P-alloy film, cm,

$i_4$  - current density of nickel deposition reaction, A/cm<sup>2</sup>,

$i_2$  - current density of phosphorous deposition reaction, A/cm<sup>2</sup>,

$M_{\text{Ni}}$  - molecular weight of nickel, 58.7 g/mol,

$M_{\text{P}}$  - molecular weight of phosphorous, 31 g/mol,

$\rho_{\text{Ni}}$  - nickel density, 8.9 g/cm<sup>3</sup>.

$\rho_{\text{P}}$  - phosphorous density, 1.82 g/cm<sup>3</sup>.

The phosphorous content in the Ni-P-alloy film can be expressed as a volumetric ratio between the overall and partial thickness  $P_{\text{vol}} = (x_{\text{P}}/x_a) 100\%$  or as a weight ratio  $P_{\text{wt}}$  between the overall and partial weights.

### Concentration Dynamics

The concentration dynamics of species (nickel, hydrogen, hypo- and orthophosphite, etc.) can be calculated from the model according to mass balance as follows.

#### *Hypophosphite feeding and consumption*

$$dc_1 = [A(2i_2 - i_1)/2F + Q_{1f}c_{1f}]dt + \sigma_1 dW, \quad c_1(t_0) = c_{1\text{ref}},$$

where

$c_1$  - hypophosphite concentration, mol H<sub>2</sub>PO<sub>2</sub><sup>-</sup>/dm<sup>3</sup>,

$c_{1f}$  - feeding solution concentration, mol H<sub>2</sub>PO<sub>2</sub><sup>-</sup>/dm<sup>3</sup>,

$Q_{1f}$  - hypophosphite dilution rate, 1/s,

$t_0$  - makeup moment for a newly made bath, sec,

$t$  - elapsed time, sec,

$\sigma_1$  - model inaccuracy,

$W$  - model-prediction error: Wiener process.

#### *Nickel feeding and consumption*

$$dc_4 = [A i_4/2F + Q_{4f}c_{4f}]dt + \sigma_4 dW, \quad c_4(t_0) = c_{4\text{ref}}, \quad (13)$$

where

$c_4$  - nickel concentration, mol  $\text{Ni}^{2+}/\text{dm}^3$ ,  
 $c_{4f}$  - feeding concentration, mol  $\text{Ni}^{2+}/\text{dm}^3$ ,  
 $Q_{4f}$  - nickel dilution rate, 1/s.

*Hydrogen formation and removal with ammonia*

$$dc_3 = [A(i_1 + 2i_2 + i_3)/F]dt - dx + \sigma_3 dW, \quad c_3(t_0) = c_{3\text{ref}}, \quad (14)$$

where

$c_3$  - hydrogen concentration, mol  $\text{H}^+/\text{dm}^3$ ,  
 $dx$  - dissociation-consumed hydrogen, mol  $\text{H}^+/\text{dm}^3$ .

The following approximation is used for control. It is valid for a steady-state process of ammonia feeding and consumption.

$$dx = Q_{3f}c_{3f}dt,$$

where

$c_{3f}$  - feeding concentration, mol  $\text{NH}_3/\text{dm}^3$ ,  
 $Q_{3f}$  - ammonia dilution rate, 1/s.

*Orthophosphite accumulation*

$$dc_2 = Ai_1 dt/2F + \sigma_2 dW, \quad c_2(t_0) = 0,$$

$c_2$  - orthophosphate concentration, mol  $\text{H}_2\text{PO}_3^-/\text{dm}^3$ .

Orthophosphite accumulation can be used to derive the bath age, meaning the accumulation of by product. The bath age can be presented as metal-turn-over (MTO) meaning how many time the bath has consumed its initial nickel.

The Wiener processes are introduced to show the model inaccuracy that should be accounted for in development of the estimation and control strategy insensitive to uncertainty.

## Measurements

The needed online measurements and the model provided estimates are shown in Figure 1. All measurements (nickel ion concentration, pH, plating time, bath temperature, loading and refreshment flows) are part of the normal process control system of the plating line.

The bath loading is defined as a total metal area of plates immersed in the solution per bath volume and the state of bath loading is equal to the number of baskets immersed.

$$A = A_p z, \quad (15)$$

where

$A$  - bath loading,  $\text{cm}^2/\text{dm}^3$   
 $z$  - loading level,  $z = Z/N$ ,  
 $Z$  - state of loading,  
 $N$  - maximum number of baskets (rack for plates) in use,  $N = 3$ ,  
 $A_p$  - maximum loading for all baskets immersed (a product constant),  $\text{cm}^2/\text{dm}^3$ .

The state loading is a finite-state Markov process defined by the down loading rate up and up loading rate; it is a completely observable process in this paper.

From these measurements the critical alloy thickness and phosphorous content are calculated along with reagent- and by-product concentration later exploited in the monitoring and control applications.

The accuracy of the proposed model was validated using independent data gathered from industrial production line by comparing model estimated values to measured ones. The valuated accuracies for alloy thickness and phosphorous content are shown in Table 1.

Table 1: Accuracies for the proposed Model

Parameter	Admissible range	Accuracy %
Film thickness	2.5 - 5 $\mu\text{m}$	9-17
Phosphorus	7 - 10 %	10-18

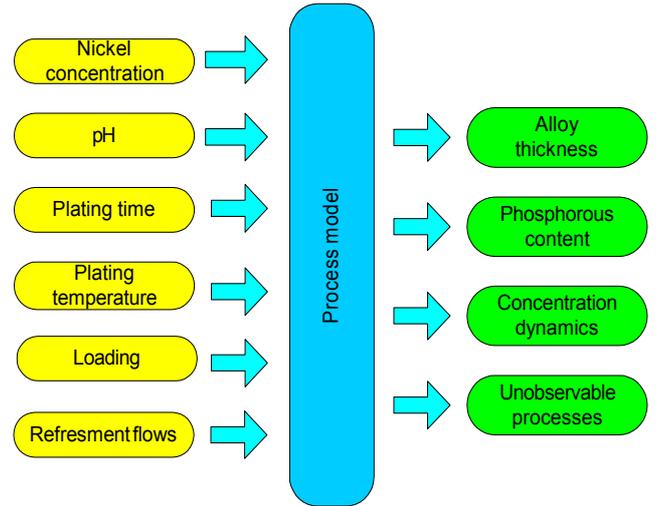


Figure 1: The Needed online Measurements of the model and produced Online Estimates

## ONLINE MONITORING

The developed model can be exploited in several ways in process control. Simplest is an online monitoring system where it provides critical information about the unobservable process online to operator as illustrated in Figure 2.

In the online monitoring concept the developed process model is used to estimate the needed process parameters (alloy thickness, phosphorous content) according to online measurable parameters (pH, nickel ion concentration, temperature, loading and plating time).

The monitoring system only supports the operator's actions taking no part to the control actions.

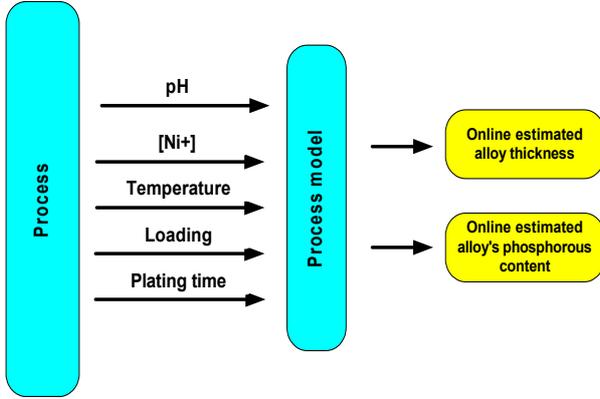


Figure 2: Schematics of the online Monitoring Concept

### AUTOMATIC BATH CONTROL

A more sophisticated application is an automatic control algorithm. It has been shown in (Tenno et al. 2005b) that the alloy thickness and phosphorous content can be controlled by controlling the pH and nickel ion concentration of the bath.

The main perturbations for the process are the bath loading (15), hydrogen formation and bath aging. These have great effects on process dynamic. To keep the process stable the loading should be kept as constant as possible, excess hydrogen should be neutralised and bath aging compensated by increasing pH-index. Unfortunately changes in loading cannot be avoided in industrial PTH production and control has to be used to eliminate this perturbation. The control algorithm has to calculate the appropriate pH-index according to bath age or MTO. To prevent contamination, the process has to be constantly balanced as discussed in following.

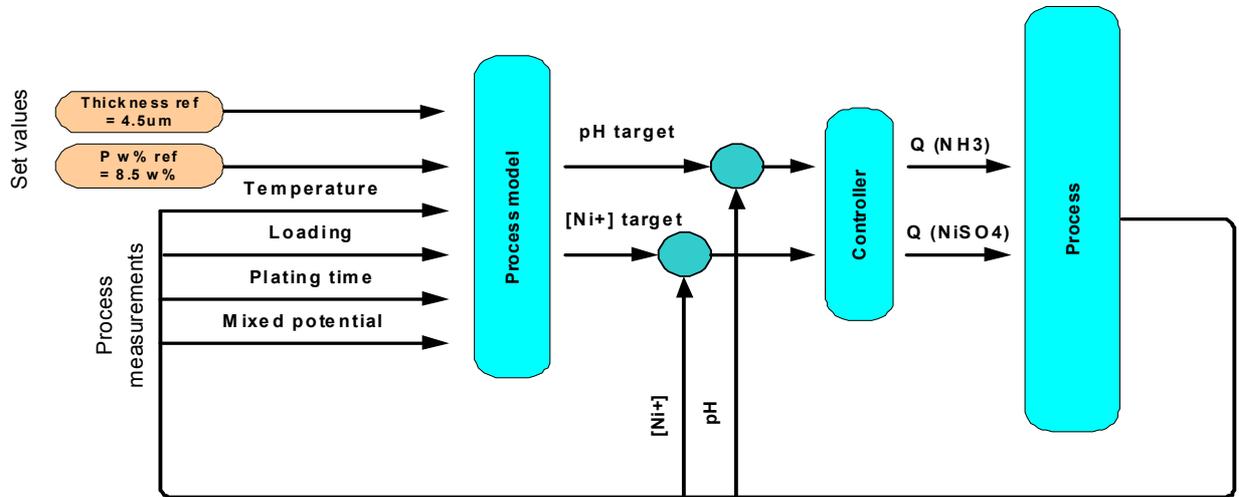


Figure 3: Schematics of the Proposed Control Algorithm

The proposed control concept is roughly depicted in Figure 3. The desirable values for the alloy thickness and phosphorous content are given to the process model along with available online measurements. According to this the model calculates trajectories for the bath's pH and nickel ion concentration and a controller adjust refreshment pumping so that bath parameters achieve the calculated trajectories.

### Optimal Trajectory

The desired thickness of Ni-P-alloy film is a  $x_a = 4 \mu\text{m}$  and phosphorous content  $P_{wt} = 8.5 \%$ . The partial thicknesses of the nickel and phosphorous can be calculated as target values for control from these desired values

$$x_{Ni}^* = x_a - x_P^*, \quad x_P^* = x_a P_{vol}/100. \quad (16)$$

These target values can be achieved and the contamination is avoided if the reaction is continuously balanced. This means that the pH-index, nickel percentage and mixed potential should satisfy the charge conservation requirement (7) involving the electrode reactions (6) and the thermodynamic equilibrium potentials (8)-(12). It also includes the deposition rate (12) and target (16) requirements, coupled in the relationship

$$-\frac{i_4}{C_{Ni}} = \frac{x_{Ni}^*}{\tau}, \quad -\frac{i_2}{C_P} = \frac{x_P^*}{\tau},$$

where

$x_{Ni}^*$  - target thickness of nickel, cm,

$x_P^*$  - target thickness of phosphorous, cm,

$\tau$  - plating time (22 min), s,

$C$  - charge density of Ni- and P-ions,  $C/\text{cm}^3$ .

Numerical minimisation of the function  $f = f\{i(c_3, c_4, \phi)\}$

$$f = (i_1 + i_2 + i_3 + i_4)^2 + (i_2 - C_P \frac{x_P^*}{\tau})^2 + (i_4 - C_{Ni} \frac{x_{Ni}^*}{\tau})^2 \quad (17)$$

is the simplest method to calculate the balanced processes for the hydrogen and nickel concentration and for

the mixed potential. The target values are coupled processes (meaning implicit use of the mixed potential); they depend on the hypo- and orthophosphate concentrations

$$\zeta_3 = c_3(c_1, c_2), \quad \zeta_4 = c_4(c_1, c_2), \quad \phi = \phi(c_1, c_2). \quad (18)$$

The plating process is well controllable as long as the balance between the target values holds.

### Tracking Control

The tracking control problem can be approximately solved as a minimum variance problem. The model (13), (14) is linear in state and control variables for fixed current densities and nonlinear for free current densities; however it is weakly depended on the state variables through the almost invariant current densities. In this situation, the following feed-forward PI-control is justified and is a logical successor to the existing control in PTH board processes.

$$u_t = A_p z_t v - K_p [\xi_t - \zeta_t + T_i^{-1} \int_0^t (\xi_s - \zeta_s) ds],$$

where

$\xi$  - measured hydrogen and nickel concentrations,  $\xi = [\xi_3, \xi_4]^T$ ,

$\zeta$  - target calculated from minimization of the criterion (17) as parameters  $\zeta = [\zeta_3, \zeta_4]^T$  dependent (18) on the hypo- and orthophosphate concentrations,

$u$  - controls: ammonia and nickel (hypophosphite) feeding rates,  $u = [Q_{3f}, Q_{4f}]^T$ ,

$A_p$  - maximum loading,

$z_t$  - bath loading level (observable process),

$v$  - control setpoint for average bath loading,

$K_p$  - control gain: diagonal matrix of weights,

$T_i$  - integration time.

This control depends on the model through the target. As a feed-forward control, it compensates the bath loading changes.

Although the operator does not need to take a part in control and the possibility of human error is avoided the reliability of the method can be guaranteed by random laboratory analysis.

### SIMULATIONS

The proposed control algorithm was tested by simulation shown in Figures 4-8. During the simulation the reference values for alloy thickness and phosphorous content were changed according to sequence shown in Figures 5 and 6. The bath loading process was simulated as Markov pure jump process shown in Figure 4. Every process was also exposed to Wiener process with 5% noise-to-signal ratio.

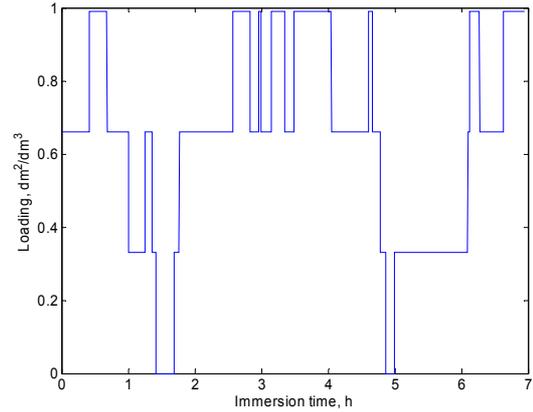


Figure 4: Simulated Bath Loading

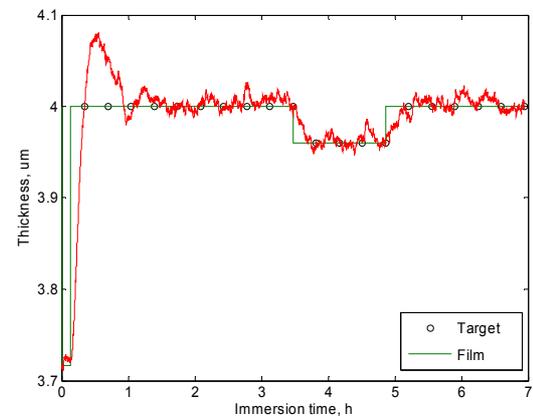


Figure 5: Reference and Simulated alloy Thicknesses

As can be seen from Figures 4 and 5 the alloy thickness and phosphorous content follows the given reference values accurately without any effect of loading perturbations. In Figures 7 and 8 it can be seen that the control is able to keep the control parameters close to the optimal trajectories which verifies the control capability to avoid contamination affected by vigorous parameter changes.

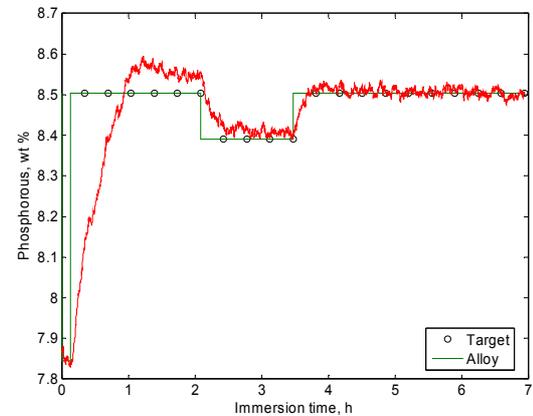


Figure 6: Reference and Simulated Phosphorous Content

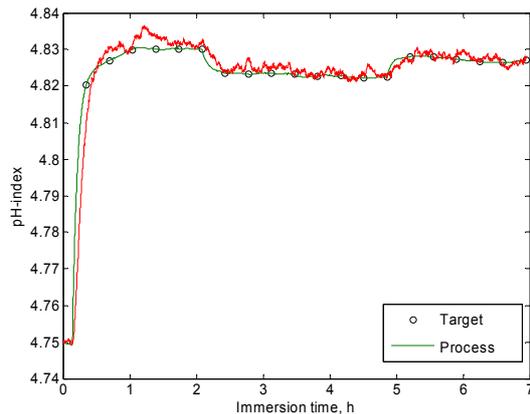


Figure 7: Model Calculated Optimal Trajectory for pH-index during simulation

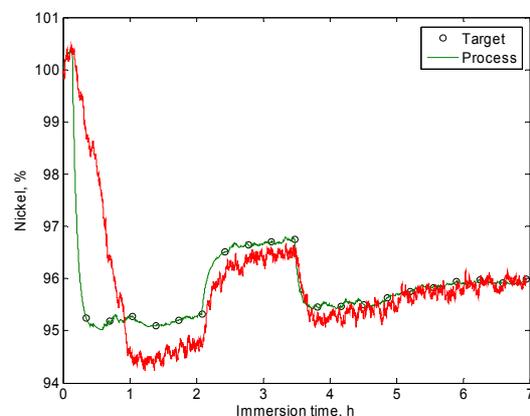


Figure 8: Model Calculated Optimal Trajectory for  $\text{Ni}^{2+}$ -ions

## BENEFITS

The obvious benefits, from the proposed online monitoring and control concepts, are improved quality and thus reduction of cost of bad quality. It also makes expensive measurements like phosphorous content analysis needless.

The electroless nickel process is not the only process where the proposed control concepts can be used. There are several similar chemical processes in only PTH manufacturing where the proposed control concepts can be used.

## CONCLUSION

Electroless nickel plating process has a critical role in printed circuit board manufacturing process. Especially thickness and phosphorous content of the alloy affect final quality of PCB. The controlling of these parameters in traditional methods is uncertain and ineffective.

A sophisticated monitoring and control concept was proposed. The online monitoring is based on electro-

chemical process model. It helps an operator to estimate online the development of the critical alloy parameters.

The proposed control algorithm uses the developed process model to calculate optimal nickel ion concentration and pH trajectories according to the given reference values. The accuracy of the algorithm was evaluated by simulations and shown to be functional.

The concrete benefits of the methods are increased quality and reduced costs of laboratory analysis and quality inspections.

## REFERENCES

- Chen, L., Crnic, M., Lai Z. and Liu, J., 2002, "Process development and Adhesion Behaviour of Electroless Copper on Liquid Crystal Polymer (LCP) for Electronic Packaging Application", *IEEE Transactions on Electronics Manufacturing*, Vol 25. No. 4. p. 273-278.
- Coombs, C. F. Jr., 2001, *Printed circuit Handbook*, 5<sup>th</sup> ed. New York, United States; McGraw-Hill. ISBN 0-07-135016-0. p. 26.1-26.13.
- Djokic, S. S., 2002, "Electroless Deposition of Metal and Alloys", *Modern Aspects of Electrochemistry No. 35*, Kluwer Academic/Plenum Publishers, p. 51-131.
- Kantola K., Tenno R. and H. Koivo., 2005, "Model for electroless nickel plating process of through holes board process: model accuracy testing". In *Proc. 5<sup>th</sup> IASTED Conf. on Modelling, Simulation and Optimization*, Aruba, (submitted).
- Kim Y.-S. and H.-J. Sohn., 1996, "Mathematical modelling of electroless nickel deposition at steady state using rotating disk electrode". *J. Electrochem. Soc.* Vol 143, No 2. P. 505-509.
- Kwok R.W.M., Chan K.C.M., Bayes M.W., 2004, "Development of an electroless nickel immersion gold process for PCB final finishes". *Circuit world* 30/3, ISSN 0305-6120, p. 37-42.
- Mallory G.O., 1990, "The fundamental aspects of electroless nickel plating" in *Electroless plating: fundamentals and applications*. Ed. Mallory G.O., Hajdu J.B. AESF Society. Orlando. P. 1-56.
- Nuzzi, F. J., 1983, "Automatic control of Electroless nickel Plating" in *Metal Finishing*. (May), p. 31-33.
- Riedel W., 1991, "*Electroless Nickel Plating*". Finishing Publications LTD, Stevenage, Hertfordshire, England, 1991.
- Rohan J., O'Riordan G., Bordman J., 2002, "Selective electroless nickel deposition on copper as a final barrier/bonding layer material for microelectronics applications", *J. Applied Surface Science*. 185, p. 289-297.
- Tenno R., Kantola K., Koivo H. and A. Tenno. 2005, "Model for electroless nickel plating through hole boards". *IFAC J. Control engineering practice*,. (Submitted)
- Tenno R., Kantola K., Koivo H., 2005, "Electroless nickel plating: Bath control". In *Proc. 16<sup>th</sup> IFAC Conf.*, (In print)
- U.S. Environmental Protection Agency (EPA), 1995, *Printed Wiring Board Industry and User Cluster Profile: Printed Wiring Board Project*. EPA (EPA/744-R-95-005), EPA Office of Pollution Prevention and Toxic, Washington D.C. (Sept.)

# EXPERIENCES ON UTILISING PLANT SCALE DYNAMIC SIMULATION IN PROCESS INDUSTRY

Jean-Peter Ylén, Matti Paljakka, Tommi Karhela, Jouni Savolainen and Kaj Juslin  
VTI, Technical Research Centre of Finland, P.O.Box 1301, FIN-02044 VTT, Finland

E-mail: peter.ylen@vtt.fi, matti.paljakka@vtt.fi, tommi.karhela@vtt.fi, jouni.savolainen@vtt.fi, kaj.juslin@vtt.fi

## KEYWORDS

Simulation, Process industry, Industrial application

## ABSTRACT

This paper will consider the role of simulation in process industry and major obstacles to adopting new technologies. The benefits of process simulators are illustrated with five practical applications. These applications were realised with APROS simulator, the structure of which is explained very briefly.

## SIMULATION IN PROCESS INDUSTRY

The recent advances in the computer technology have made it possible to perform complex calculations efficiently and even faster than real time. Complicated dynamic equations can be simulated and the results used in model predictive control systems, on-line process optimization, fault diagnosis and production planning.

A good process simulator makes dynamic simulation a powerful tool throughout the complete life cycle of a plant: from pre-engineering and commissioning planning to operator training and troubleshooting. These different areas are illustrated in Fig. 1.

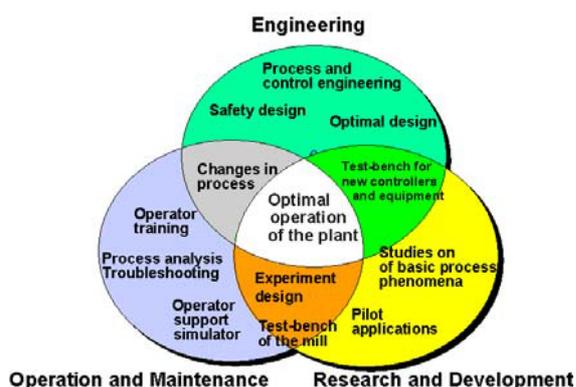


Figure 1: Role of a simulator in process industry

In a typical chemical plant, for example, the behaviour of individual reactors, distillation columns, and pumps can be monitored continuously, and process parameters changed automatically in real time to maintain desired setpoints. The behaviour of the entire plant is also monitored and setpoints can be altered in order to maximize profit on a daily or even hourly basis in

response to variations in feedstock and product prices and utility costs.

In scope of all that is said above, process industry in general is playing by the ear, when it comes to the critical area of forecasting and maintaining production from a field. In many cases the plant is considered as a 'black box' with heuristic rules of thumb. Production data is rarely compared with expectations from simulations done in the early stages of field development, and problem areas are often treated on an ad hoc, individual basis without considering the influence of the other areas. The consequences of individual actions of operators are seldom linked to surprising and unwanted side-effects – especially if the distance of side-effects from operating point is far in time and space.

The main reason for flying blind by not using advanced e.g. modeling and simulation techniques is the “if it’s not broken – don’t fix it” syndrome. Adopting new technologies requires work and other resources, which has to be justified. The plant and its processes have been running for a long time and as long as the plant is running there is no need to fix it. The question whether the plant is running optimally and how significant improvements in quality, operation and profit could be made by new technologies is seldom raised.

Another major factor contributing to the reluctance to adopt simulation aided technologies is the established view that modelling and simulation is something extra that is nice to know but not necessarily needed. In planning, design and improvement projects modelling and simulation is not included in the project plan and consequently does not have any budget.

Even people who are familiar with simulation and use it to their benefit often consider simulation to be best suited for small subsystems, e.g., one chemical reactor but not applicable in large scale systems, i.e. the entire plant. With this view any optimisation routine only finds local optima and it has been shown in many cases that even though every individual component is locally optimal the entire plant can be operating badly and very far from global optimum.

Many consider the lack of tools and services to be a significant hindrance to new technologies, but in most

cases they have not been looking very hard. There are armies of simulation experts, service providers and simulation tool vendors just waiting to help plants in need – for a monetary compensation, of course. There is no such thing as a free dinner. If they are looking for a free tool just designed for their particular needs, they are correct; there is a lack of such tools.

The nature of some of the processes is less deterministic than others and in many cases there are simply no measurements on the required entities. The overall mass and energy balances can be determined but the same can not be said about the concentrations of individual reaction components. For instance, if the composition of raw materials changes considerably and there are no measurements, there is no chance of accurately predicting the behavior of the entire system. In these cases simulation is seldom used, which is a shame because in chaotic unknown systems little knowledge (in the form of simple models) can gain the most benefits.

## APROS

There are many different simulator tools for various simulation problems. In fact, any algorithm capable of integrating differential equations numerically can be used as a simulator for dynamic systems. For large scale systems in the process industry it is not usually worthwhile to start from the scratch – in these problems a modular simulator consisting of smaller pre-modelled process/equipment components is the right way to go.

APROS (Advanced PROcess Simulator) was developed by the Technical Research Centre of Finland (VTT) and Imatran Voima Co (IVO). Originally the simulation environment was built for the needs of the nuclear and conventional power industries. It can be used for training, research, automation design and process design purposes. It provides solution algorithms and model libraries for the simulation of different flow and heat transfer processes. The simulation system can be defined by using the APROS command interpreter or the graphical interface Grades. A view to Grades is presented in Fig. 2.

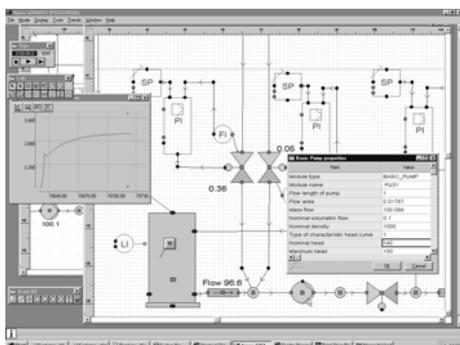


Figure 2: Grades graphical interface for APROS

The calculation is based on physical principles and empirical correlations. The network oriented solvers of APROS are table-driven, and accordingly, no programming, compilation or linking is needed during model development and simulation runs.

Aprós can be furnished with generic model libraries for several types of processes such as:

- Combustion power plants – Aprós Combustion
- Nuclear Power plants – Aprós Nuclear
- Pulp and paper mills – Aprós Paper

Aprós supports the use of dynamic simulation in all different phases during the life span of a process plant, avoiding unnecessary data transfer and reconfiguration of the simulation model. Aprós enables the use of an engineering simulator application as a basis for a training simulator. Once the process simulation model has been completed in the design phase it can be re-used with DCS (Distributed Control System) as a checkout and operator training tool in a cost-effective way.

## INDUSTRIAL APPLICATIONS

There are five different industrial applications presented in this paper. Two applications describe training simulators for Mälär Energi ab in Sweden and Suomenoja power plant in Finland. One application describes a case in which the simulator was used for automation testing and control design in Narva power plant, one presented application shows a grade change optimisation in a Stora Enso paper board mill and another application a safety analysis in Loviisa Nuclear power plant – both in Finland.

### Narva Power Plant

The Estonian power plant at Narva is an oil shale combusting power plant that consists of 8 units each with two boilers and one steam turbine. Each unit has an electric output of 200 MW. The plant is of Soviet make dating to the 1970's.

In 1999 - 2000 Fortum Engineering carried out a modernisation project of one unit, in parallel to which a full-scale dynamic simulation model on the unit was built. The process was modelled using APROS, and the automation was implemented in the virtual metsoDNA environment, all running in Windows NT environment on standard PCs. The systems were connected through their OPC interfaces. At the highest there were about 2000 signals conveyed simultaneously through this link.

The simulation model was used for testing the automation, mainly concentrating on the control loops. The testing procedure was rather similar to the control loop commissioning procedure at site. The loops tested included steam temperature, steam pressure, drum level, flue gas oxygen, electric power, feedwater tank pressure

and mill temperature control loops. The testing uncovered a number of flaws in the control application, which resulted in a shorter commissioning time.



Figure 3: Narva Power Plant simulator

Additionally, different variations of the master control were tested by simulation and compared. The master control of the power plant includes the control of the electric power, turbine initial pressure and pressure in the two boilers. After the operation in steady state and in load changes was checked, it was verified that the control loops worked properly in certain disturbance situations, including a forced draught fan trip, a fuel mill trip and quick bypass of the preheaters.

### Mälär Energi Ab

Process Vision Ltd supplied a training simulator to Mälär Energi Ab, Västerås, Sweden for the new CFB boiler by Foster Wheeler Energia Oy. In addition to operator training, the simulation system was used by Foster Wheeler Energy for control system application planning before implementing the actual DCS system. The process concept was unique, and simulation gave the project parties confidence in the process design.

The simulation system consists of Trainer Grades DCS emulator and Control Grades simulation interface by Process Vision, CFB model by Foster Wheeler R&D centre and Apros simulation engine.

In control application planning, Control Grades toolkit was used for building a simulation model of both the control system and the process in the simulation engine and for testing the control application plan using the models. The toolkit features page templates for documentation, an easy-to-use signal cross-reference mechanism with automatic reference texts and versatile project management tools.

The operator displays for the simulator were built up with the Trainer Grades DCS emulator package utilizing information entered in the previous phase. After that, the dynamic behaviour of the boiler was studied, e.g. the challenging combination of the parallel steam lines of the new natural circulation boiler and the

older once-through boiler to the sliding pressure turbine generator. Also the preliminary controller tuning and adjusting of various set point curves and other parameters were accomplished with the simulator.

The system gave a unique opportunity to train the operators well before the commissioning period of the new boiler. It was possible to use a complete copy of the DCS system in the operator training with the look and feel of the plant, although the actual DCS system was not available.

From the end-user's perspective a significant benefit from simulation was in this case that despite the unique process concept the operators were able to confidently take over the plant at commissioning.

### Suomenoja Power Plant

The Suomenoja power plant, owned by Espoon Sähkö Inc., generates electricity and district heat for the municipalities of Espoo, Kirkkonummi and Kauniainen in Southern Finland. The Suomenoja plant is powered mainly by coal (60%) and natural gas (40%), and it has a high production efficiency due to the combined production of heat (350 MW) and electricity (120 MW).

In addition to several automation deliveries to Suomenoja power plant since the year 1986, Metso Automation supplied an advanced training simulator for a pulverized coal-fired power plant unit So1 in December 2000.

The supplied simulator system consists of the virtual metsoDNA environment and the Apros dynamical process simulation software, both of which have OPC data access interfaces, and OPC-based communication software. The system runs on two standard PC's, and it is operated through real operator stations.

In the virtual metsoDNA environment, the process control applications are executed exactly as in the real system on the plant, using the original application configuration without modifications. The virtual metsoDNA has simulator features, e.g. saving and loading the state, and freezing and resuming execution.

The scope of the simulation model was specified together with the customer. There are about 1300 simulated I/O connections and the simulated process areas include:

- feed water and steam
- air and flue gas
- coal pulverizers and coal feeding
- oil feeding and burners
- steam turbine and district heat

In addition to operator training, the simulation system was later used for planning of operator actions.

## Stora Enso

One of the leading paper and board producer Stora Enso Oyj wanted to shorten grade change times to improve productivity. An automatic grade change program was tuned using dynamic board machine simulation model.

The board machine in question produces 3-ply packaging and graphical boards having basis weight area of 170-350 g/m<sup>2</sup>. Typically there is at least one GC per day on the machine. The changes can be big, for example basis weight changes normally 20 g/m<sup>2</sup> or more.

It is very challenging to try to figure out the right actions needed to improve GC performance using just a mind model. In a multiply machine like this the number of tuning parameters of the AGC is as remarkable as 84 and describes the complexity of the tuning. Different ideas compete and conflict. The simulator was seen as a possibility to test new ideas before anything is done on the machine.

A decision was made to use simulator to get better understanding of the factors affecting on the grade change (GC) time, and to optimise the tuning of the automatic grade change program (AGC).

The accomplished process model covers the board making process from pulp chests to the end of the base board drying. The control system model includes 74 control loops. The model was built using the APROS Paper platform.

The focus was on optimising AGC parameters that define the mutual coordination and rates of the ramps of the controlled variables. To be able to do that, the simulator must confidently predict the effects of the simultaneous changes in operating variables.

Simulator was validated against measurement data from real GCs on the machine. Number of reference GCs was over 50. During the validation work, undesirable operator actions in using the AGC, were noticed. After these faults were identified, the operators were advised to the optimal and consistent use of the AGC.

After confidence was gained that the simulator can consistently repeat the GCs that have been made in the real machine, the what-if experiments were started. Fig. 4 shows some results of such what-if experiments. In the simulation runs different AGC parameters were experimented with.

Simulations concentrated on speeding up the GC's simply by increasing the ramping rates of those variables, which most often limited the total ramping time. One by one higher rates were tried out and the effects on the performance analyzed.

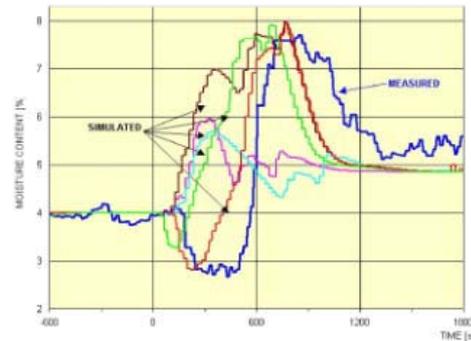


Figure 4: Grade Change scenarios

After extensive testing with the simulator, the best set of tuning parameters were taken into production use on the real machine. The machine's GC performance was monitored before and after the changes and positive development was observed.

For a multi-ply process, it is a demanding task to figure out how the automatic grade change program (AGC) parameters should be changed to speed up grade changes and simultaneously maintain high paper quality. The simulator helped to cut the problem into pieces, and offered a way to visualize the problem and compare the solution candidates. The parameter changes have been carried out to the real AGC. As a result the grade change time has been shortened considerably. Because of that the pay-back time of the simulation and optimisation project was 8 to 12 months.

Modeling and validation phase helped process engineers to understand the interactions during grade changes. Additionally, they were able to spot and remove some weaknesses in operators' practices concerning the use of the AGC program.

Besides the grade change development, also new applications for the model have been found. The model has already been used in studies concerning drying capacity increase and controllability with a new type of dryer.

The developed model offers a platform for troubleshooting, related to GCs or other issues on the simulator's scope. Concerning the development of GCs, with the existing simulation model it can be quickly checked if a new idea is worth of further studies. Additionally, the model offers an excellent basis for development of training or operator support simulator.

## Loviisa

The safety analyses were needed to prove that the uprated 1500 MWth power level of Loviisa Nuclear Power Plant does not cause any safety problems. The safety analysis included dynamic simulation studies of a

series of accident scenarios. Main tool in the analysis work was APROS Simulation Software.

A feasibility study for modernizing the two 465 MWe VVER units of Loviisa Nuclear Power Plant was carried out starting in spring 1994. During the study no technical, safety or licensing issues were identified which would have prevented raising the reactor thermal output up to 1500 MWth from the level of 1375 MWth. Thus the modernization project including a 9.1 % reactor power uprating was launched in summer 1995.

It was obvious from the very beginning that reactor power uprating would bring about a need for extensive revision of the Loviisa Final Safety Analysis Report (FSAR) including the safety analyses. It was also clear that tasks related to the safety assessment by appropriate computer codes and models would be on the critical path on the project schedule. That is why this work was started at the same time as the project organization was put together and the master plan prepared. The intention was to increase in the plant capacity, by about 50 MWe per unit. This was planned to be achieved by a combination of reactor thermal power uprating and by improving turbine efficiency.

The uprated reactor power was expected to raise the temperature difference over the reactor by three degrees and the temperature of the sea water trough condenser by one degree.

The analyses were needed to update the Loviisa FSAR to correspond to the uprated 1500 MWth power level and thus to prove that power uprating does not cause any safety issue.

In 1995 Fortum Oy (former Imatran Voima Oy) decided to implement Loviisa modernization and power uprating project major. The major part of the revised Loviisa FSAR thermal hydraulic analyses were calculated using the APROS simulation software.

A completely new simulation model of Loviisa NPP was build for the project. This was to ensure that all the input data was correct and the sources properly documented. The model included whole primary circuit including safety systems, steam generators, steam lines and safety critical automations systems. The model was extensively validated against measurement data from the plant commissioning tests etc.

Around 30 different initiating events and scenarios were calculated and their sensitivity to various parameters investigated. Simulations were done using the uprated power level:

- Large break and small break loss of coolant accidents
- Anticipated transients without scram (ATWS)
- Primary to secondary leakages

- Several different pump trips, line breaks, blackouts and valve malfunctions etc.

Based on the analysis results 1500 MWth is a safe power level to operate Loviisa reactors from the safety analyses point of view. This result was also expected because earlier licensing analyses, concerning 1375 MWth nominal power level, had shown that the margins to the acceptance criteria were in most cases substantial.

The large break loss of coolant accident (LBLOCA) is generally regarded as one of the critical accident scenarios. That is because virtually the whole reactor core will be dry shortly after the break. The results showed that even in this case the hot rod cladding temperature can be maintained in safe region and the whole core is rewetted in five minutes after the break.

APROS proved to be an excellent tool on safety analysis field and Fortum Nuclear Services is currently doing practically all the safety analyses using it. The earlier major tool RELAP5 code has a role in assessing the APROS analysis results.

The power uprating was successfully concluded in 1999. Presently Loviisa NPP is operating at the uprated power level meaning on average 50 MWe higher electrical power per unit. Depending on the price of electricity this translates into additional revenue of 10-15 M€/year and unit.

## CONCLUSIONS

On the basis of the example cases presented, it is easy to see that simulation brings about great benefits throughout the life span of a process plant.

1. The process and automation concepts can be studied and validated in the planning phase, before selecting the suppliers (case Västerås). This reduces risks in the commissioning projects and improves the mutual understanding of project deliverables within the consortium.
2. The automation application can be verified before the commissioning by using virtual automation (case Narva). This shortens the commissioning time as there are fewer flaws in the system and the personnel has done the site acceptance test operations on the simulation system before coming to the site.
3. The benefits of training simulators (cases Västerås, Narva, Suomenoja) are widely acknowledged. A simulation environment is an invaluable tool in transferring knowledge from the commissioning staff to the operators in the end-user organisation and from experienced operators to novices.
4. The control system and operative procedures can be further developed and optimised after commissioning (case Stora-Enso). The simulation

system can also be used by the end user for the planning of renewals.

5. In safety-critical industries, simulation is practically taken required for various kinds of analyses (case Loviisa). It can be estimated that in time requirements like these will spread to other industries.

As a rule, when planning the use of simulation to support a commissioning project, one should try to find more than one use for simulation in and after the project.

In the future, the standardisation of plant data models will greatly reduce the manpower, time and costs of building up a simulation model. That development is likely to make simulation a rule, not an exception.

Referred modelling and simulation services have been completed by one or several of engaged parties: Process plant supplier, control system supplier, consultancy supplier, or software supplier.

The services have been supplied during different phases of the life cycle of the plants: Validation of plant performance in postulated operational and disturbance situations, evaluation of joint operation of process and automation design, optimisation of operational procedures, testing of functionality of real control system implementations, and use of simulator for operator training purposes, as well.

Combining the scopes of the presented applications gives a view of what a whole life cycle lasting integrated service concept could provide for: Optimisation of design, mitigation of economical risks, and a long lasting competitiveness.

According to our experiences, it would be very advantageous if the plant designer, resource optimiser, personnel instructor, trouble shooter, plant operator, who ever involved, could get access to the model required for his study, ubiquitously from his PC, regardless of where he is, where the repository resides or where the model runs.

The advent of persistent process design data repositories is the key issue. Existing basic standards for semantic and ontological specifications enable the required developments. Success stories like the above, help to ensure the different stake holders of the benefits, and accordingly promote the introduction of new required practises of model supported design and operation.

## REFERENCES

- Ansaharju, J. (2002). *Increasing drying capacity on a board machine*. Master's thesis. Helsinki University of Technology, Department of Forest Industry, Espoo.

Hänninen, M.; K. Juslin and K. Porkholm (1987) A flexible simulation system for the design and analysis of nuclear processes. American Nuclear Society Winter Meet. 1987. Los Angeles, 15 - 19 Nov. 1987. American Nuclear Society. Los Angeles, 528 - 530

Lappalainen, J., T. Myller, O. Vehviläinen, S. Tuuri and K. Juslin (2001). Enhancing grade changes with dynamic simulation. Proceeding of Tappi Engineering Conference 2001, San Antonio, 11 p.

## AUTHOR BIOGRAPHIES



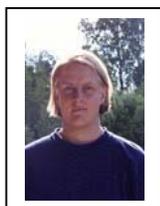
**JEAN-PETER YLÉN** was born in Helsinki, Finland and went to Helsinki University of Technology, where he received a master's degree in chemical engineering and a doctor's degree in automation and system science. He is currently working for VTT in the field of modeling and simulation.



**MATTI PALJAKKA** was born in Lappeenranta, Finland and went to Helsinki University of Technology, where he received a master's degree in nuclear engineering. He is currently working for VTT in the field of modeling and simulation.



**TOMMI KARHELA** was born in Tampere, Finland. He received his master's degree in computational physics at Tampere University of Technology and doctor's degree in information technology in automation at Helsinki University of Technology. He is currently working for VTT in the field of modeling and simulation of industrial processes.



**JOUNI SAVOLAINEN** was born in Helsinki, Finland and went to Helsinki University of Technology, where he received a master's degree in chemical engineering. He is currently working for VTT in the field of modeling and simulation.



**KAJ JUSLIN** was born in Turku, Finland and studied at Helsinki University of Technology. He has actively contributed in the work of such simulation organisations as SIMS, EUROSIM and Sim-Serv. He is currently manager of the System Dynamics research group at VTT.

# Simulation of Spacecraft Attitude and Orbit Dynamics

Pasi Riihimäki, Jean-Peter Ylén  
Control Engineering Laboratory  
Helsinki University of Technology  
PL-5500, 02015 TKK  
E-mail: pasi.riihimaki@tkk.fi, peter.ylen@tkk.fi

## KEYWORDS

Simulation Model, Satellite, FDIR, Quaternion

## ABSTRACT

In this paper, the simulation model of satellite attitude and orbit dynamics is discussed. The satellite attitude model has been represented in term of a quaternion and a ordinary differential equation is used to describe the satellite orbital motion. The different actuators and sensors have been modeled with suitable faults and failures. The simulation model enables us to consider the satellite motion under different environmental perturbations (for example aerodynamic drag, external celestial body etc.) and failure in actuators and sensors. The simulation model is utilized in the development of attitude and orbit control algorithms or fault detection, isolation and recovery (FDIR) technologies. Simulation results are also given.

## INTRODUCTION

During the last decades modeling, simulation, and wider computational science and engineering have become more and more important tools in the research and development projects. The design phase has to be reduced in time and cost when the use of new ideas and tools becomes possible. This is also the trend in space application in which the real tests are not possible or at least they are expensive. New demands on the aerospace and control engineering have become up and they have to be able to answer to requirements.

Spacecraft simulators or simulators in general, are software tools that can be used by researchers, engineers, students or everybody to analyze and assess system operations, behaviors, and to answer to the questions regarding phenomenon or product. The simulations are essential tools in the mission and spacecraft control design. For example, the scientific missions are unique and the instrumentation of a spacecraft is designed only for this specific mission. There are not any ready-to-use platforms that can be used. Hence, it is not possible to verify the operation of control algorithms and strategies in real process but the simulation environments can be used.

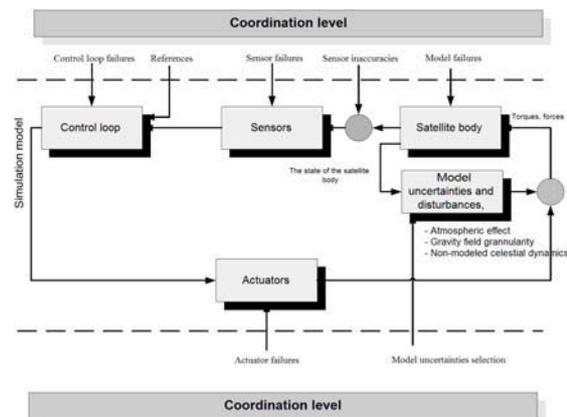
There are plenty of companies that offer their simulation services to the research institutes and space companies.

## MODEL STRUCTURE AND MATHEMATICS

### Model Structure

The spacecraft simulation model is organized like any actual control loops (See Figures 1.). The interfaces of the components are defined and modeled in such a way that the simulation model would be as modular as possible. Modifications to the simulation model are easy to do and one part of the model can be easily replaced with another.

The model is initialized and controlled from the coordination level. This means that the model parameters and possible faults and failures in the FDIR simulation case are defined.



Figures 1. Spacecraft simulation model structure.

### Coordinate Systems

Three different coordinate systems are defined in the simulator:

1. Inertial Coordinate System (ICS),
2. Orbit Coordinate System (OCS), and
3. Body Coordinate System (BCS).

The inertial coordinate system is usually defined such that the center of mass of the Earth (cm) acts as origin and the direction of the axes are fixed to the solar system. This kind of coordinate system is not exactly inertial but it is enough for all engineering purposes (Sidi 1997). The **Z**-axis of the ICS is the rotation axis of the Earth in a positive direction and the **X-Y** plane is the equatorial plane of the Earth, which is perpendicular to the Earth's rotation axis. The vernal equinox vector  $\Upsilon$  is selected to be the **X**-axis of the ICS. Finally, the **Y**-axis

has been chosen in such a way that the ICS is right-handed orthogonal coordinate system.

Orbit coordinate system is also a right-handed orthogonal coordinate system with origin in the center of the satellite mass. The **Z**-axis is pointing towards the center of the Earth; **X**-axis to the direction of satellite perpendicular to the **Z**-axis, and **Y**-axis completes the coordinate system such that it is right-handed and orthogonal. The third coordinate system, which has been fixed to the moving and rotating spacecraft, defines satellite orientation.

### Rotation

The attitude transformation in space can be executed by using various different aspects. In the simulation model, the quaternion technique is used. The main feature of quaternions is that they provide a convenient product rule for successive rotations and they have simple form of kinematics (Wertz 1978, Wis'niewski 1996).

The basic definition of the quaternion is a consequence of the property of the direction cosine matrix **A** that it has at least one eigenvalue of unity. This means that there is an eigenvector **e** (Euler axis) that is unchanged in every rotation. The quaternion is defined as a vector (1) where  $q_i \in R$ , **i**, **j** and **k** satisfy the Hamilton's rule (2) and where the length of the quaternion is unity. (Sidi 1997)

$$\mathbf{q} = q_4 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k} \quad (1)$$

$$\begin{aligned} \mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1 \\ \mathbf{ij} = -\mathbf{ji} = \mathbf{k} \\ \mathbf{jk} = -\mathbf{kj} = \mathbf{i} \\ \mathbf{ki} = -\mathbf{ik} = \mathbf{j} \end{aligned} \quad (2)$$

When the Euler axis **e** of the rotation is known the connection between quaternion and the rotation Euler axis is

$$\begin{cases} q_1 = e_1 \sin(\alpha/2) \\ q_2 = e_2 \sin(\alpha/2) \\ q_3 = e_3 \sin(\alpha/2) \\ q_4 = \cos(\alpha/2) \end{cases}$$

where  $e_i$  is a component of Euler axis and  $\alpha$  is the magnitude of the rotation.

The final combined rotation of two successive rotations can be performed as a matrix-vector multiplication (3) where **q** and **q'** are the individual rotations.

$$\mathbf{q}'' = \mathbf{q}\mathbf{q}' = \begin{pmatrix} q_4' & q_3' & -q_2' & q_1' \\ -q_3' & q_4' & q_1' & q_2' \\ q_2' & -q_1' & q_4' & q_3' \\ -q_1' & -q_2' & -q_3' & q_4' \end{pmatrix} \begin{pmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{pmatrix} \quad (3)$$

### SIMULATION MODEL

The simulation model is realized in the MATLAB/SIMULINK-environment.

#### Orbit Model

The motion of a celestial body is based on the quite elementary principles of celestial mechanics. In the 17<sup>th</sup> century, J. Kepler provided three basic empirical laws that describe the motion of planet in unperturbed planetary orbit. The orbital dynamics of a satellite is extensively explained in many books, for example (Sidi 1997) and (Wertz 1978).

If we consider a system of two particles  $P_1$  and  $P_2$  of masses  $m_1$  and  $m_2$  and apply Newton's second law and the law of gravity to the two-body system, we can get the fundamental equation (4) of the motion of the two-body system where the symbol  $\mu = G(m_1+m_2)$  and  $G$  is the universal constant of gravitation. This equation describes the motion of the particle  $P_1$  relative to the second mass  $P_2$ .

$$\ddot{\mathbf{r}} + \frac{\mu}{r^3}\mathbf{r} = 0 \quad (4)$$

In general, if a particle  $P$  moves in a force field **F**, the momentum of the force **F** about origin  $O$  is

$$\mathbf{M} = \mathbf{r} \times \mathbf{F}$$

where **r** is the position vector of the particle  $P$ . The angular momentum about origin is

$$\mathbf{h} = m(\mathbf{r} \times \mathbf{v}) = \mathbf{r} \times \mathbf{p}$$

where **p** is the linear momentum of the particle. Thus, the time rate of the angular momentum **h** is equal to the moment of the force **F**.

$$\frac{d\mathbf{h}}{dt} = \frac{d}{dt}(\mathbf{r} \times m\mathbf{v}) = 0 + \mathbf{r} \times \mathbf{F} = \mathbf{M} \quad (5)$$

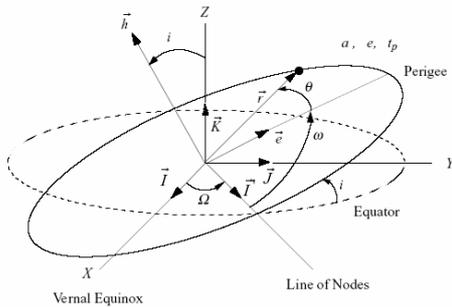
The equation (5) states the fundamental fact that the momentum acting on a particle is equal to the time rate of the change of its angular momentum.

In space science it is common to describe the satellite orbit by five numbers, known as orbital elements or classical orbital elements (COE). A sixth element is added to determine the location of the satellite in its orbit (Wertz 1978 and Sidi 1997). The classical orbital

elements have been described in the Table 1. Because these elements are poorly defined if  $e$  and/or  $i$  is equal to zero, so-called equinoctial orbital elements (EOE) have been defined in terms of the classical orbital elements. The equinoctial orbital elements have been defined in Table 2.

Table 1. The classical orbital elements. (Sidi 1997)

Symbol	
$a$	the semi major axis
$e$	the eccentricity
$i$	the inclination
$\Omega$	the right ascension of the ascending node
$\omega$	the argument of perigee
$M$	the mean anomaly



Figures 2. The definitions of the elements.

Table 2. The definitions of the equinoctial orbital elements (EOE). (We and Roithmayr, C.M. 2001)

EOE	
$a$	$a$
$P_1$	$e \sin(\Omega + \omega)$
$P_2$	$e \cos(\Omega + \omega)$
$Q_1$	$\tan\left(\frac{\theta}{\Omega}\right) \sin(\Omega)$
$Q_2$	$\tan\left(\frac{\theta}{\Omega}\right) \cos(\Omega)$
$l$	$\Omega + \omega + M$

In Keplerian orbit the derivative of the first five orbital elements are equal to zero. If the satellite orbital elements are known the satellite location  $\mathbf{r}$  and the velocity vector  $\mathbf{v}$  can be calculated, and vice versa. Algorithms to do this can be found in any textbook concerning orbital dynamics, for example (Sidi 1997), (Wertz 1978).

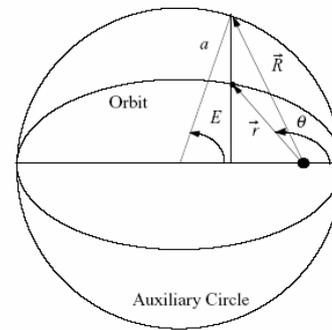
In the general case, in which any kind of perturbing force can exist, the equation of orbital motion is

$$\ddot{\mathbf{r}} + \frac{\mu}{r^3} \mathbf{r} = \mathbf{f}_p$$

with initial condition and where  $\mathbf{f}_p$  is the perturbing force per unit mass. Due to the perturbing acceleration the orbital elements are not constants. Hence, so-called Gauss form of Lagrange's planetary equations describes

the derivative of classical orbital elements when the perturbing force is conservative or non-conservative. Knowing the initial condition of COE the Gauss equations can be integrated to calculate the evolution of the elements. Gauss equation is represented in equation (7), where  $\theta$  is the angle between satellite location vector and the vector pointing towards perigee (See Figures 3.),  $p = a(1-e^2)$ ,  $n = \sqrt{\mu/a^3}$ ,  $r = p/(1+e \cos(\theta))$ , and  $f_r$ ,  $f_\theta$  and  $f_z$  are the components of the perturbing force along the radius vector direction  $\mathbf{r}$ , the transverse orbit direction and the direction of the normal to the orbit plane, respectively.

To avoid the singularity due to the poorly defined parameters, the Gauss equations can be rewritten in the terms of the equinoctial elements as in (8), where  $b = a \sqrt{1-P_1^2-P_2^2}$ ,  $h = nab$ ,  $p/r = 1 + P_1 \sin(L) + P_2 \cos(L)$ ,  $L = \omega + \Omega + \theta$ , and  $K = \omega + \Omega + E$ . (We and Roithmayr, C.M. 2001)



Figures 3. The spacecraft orbit and auxiliary circle.

## Attitude Model

### Dynamics

From equation (5) we get that the torque acting on the satellite body is equal to the derivative of the angular momentum of the spacecraft in the inertial coordinate system. Hence, in the rotating body coordinate system

$$-\mathbf{T} = \dot{\mathbf{h}}_I = \dot{\mathbf{h}} + \boldsymbol{\omega} \times \mathbf{h}$$

If momentum exchange devices are used in the control, the angular momentum vector  $\mathbf{h} = \mathbf{h}_B + \mathbf{h}_w$  where  $\mathbf{h}_B$  is the angular momentum of satellite rigid body and  $\mathbf{h}_w$  is the angular momentum of the momentum exchange devices. Hence, the time rate of angular velocity of the satellite body is like in equation (6).

$$\dot{\boldsymbol{\omega}} = (\mathbf{I}_s)^{-1} \cdot \left( -\dot{\mathbf{I}}_s \boldsymbol{\omega} + \sum_i \mathbf{T}_i - \dot{\mathbf{h}}_w - \boldsymbol{\omega} \times (\mathbf{I}_s \boldsymbol{\omega}) - \boldsymbol{\omega} \times \mathbf{h}_w \right) \quad (6)$$

### Kinematics

The spacecraft attitude has been modeled as a quaternion representation  $\mathbf{q} = (q_1, q_2, q_3, q_4)$ . Hence, the equation (9), where  $\omega_i$  is the satellite angular velocity about satellite body axis  $i$ , gives the derivative of quaternion vector.

$$\begin{aligned}
\dot{a} &= \frac{2a^2}{\sqrt{\mu p}} \left( f_r e \sin(\theta) + f_\theta (1 + e \cos(\theta)) \right) \downarrow \\
\dot{e} &= \sqrt{\frac{p}{\mu}} \left[ f_r \sin(\theta) + f_\theta \frac{\textcircled{R}}{\textcircled{TM}} \cos(\theta) + \frac{e + \cos(\theta)}{1 + e \cos(\theta)} \right] \downarrow \\
\dot{i} &= \frac{r f_z \cos(\omega + \theta)}{\sqrt{\mu p}} \\
\dot{\Omega} &= \frac{r f_z \sin(\omega + \theta)}{\sqrt{\mu p} \sin(i)} \quad (7) \\
\dot{\omega} &= -\frac{f_z r \sin(\omega + \theta) \cos(i)}{\sqrt{\mu p} \sin(i)} \\
&\quad - \frac{1}{e} \sqrt{\frac{p}{\mu}} \left[ f_r \cos(\theta) - f_\theta \frac{\textcircled{R}}{\textcircled{TM}} + \frac{r}{p} \right] \sin(\theta) \downarrow \\
\dot{M} &= n - \frac{2r f_r}{na^2} + \frac{1-e^2}{nae} \left[ f_r \cos(\theta) - f_\theta \frac{\textcircled{R}}{\textcircled{TM}} + \frac{r}{p} \right] \sin(\theta) \downarrow
\end{aligned}$$

$$\begin{aligned}
\dot{a} &= \frac{2a^2}{h} \left( (P_2 \sin(L) - P_1 \cos(L)) f_r + \frac{P}{r} f_\theta \right) \downarrow \\
\dot{P}_1 &= \frac{r}{h} \left[ \frac{P}{r} \cos(L) f_r + \left( P_1 + \frac{\textcircled{R}}{\textcircled{TM}} + \frac{P}{r} \right) \sin(L) \right] f_\theta \\
&\quad - P_2 (Q_1 \cos(L) - Q_2 \sin(L)) f_z \downarrow \\
\dot{P}_2 &= \frac{r}{h} \left[ \frac{P}{r} \sin(L) f_r + \left( P_2 + \frac{\textcircled{R}}{\textcircled{TM}} + \frac{P}{r} \right) \cos(L) \right] f_\theta \\
&\quad + P_1 (Q_1 \cos(L) - Q_2 \sin(L)) f_z \downarrow \\
\dot{Q}_1 &= \frac{r}{2h} (1 + Q_1^2 + Q_2^2) \sin(L) f_z \\
\dot{Q}_2 &= \frac{r}{2h} (1 + Q_1^2 + Q_2^2) \cos(L) f_z \\
\dot{i} &= n - \frac{r}{h} \left( \frac{a}{a+b} \frac{P}{r} \frac{\textcircled{R}}{\textcircled{TM}} \sin(L) + P_2 \cos(L) + \frac{2b}{a} \right) f_r \\
&\quad + \frac{a}{a+b} \frac{\textcircled{R}}{\textcircled{TM}} + \frac{P}{r} \left[ (P_1 \cos(L) - P_2 \sin(L)) f_\theta \right. \\
&\quad \left. + (Q_1 \cos(L) - Q_2 \sin(L)) f_z \right] \downarrow
\end{aligned} \quad (8)$$

$$\frac{d}{dt} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 0 & \omega_3 & -\omega_2 & \omega_1 \\ -\omega_3 & 0 & \omega_1 & \omega_2 \\ \omega_2 & -\omega_1 & 0 & \omega_3 \\ -\omega_1 & -\omega_2 & -\omega_3 & 0 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} \quad (9)$$

## Actuators

The actuators are used to produce the control torques and forces for the satellite attitude control. The modeled actuators are:

- thruster,
- reaction-wheels, and
- magnetotorquer.

The thruster has been modeled as a thrust force vector  $\mathbf{F}_t$  affecting the satellite in position  $\mathbf{r}_t$ . Hence, the torque about the center of the mass of the spacecraft is the cross product between the position and the force vectors (Equation (10)).

$$\mathbf{T}_t = \mathbf{r}_t \times \mathbf{F}_t \quad (10)$$

The idea of reaction wheels (or momentum exchange devices) is to transfer the angular momentum of the whole system between different parts of the spacecraft without changing its overall internal angular momentum. The achieved torque level is of the order of 0.05 – 2 Nm. (Sidi 1997)

The reaction-wheel is modeled as equation (11).

$$\begin{cases} \dot{\boldsymbol{\omega}}_w = \mathbf{f}(\mathbf{T}_{dem}) - \mathbf{f}_\mu \\ \mathbf{h}_w = I_w \boldsymbol{\omega}_w \end{cases} \quad (11)$$

In magnetotorquer, the control torque  $\mathbf{T}_{mag}$  is generated by an interaction of the Earth's geomagnetic field  $\mathbf{B}(t)$  with the magnetic dipole momentum  $\mathbf{m}(t)$  (See equations (12) and (13)) where  $n_{coil}$  is the number of coil,  $i_{coil}(t)$  is the magnetotorquer current,  $A_{coil}$  loop area, and  $\hat{\mathbf{n}}$  is the unit normal vector to the plane of the loop.

$$\mathbf{T}_{mag}(t) = \mathbf{m}(t) \times \mathbf{B}(t) \quad (12)$$

$$\mathbf{m}(t) = n_{coil} i_{coil}(t) A_{coil} \cdot \hat{\mathbf{n}} \quad (13)$$

## Sensors

In the simulation model the modeled sensors are:

- coarse Earth and Sun Sensor (CESS),
- star tracker,
- magnetometer,
- gyro, and
- GPS.

The CESS is modeled as a component that gives the direction of Sun and Earth in the body coordinate system.

The star tracker is modeled as a component that gives the satellite attitude contaminated with an uncertainty that depends on the satellite angular speed. Magnetometer is modeled as a component that gives the magnitude and direction of the Earth's magnetic field. WMM magnetic model is used in the simulator.

A gyroscope is modeled as an instrument that measures the angular speed of the spacecraft. The actual angular speed is contaminated with relatively small Gaussian random uncertainty. A GPS is modeled as an instrument that gives the satellite location in the inertial Earth centered coordinate system.

## Faults

One of the main aims of the spacecraft simulation model is that it can be used in the FDIR-simulation. Hence, the faults and failures have to be taken into account already in the design and modeling phase. The faults can occur in any part of the model and any kind of

faults are possible. Usually, the faults are either additive or parametric but also a total blackout of a component is possible.

Perhaps, the most prevalent fault is ice building on the surface of any optical instrument increasing the inaccuracy of this element.

### Environmental Torques

The main sources of the environmental torques are represented in the Table 3.

Table 3. The main environmental torques (Wertz 1978).

Source	Dependence	Dominant
Aerodynamic	$e^{-\alpha r}$	below ~ 500 km
Magnetic	$1/r^3$	~ 500 - 35 000 km
Gravity Gradient	$1/r^3$	~ 500 - 35 000 km
Solar Radiation	Independent	Interplanetary space above synchronous altitude
Micrometeorites	independent	Normally negligible

The aerodynamic drag is one of the main environmental torques for the spacecraft in low orbit. The aerodynamic drag model has been explained extensively, for example, in the book (Wertz 1978).

The force  $d\mathbf{f}_a$  on the surface elements  $dA$  is given by equation (14) where  $\hat{\mathbf{N}}$  is a outward normal of the surface element  $dA$ ,  $\hat{\mathbf{V}}$  a unit vector of the translational velocity,  $\rho$  is the air density and  $C_D$  is the drag coefficient of the surface. In real terms, the drag coefficient  $C_D$  is a function of the surface structure and the local angle of attachment and its value is usually between 1 and 2. For all practical applications, the value  $C_D=2$  can be used. (Wertz 1978)

$$d\mathbf{f}_{aero} = -\frac{1}{2} C_D \rho V_0^2 (\hat{\mathbf{N}} \cdot \hat{\mathbf{V}}) \hat{\mathbf{V}} dA \quad (14)$$

In the simulation model, the satellite structure has been approximated by a collection of simple geometrical figures. Hence, the total aerodynamic torque is the sum over the torques acting on individual parts of the spacecraft.

$$\begin{aligned} \mathbf{T}_{aero} &= \sum_i \mathbf{r}_i \times \mathbf{f}_i \\ &= \frac{1}{2} C_D \rho V_0^2 \sum_i A_i (\hat{\mathbf{N}}_i \cdot \hat{\mathbf{V}}) \hat{\mathbf{V}} \times \mathbf{r}_i \end{aligned}$$

Any nonsymmetrical body in orbit is subject to a gravitational torque because of the variation in the Earth's gravitational force over the object. Usually in the literature, the gravity-gradient is only derived for the unrealistic spherical Earth model. Due to the non-sphericity and the non-homogenous mass distribution of the Earth the real gravitational field is granular.

For spherical Earth, the gravitational force  $d\mathbf{f}_i$  acting on a s/c mass element  $dm_i$  located at a position  $\mathbf{R}_i$  is

$$d\mathbf{f}_i = \frac{-\mu \mathbf{R}_i dm_i}{R_i^3}$$

Hence, the torque about the satellite geometric center due to a force  $d\mathbf{f}_i$ , at position  $\mathbf{r}_i$  is

$$d\mathbf{T}_i = \mathbf{r}_i \times d\mathbf{f}_i = (\boldsymbol{\rho} + \mathbf{r}'_i) \times d\mathbf{f}_i$$

where  $\boldsymbol{\rho}$  is the vector from the geometric center to the cm and  $\mathbf{r}'_i$  from cm to the mass element  $dm_i$ . Assuming that the cm and the geometric center of the s/c lie in the same point the gravity-gradient is

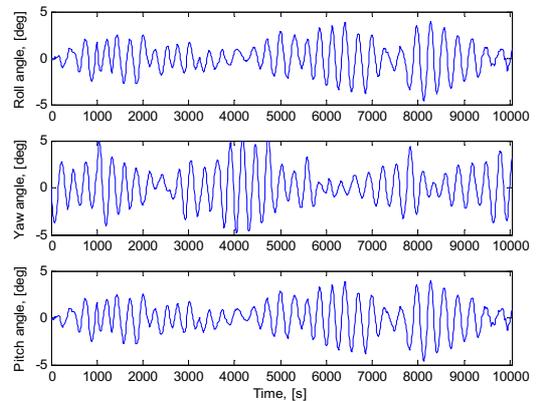
$$\mathbf{T}_{gg} = \frac{3\mu}{R_s^3} (\hat{\mathbf{R}}_s \times \mathbf{I}_s \hat{\mathbf{R}}_s)$$

where  $\hat{\mathbf{R}}_s$  is a unit vector along  $\mathbf{R}_s$  and  $\mathbf{I}_s$  is the spacecraft inertial matrix. (Wertz 1978)

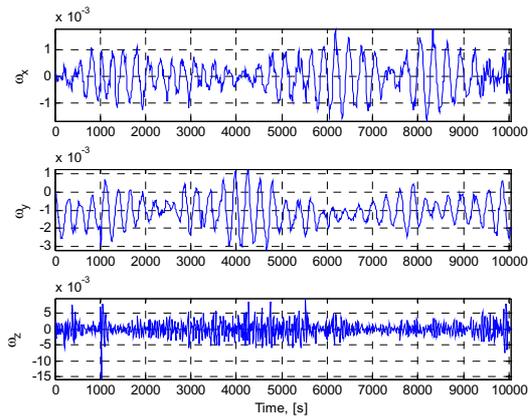
### A SIMULATION CASE

In this section, some simulation results obtained by the above-described simulation model are presented. The simulation case is simple and fancied.

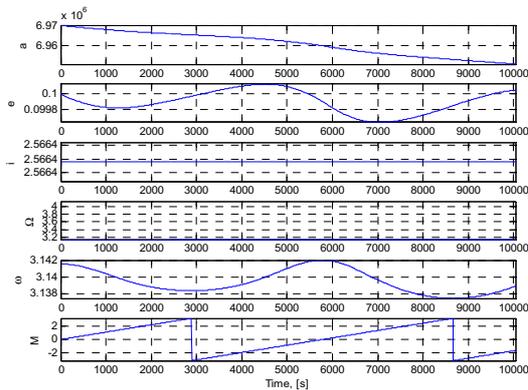
The orbit of the simulation case is circular with an altitude of 450 km and inclination  $87^\circ$ . The moments of inertia of the satellite are  $I_{xx}=36$ ,  $I_{yy}=17$ ,  $I_{zz}=26$ , and  $I_{xy}=I_{yz}=I_{xz}=0 \text{ kgm}^2$ . The aim is that the attitude control system shall ensure a three-axis stabilization of the satellite. The satellite attitude is measured by GPS sensor and only three reaction wheels are used in the control. The reaction wheels are mounted orthogonally such that the rotation axes are along  $\mathbf{X}$ ,  $\mathbf{Y}$  and  $\mathbf{Z}$ -axis of the satellite body. Three PID-controllers are used to calculate the control torques. The simulation results have been represented in Figures 4-8.



Figures 4. Attitude angles.



Figures 5. The satellite angular rates.

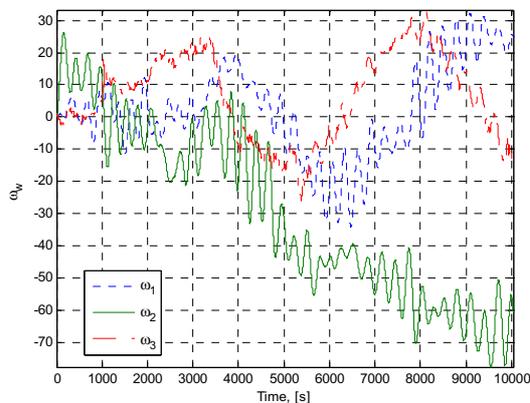


Figures 6. The classical orbital elements in simulation.

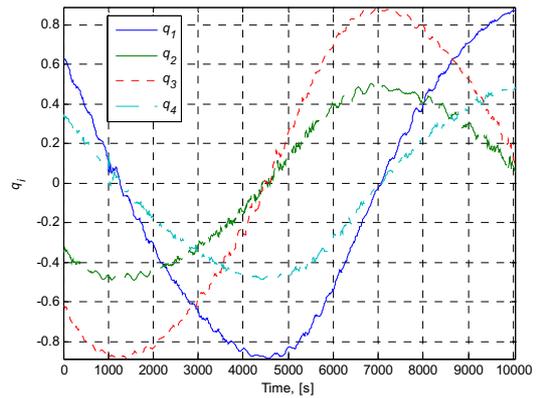
## CONCLUSION

The satellite attitude and orbit simulation model with the most common actuators and sensors have been introduced in this paper. The simulation model can be utilized both in the control algorithm designs and in the development of FDIR methods.

The simulation models have been implemented in the MATLAB/SIMULINK environment.



Figures 7. The angular rates of reaction wheels.



Figures 8. The attitude quaternions.

## REFERENCES

- Naasz, B.J. 2002. "Classical Element Feedback Control for Spacecraft Orbit Maneuvers" Thesis, Master of Science. Virginian Polytechnic Institute and State University, Aerospace Engineering. 90 p.
- Sidi, M.J. 1997. "Spacecraft Dynamics & Control – A Practical Engineering Approach" Cambridge University Press. 409 p. ISBN 0-521-78780-7
- We, B. Roithmayr, C.M. 2001. "Integrated Orbit, Attitude, and Structural Control Systems Design for Space Solar Power Satellites" NASA/TM-2001-210854. [10.3.2005] <http://techreports.larc.nasa.gov/ltrs/PDF/2001/tm/NASA-2001-tm210854.pdf>
- Wertz, J.R. 1978. "Spacecraft Attitude Determination and Control" D. Reidel Publishing Company, Dordrecht, Holland. 858 p. ISBN 90-277-0959-9
- Wis'niewski, R. 1996. "Satellite Attitude Control Using Only Electromagnetic Actuation" Ph. D. Thesis. Denmark, Aalborg University, Department of Control Engineering. 175 p.

## AUTHOR BIBLIOGRAPHIES



**PASI RIIHIMÄKI** was born in Ähtäri, Finland and went to Helsinki University of Technology, where he received a master's degree in automation and system science. Nowadays, he is a postgraduate student in the Control Engineering Laboratory in Helsinki University of Technology.



**JEAN-PETER YLÉN** was born in Helsinki, Finland and went to Helsinki University of Technology, where he received a master's degree in chemical engineering and a doctor's degree in automation and system science.

# SIMULATION-BASED PREDICTIVE EMISSION MONITORING SYSTEM

Mincho Hadjiski, Kosta Boshnakov and Nikolinka Christova  
Department of Automation in Industry  
University of Chemical Technology and Metallurgy  
Blvd. Kliment Ohridski, No. 8, 1756 Sofia, Bulgaria  
E-mails: [hadjiski@uctm.edu](mailto:hadjiski@uctm.edu), [kb@uctm.edu](mailto:kb@uctm.edu), [nchrist@uctm.edu](mailto:nchrist@uctm.edu)

## KEYWORDS

Hybrid modeling, inferential measurement, mathematical model, predictive emission monitoring system, thermal power station

## ABSTRACT

A development and investigation of mathematical models for Predictive Emission Monitoring Systems (PEMS) in order to reconstruct the missing data on the base of low frequent direct measurements are discussed. A comparative analysis is provided among variety of individual and hybrid models using real experimental data. Practical recommendations are derived for design of PEMS oriented to thermal power plants.

## INTRODUCTION

The main functions of the emission monitoring systems in thermal power stations are the measurements and the registration of the harmful emissions and the preparation of the reports for the inspections for environmental protection.

The Regulating agencies in USA and in the countries of the EC have regulated two ways of reporting data (readings) formation in the emission monitoring systems (Eberhard 1995; Joseph and Macak 1996; Ekhus and Black; Gimenez et al.; EPA Handbook 1997; US Environmental Protection Agency 2000):

- By using direct measurements in the Continuous Emission Monitoring Systems (CEMS), which is the traditional approach;
- By using mathematical models based on the indirect measurements for realization of the ecological goals, known as Predictive Emission Monitoring Systems (PEMS).

In both of the reporting data sources the requirements of the national legislations (Bulgarian Ministry of Environmental Protection) and the statutory procedures and the standards in USA and EC should be satisfied (EPA Handbook 1997; Alberta Environmental Protection 1998; US Environmental Protection Agency 2000; Code of Federal Regulations, Part 60; Code of Federal Regulations, Part 75).

The Predictive Emission Monitoring has strengthened its position in USA and it's provided for with the respective

normative documents (Eberhard 1995; Ekhus and Black; Topical Report №17 2001; Ghien et al. 2003). PEMS have been introduced in power stations in USA and Europe with total capacity of 10000 MW. The emission monitoring on the base of indirect measurements is considered as an alternative of the traditional approach (CEMS). Besides the carrying out of the basic functions, the predictive monitoring systems enable realization of optimal control of the combustion processes under the ecological restrictions, realization of some diagnostic functions as well as usage for generation of recommendations for operational and control activities, which will improve the total effectiveness of the equipment works on lower investment expenses.

The main goals of the current study are construction and investigation of mathematical models for PEMS applied to thermal power plant and development of procedures for missing data reconstruction on the base of low frequent direct measurements when CEMS and PEMS work together.

The criteria for discrimination of the particular solutions are: accuracy, validity for all the technological regimes, operational reliability and maintenance expenses.

## RECONSTRUCTION OF THE MISSING DATA IN THE EMISSION MONITORING SYSTEMS

The aim of this study is development of Predictive Emission Monitoring Systems (PEMS) in thermal power station on the base of expert knowledge and by using simulation approach. The system for ecological monitoring of the considered project is realized as multiplex system for consecutive analysis of the pollutants for each gas outlet. The used scheme allows multifunctional carrying out of the Continuous Emission Monitoring Systems (CEMS) for realization of optimal control of the combustion processes except for performance of its main purpose. The time period of multiplexing of each gas outlet is  $T_0 = 150$  s (2.5 min). In this way the analysis data of oxygen ( $O_2$ ), nitrogen oxides ( $NO_x$ ) and sulphur dioxide ( $SO_2$ ) for each gas outlet enter every 5 min, 10 min, or 15 min depending on the number of the working steam generators (respectively 1, 2 or 3).

The determination of the substituted values of gas emissions of O<sub>2</sub>, NO<sub>x</sub> and SO<sub>2</sub> for each gas outlet (every 2.5 min) is accomplished according to the principals of the predictive emission monitoring. The high accuracy of the predictive emission monitoring makes it equivalent and in some cases better than the continuous emission monitoring or sometimes it's the only possible method.

The necessary information for reconstruction of the missing data of the components of the flue gases is delivered from two sources:

- From the continuous emission monitoring system, which analyses the concentrations of O<sub>2</sub>, NO<sub>x</sub> and SO<sub>2</sub> of the waste flue gases in each gas outlet at the above mentioned time intervals.
- The basic technological variables enter from the decentralized control system of the unit.

The using of simple interpolation methods is feasible only under certain regimes. In other cases the obtained substituted data contains considerable errors. That imposed development of methods for reconstruction of the missing data, which are valid in all the technological conditions. These methods could be classified as follows:

- without using of indirect sources of information – interpolation and/or filtration;
- by using indirect sources of information – regression relations and/or neural networks;
- combined approach – with interpolation and by using of indirect sources of information.

## MATHEMATICAL MODELS FOR PEMS

Detailed analysis of the predictive abilities of eight types of mathematical models is provided. They describe the relationships between the O<sub>2</sub>, NO<sub>x</sub> and SO<sub>2</sub> concentration in one side flue gases duct as a function of the direct measurements in the same duct (but with a bigger sample time), the direct measurements of gases concentration in the neighbor duct and all available additional measurements received from SCADA or DCS.

The investigated mathematical models are given in Table 1.

The evaluation of the investigated mathematical models accuracy is accomplished by calculation the relative mean square error (MSE) ( $\delta$ ) between predicted and directly measured values for the same discrete time. The following formulas are used:

$$\delta = \frac{\sigma_c}{m_c} 100, \% \quad (1)$$

$$\text{where } \sigma_c = \frac{1}{N-1} \sum_{i=1}^N \left( \hat{C}_i - C_i^e \right)^2,$$

$$m_c = \frac{1}{N} \sum_{i=1}^N C_i^e,$$

where N is the size of the test data set;  $\hat{C}_i$  are the predicted values of the gas concentrations obtained from the mathematical models and  $C_i^e$  are the corresponding measured values.

Table 1: Investigated mathematical models

№	Description of the used sources of information and the type of mathematical relationships	Notation
1	Equal concentrations in the output ducts A and B	MA1
2	Definition of the concentration using direct measurements and interpolation	MA2
3	Correction using the ratio between “wet” oxygen concentrations in the output ducts A and B	MA3
4	Mathematical models derived on the base of direct measurements	MA4
5	Hybrid mathematical models (direct measurements)	MA2+MA3
		MA2+MA4
		MA2+MA3+MA4
6	Model based on the ratio “front”/“back” oxygen concentration	MB1
7	Mathematical models on the base of control actions and states - Regressive models - Neural Network models	MB2
		MB21
		MB22
8	Combined mathematical model	MA2+MB2

The most appropriate mathematical models for development of predictive monitoring systems are presented and discussed in the following sections.

## MATHEMATICAL MODELS USING DIRECT MEASUREMENTS

### Concentration determination by interpolation of direct measurements (Model MA2)

Direct measurements only from CEMS during the corresponding sample time are used depending on the load of the power station. When one, two or three units work simultaneously direct measurements in each one of the gas outlets “left” (A) and “right” (B) are carried

out respectively every 5 min, 10 min or 15 min. The opportunity to reconstruct the missing measurements in order to provide a set with 2.5 min sample interval by interpolation is investigated.

The reconstruction of the missing data for the gas concentrations is accomplished using Lagrange interpolation polynomial. The interpolation approach defines a function  $f(x)$ , which pass through given number of experimental measurements  $A_i$  (Fig. 1). Using this function the missing values  $B_1, B_2, B_3, B_4$  and  $B_5$  of the relevant component  $y(x_i)$  are calculated.

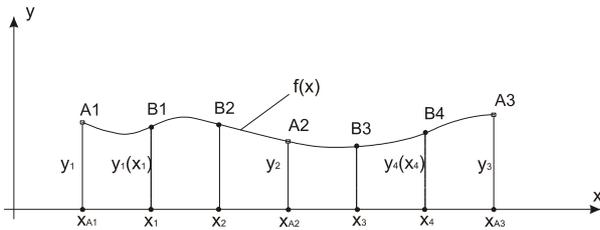


Figure 1: Interpolation procedure

The interpolation could be realized in the following way:

- depending on the number of the points used for interpolation (2, 3, 4,...);
- with averaging or not the interpolated values, using or not moving horizon of interpolation.

During the reconstruction of the missing values of  $NO_x$  the relative mean square error ( $\delta$ ) increases linearly depending on the sampling time. The results for  $\delta$  are presented in Table 2.

Table 2: Summarized interpolation results

Number of the working units	One unit $T_0 = 5 \text{ min}$	Two units $T_0 = 10 \text{ min}$	Three units $T_0 = 15 \text{ min}$
$\delta, \%$	3.48	5.45	6.79

### Using of «wet» oxygen (Model MA3)

The information sources are:

- specially installed sensors on the base of zirconium oxide ( $ZrO_2$ ) in each of the gas outlet ducts A and B, measuring continuously the concentrations of the «wet» oxygen  $O_2^{WAe}$  and  $O_2^{WBe}$ ;
- the Continuous Emission Monitoring Systems (CEMS).

The missing concentrations of the gas components  $O_2$ ,  $NO_x$  and  $SO_2$  in the gas outlet duct, where there is no current measurement when in the other gas outlet duct of the same unit there is measurement by the Continuous

Emission Monitoring Systems (CEMS), are using linear relation. For example the missing concentration of  $NO_x^B$  in the outlet duct B is estimated using the following expression:

$$\hat{NO}_x^B = \frac{O_2^{WAe}}{O_2^{WBe}} NO_x^{Ae} \quad (2)$$

In Table 3 the average values of the relative mean square error  $\delta$  for the predicted estimations of  $NO_x$  and  $SO_2$  obtained from the multitude simulations are presented.

Table 3: Average values of  $\delta$

Gas	$NO_x$	$SO_2$
$\delta, \%$	5.77	1.68

### Mathematical models using the direct measurements in the gas outlet ducts A and B (Model MA4)

This mathematical model is created on the base of the data from the Decentralized Control System (DCS) and the Continuous Emission Monitoring Systems (CEMS).

The corresponding concentrations of the gas components in the ducts A and B  $O_2^{Ae}$ ,  $NO_x^{Ae}$  and  $SO_2^{Ae}$  respectively  $O_2^{Be}$ ,  $NO_x^{Be}$  and  $SO_2^{Be}$  are measured in accordance with the the set sequence for analysis. For calculation of the gas emission vector in the duct B  $\hat{C}_B(k)$ , when only the emissions in the duct A are measured, the next relation is used:

$$\hat{C}_B(k) = f(V^L, V^R, O_2^{fe}, O_2^{be}, T, O_2^{Ae}, O_2^{WBe}, NO_x^{Ae}, SO_2^{Ae}) \quad (3)$$

The dependence for the emissions in the duct A is determined in analogous way.

The relative errors for prediction of the gas emissions from the mathematical models are given in Table 4.

Table 4: Relative errors of the gas emissions prediction

$\delta, \%$	$O_2$	$NO_x$	$SO_2$	$\bar{\delta}$
Exit flue A	3.46	3.22	3.74	3.47
Exit flue B	3.36	3.29	3.59	3.41

The experimental values (\*) and the predictive values of the concentrations of  $NO_x$  in the duct B, where  $\delta=3.29\%$ , are illustrated at Figure 2.

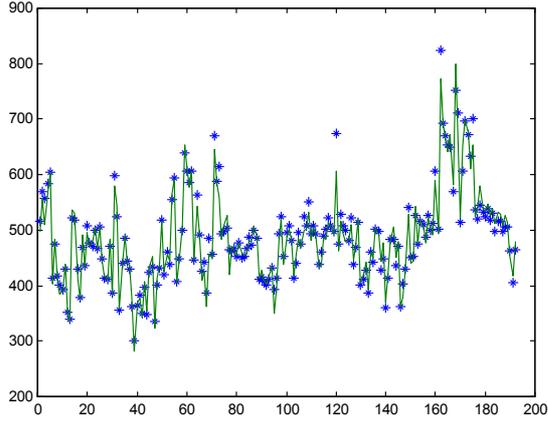


Figure 2: Experimental and predicted values of the  $\text{NO}_x$  concentrations

### MATHEMATICAL MODELS BASED ON THE UNIT STATE AND CONTROL DATA (MODEL MB2)

The development of these mathematical models is based on the data from the unit Decentralized Control System. It's not necessary ecological monitoring system to be available. The physical insight considerations and the preliminary investigations of the models have shown their sensibility to the total air flow rate  $V$  and to the fuel flow rate  $B$ . The mathematical models are individually derived for prediction of the gas concentrations in each one of the exit ducts A and B. For example the created model for exit duct A has the following form:

$$\hat{C}_A(k) = f(B^L, B^R, V^L, V^R, O_2^{fe}, O_2^{be}, T) \quad (4)$$

where  $\hat{C}_A$  is a vector of the predicted concentrations of gas components  $\hat{O}_2^A$ ,  $\hat{NO}_x^A$  and  $\hat{SO}_2^A$ .

All the variables in the right part of the equation (4) are measured continuously and are defined as follows:

$B^L$  and  $B^R$  are the fuel flow rate correspondingly in the left and in the right combustion chamber;

$V^L$  and  $V^R$  are respectively the flow rate of the total air in left and right air duct;

$O_2^{fm}$  and  $O_2^{bm}$  are the measured concentrations in the front and the back part of the corresponding combustion chamber;

$T$  is the temperature before the secondary steam superheater.

Relation approximations by regressive models (Model MB21) and neural network models (Model MB22) are studied. The average values of the unit emissions of both flue gas ducts are accepted as representative value.

### Regressive models (Model MB21)

In Table 5 the relative mean square errors ( $\delta$ ) of the predicted values of the gas concentrations, given from the regressive model (Model MB21) in comparison to the real experimental data are shown.

Table 5: Results from the regressive model

Gas	$O_2$	$NO_x$	$SO_2$
$\delta, \%$	3.22	6.91	4.07

### Neural network models (Model MB22)

According to the dependence (4) mathematical models based on neural networks (Model MB22) for prediction of the concentrations of  $O_2$ ,  $NO_x$  and  $SO_2$  in each of the gas outlet ducts A and B are developed. The highest prediction accuracy is achieved by using two layers neural networks of Cascade-Forward Backdrop type with sigmoid transfer function in the hidden layer and with linear one in the output layer.

The relative mean square error ( $\delta$ ), comparing the predicted values of the gas concentrations towards the real experimental data, are given in Table 6.

Table 6: Results from the neural models

Gas	$O_2$	$NO_x$	$SO_2$
$\delta, \%$	2.83	6.60	2.92

### HYBRID MATHEMATICAL MODELS

Combining different types of modeling techniques the total estimation accuracy could be improved with 10-15% (Hadjiski 1999). Extensive simulation studies on the possibilities of combining the considered mathematical models and other created mathematical models in hybrid ones are carried out. The aggregation is performed by weighted summation of the models outputs. Under aggregation of two mathematical models the predicted concentration of the analyzed gas component  $\hat{C}_i$  obtained from the hybrid model is calculated as follows:

$$\hat{C}_i = \alpha_i \hat{C}_{1i} + (1 - \alpha_i) \hat{C}_{2i} \quad (5)$$

By analogy at aggregation of three mathematical models the expression for  $\hat{C}_i$  is:

$$\hat{C}_i = \alpha_i \hat{C}_{1i} + \beta_i \hat{C}_{2i} + (1 - \alpha_i - \beta_i) \hat{C}_{3i} \quad (6)$$

The index  $i$  in the expressions (5) and (6) is addressed to one of the gases  $O_2$ ,  $NO_x$  or  $SO_2$ . The predicted concentrations of the considered gas on the individual mathematical models are denoted by  $\hat{C}_{1i}$ ,  $\hat{C}_{2i}$  and  $\hat{C}_{3i}$ .  $\alpha_i$  and  $\beta_i$  represent weight coefficients, varying in the interval  $[0,1]$ .

The selection of the combination of the individual mathematical models, which are investigated to be aggregated, is accomplished taking into account such criteria as accuracy, reliability of the measurement and the costs minimization for sensors and analyzers of the gas components.

A procedure for aggregation of the mathematical models, based on the methods of the nonlinear programming is proposed. In this procedure the independent variables are  $\alpha_i$  and  $\beta_i$ . The objective function is equation (1) of the relative mean square error  $\delta_i$  for the concentration  $\hat{C}_i$ , estimated by using expressions (5) or (6). The suggested procedure is developed on the base of parallel simulation computations of the individual models for test data sets completed with experimental data of the gas concentrations and subsequent aggregation of the models. On Figure 3 a graphic illustration of created hybrid mathematical model, applying the proposed aggregation procedure is presented.

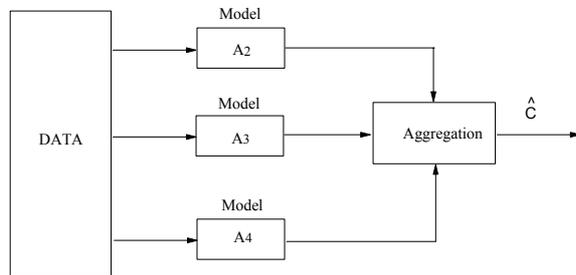


Figure 3: Hybrid mathematical model

Numerous statistical analyses have shown the dominant efficiency of the next three combinations of models used direct measurements:

1. Interpolation model (MA2) and model based on the ratio of the oxygen content in flue gas ducts A and B (MA3)
2. Interpolation model (MA2) and model based on the direct measurements (MA4)
3. Interpolation model (MA2), model using “wet” oxygen (MA3) and model based on the direct measurements (MA4).

A variety of combinations of models MA2, MA3 and MA4 with the models MB1 and MB2 are possible. The carried out analysis has shown that the hybrid models of type MA2+MB2 are more perspective for the purposes of the ecological monitoring.

The results of the accuracy evaluation of the individual models and of the hybrid models are summarized in Table 7. The investigated models are ranged according to their accuracy.

Table 7: Ranging of the models on their accuracy

Notation	Mean square error $\delta$ , %	Range position
MA1	10.7	12
MA2	6.79	10
MA3	3.72	2
MA4	3.44	1
MA2+MA3	6.43	9
MA2+MA4	6.33	8
MA2+MA3+MA4	5.4	6
MB1	9.7	11
MB21	4.73	4
MB22	4.12	3
MA2+MB21	5.91	7
MA2+MB22	5.14	5

## CONCLUSIONS

The simulation results using developed algorithms for reconstruction of the missing data of the flue gases concentration could be summarized as follows:

1. The methods using direct measurements guarantee higher accuracy.
2. The ranging of the methods on the average relative quadratic error, presented in Table 7, shows distinguish priority of the methods based on models using direct measurements (MA4) and on measuring the “wet” oxygen (MA3).
3. The results obtained from the model based on measuring of “wet” oxygen (MA3) depend only on the direct measurement data and they are not influenced by the operational conditions. That permits these algorithms to be used in the whole range of technological regimes, namely static, quasi-static and dynamic, without applying of any adaptive procedures.
4. If during the exploitation of the “wet” oxygen sensor some difficulties of fouling and similar ones appear, which decrease the measuring precision, the interpolation model (MA2) will

remain to be efficient with the previous accuracy.

5. The hybrid mathematical model (MA2+MA3+MA4) (Figure 3) guarantee high accuracy. The eventual automatic elimination consequently of the models MA3 and MA4 will maintain the efficiency of the emission monitoring system. Thus the hybrid model (MA2+MA3+MA4) possesses robust properties with respect to the measuring conditions.
6. The models using data from the Decentralized Control System are more important in solving the task of optimization of the combustion process in the steam generators under the ecological restrictions for the gas emissions and when Continuous Emission Monitoring Systems (CEMS) is not available.

#### ACKNOWLEDGMENTS

This paper is supported in part by the contract No. TH1408/04 of the Bulgarian National Science Fund.

#### REFERENCES

- Alberta Environmental Protection. 1998. "Continuous Emission Monitoring System (CEMS) Code". Environmental Service.
- Bulgarian Ministry of Environmental Protection. 1999. *Regulation №6: Measurements of the Air Pollutants from Industrial Sources* (in Bulgarian).
- Bulgarian Ministry of Environmental Protection. 2003. *Instruction for the Requirements for Registration, Processing, Storage, Presentation and Evaluation of the Results from the Continuous Emission Monitoring Systems* (in Bulgarian).
- Code of Federal Regulations, Part 60, Appendix A.
- Code of Federal Regulations, Part 75, Appendix E.
- Cretnik, J.; M. Slibar; M.Vedenik-Novak; and M.Hocevar. Continuous Emission Monitoring Systems in Termoelektrarna Toplarna Ljubljana. <http://www.raci.si>.
- Eberhard, W.H. 1995. "Design of an Alternative Emissions Monitoring System for Clean Air Act Compliance". *Power-Gen '95*, Anaheim, CF.
- Ekhus, L. and L.Black. "NOx Predictive Emissions Monitoring Systems at a TMP Mill". <http://epa.gov.ttn/emc/cem/pems.pdf>.
- EPA Handbook. 1997. *Continuous Emission Monitoring Systems for Non-criteria Pollutants*. EPA/625/R-97/001/.
- Ghien, T. W.; H. Chu; W. C. Hsu; T. K. Tseng; C. H. Hsu; and K. Y. Chen. 2003. "A Feasibility Study on the Predictive Emission Monitoring System, Applied to the Asinta Power plant of Taiwan Power Company". *Journal of the Air&Waste Management Association*, Vol. 53.
- Jimenez, A.; M. Estela; B. Benoit; F. Peledan; T. Abbas; and J. L. P. Nieto. "Advance Predictive Tool to Optimize Combustion and Emission Performance". Cordis, RTD-Projects, <http://dbs.cordis.lu>.
- Hadjiski, M. 1999. "Hybrid Modeling and Inference Model-Based Control of Complicated Technological Plants". *Studies in Informatics and Control*, Vol. 8, No.1.
- Joseph, J. and I. Macak. 1996. "The Proc. and Cons. of Predictive, Parametric and Alternative Emission Monitoring Systems for Regulatory Compliance". *Proc. of the Air and Waste Management Association Conference*, Nashville, TN, USA.
- Topical Report №17. 2001. *CLEAN Coal Technology, Software Systems in Clean Coal Demonstration Projects*.
- United States Environmental Protection Agency (EPA). 1992. *Continuous Emission Monitoring Systems*. September.
- United States Environmental Protection Agency. 1994. *Calculation and Interpretation of Accuracy for Continuous Emission Monitoring Systems (CEMS)*.
- US Environmental Protection Agency. 2000. *Continuous Emission Monitoring – Information, Guidance, etc.* Technology Transfer Network Emission Measurement Center, Final report.

# **Computer Games and Simulation**



# INTERACTION-BASED APPROACH FOR GAME AGENTS

Damien Devigne

Philippe Mathieu

Jean-Christophe Routier

LIFL - CNRS UMR8022

Université des Sciences et Technologies de Lille

59655 Villeneuve d'Ascq Cédex - FRANCE

## KEYWORDS

Cognitive agents, Agent model, Believable behaviour, Simulation design

## ABSTRACT

Most often, agents in simulations are based on reactive models. Such systems do not plan actions for the agents and are too limited to express complex realistic behaviour. We propose here an agent model for spatially situated simulations, like computer games are. The involved agents are cognitive (or deliberative) ones: they are able to build plans and to adapt them according to the dynamics of the simulation. Our main goal is to obtain believable behaviours for the agents in simulations.

Thus we propose a generic model for cognitive situated agent in simulations of which video games are a typical example. The proposed ideas promote re-usability from one simulations to another and then favour good software design.

The two main problems can easily (and without surprise) be identified: *how to represent knowledge ?* and *how to build plan using this knowledge*. To solve the first we propose to describe the laws that manage the simulated world in term of interactions that can be performed by some agents and suffered by others. Concerning the second problem, we propose a planning algorithm that is based on the interactions and take into account the facts that agents are situated and that plans must be executed in a situated environment that is in permanent evolution. Thus the plans are actually incrementally built through partial replanning.

## INTRODUCTION

The simulation of rational or believable behaviour is, quite from the beginning of Computer Science, one of the major objectives of this field, especially in the purpose of Artificial Intelligence, ie. to reproduce the intellectual abilities of the human being. This issue has been initially addressed from a logical and linguistic viewpoint, which raises huge difficulties. In addition, it appeared rapidly that a large number of AI *applications* did not require a human-like intelligence level.

Now this is not still true. New research fields need a human-like level AI, not in order to *solve complex problems*, but rather to *develop harmonious interactions* with human partners: for instance, social robotics (Brooks and al. 1999), the use of virtual reality in teaching or training, or the large domain of video games (Nareyek 2004, Magerko, Laird, Assanie, Kerfoot, and Stokes 2004) are illustrative examples.

According to J. Laird, the latter constitutes a “Killer Application” for human-level AI (Laird and van Lent 2000). The characters involved in video games have indeed to be perceived as autonomous entities with increasing realistic behaviours. They have to be *convincing*, thus their behaviour must comply with the rational expectations of their partner or opponent human players. They also need to adapt to new situations, acquire additional abilities throughout the game, etc. In addition, team strategies are also often useful. In order to develop such kind of interactions, the agents have to make the human observer thinks that, in order to achieve their goals, they behave in an “intelligent”, “rational” way, ie. like the human would have behaved. Our research aims at this goal : modelling believable characters for simulations in general and games in particular. Let us precise at this point that we do not consider here the problem of the simulation of “emotions” (Allbeck and Badler 2003), but consider “simulation” in the sense of “simulation of sequence of actions”.

In the case of video games, theses “cognitive” constraints meet additional “economical” ones: the time needed for developing the game. This depends to a large extent on the reusability of previous works. In the case of character’s AI, it is often difficult to reuse from one game to another or even, inside a game, from one character to another. This is mainly due to the almost systematic use of scripts whose drawbacks have been many times underlined (Tozour 2002), and that are only partially solved with dynamic scripting (Spronck, Sprinkhuizen-Kuyper, and Postma 2004).

More generally, this domain of modelling believable characters combines difficulties that can be encountered in classical AI (knowledge representation), in distributed AI (coordination of agents having most of the time different individual goals), and in Software Engineering

(reusability of conceptual and software tools).

Some propositions have been done concerning agents and games (Nareyek 2000), and most of them concern reactive agents (Niederberger and Gross 2003). But reactive agents, while effective in several cases, offer limited behaviours. Indeed their behaviours are “short term directed” and not “goal oriented”. Their ability to perform some tasks depends on the immediate surroundings and does not result of wilful acts. Our proposition aims at offering cognitive (or deliberative), driven by goals, proactive agents. Let us precise that we do not consider the interesting problem of behaviour’s learning (Ponsen and Spronck 2004).

We promote a generic approach that assumes that a single formalism can be used to design realistic (ie. believable) behaviours in an artificial world in general and in games in particular. Thus from one simulation to another the cognitive behavioural engine stays the same even if the context changes and the behavioural components can be (partially) reused. The main principle is to base the dynamics of the simulations (and the knowledge representation too) on interactions between agents: some agents can perform interactions and other agents can suffer them. Our main goal is then to provide a uniform and generic frame for simulations. Our target is multi-agent spatially situated simulations like most of computer games are. More precisely role-playing games are a privileged target for our work. Agents are situated in an environment provided by an euclidean space. This space has a “geography” (a “map”) and notions like “position”, “neighbourhood”, “move”, “distance”,... have a meaning. The agents have, at each moment, a partial perception of their environment. This environment is dynamic and concurrent, and thus non monotonic (insofar once an agent knows some data, this knowledge can become wrong - or irrelevant - after some times). The agent’s knowledge about the environment is incomplete and can be wrong. The abilities of the agents can differ. Agents may have cognitive abilities. Some of them have objectives (or goals) that direct their actions in the environment. To achieve its goals each cognitive agent has a behavioural engine. This engine chooses at every moment an action to do. This action must allow the agent to fulfil its goals “at best” and rationally. The “rational” notion of a behaviour is rather subjective and is actually evaluated by a jury that is external to the simulation. Thus, we will consider as rational a behaviour if the decision to perform an action could have reasonably been taken by a human which would have had the same information than the agent.

First section concerns knowledge representation. We first present the environment that models the geography of the simulated world, second the agent model is described. These two points are not sufficient, we must precise how these agents can have an influence on the environment, that is, what the laws that rule the world are. This knowledge representation is crucial since it is used by the agent’s behavioural engine in order to act in the environment. We use what we call interactions to achieve

this. The following section is dedicated to the agent’s behavioural cognitive engine. We present the structure of this engine and more precisely the planning and re-planning algorithm.

## KNOWLEDGE REPRESENTATION

Simulations consist in *agents* that evolve in an *environment* and *interact* with the environment and other agents, according to the laws that rule the environment. Therefore, it is essential to describe these different core notions. We will first present the environment that is the basis of the situated side of the simulations. Second, we define the agents involved in our simulations, they are divided in passive (closer to “things”) and active agents that are responsible of the dynamics of the simulation. These agents can suffer or perform interactions that are described in the third part. They represent the atomic knowledge beans used by the deliberative agent to act.

### Environment

The environment describes the geography of the simulation. It provides the support to situate the agents and then to control the possibility of the realisation of some of their actions, when the notion of neighbourhood has an importance for example. The environment gives a meaning to the notion of *move* for an agent, although it is simple, this notion is full of importance since it impacts a lot on the dynamics, at least the visible one, of the simulation and it makes our concerns different from the pure planning problems. It is the environment too, that is in charge to determine which information can be perceived by an agent.

We represent the environment by a graph where vertices are *places* and edges denote *path* from one place to another.

```
environment := {place*, path*}
path         := (placeorigin,
                placedestination,
                condition)
condition    := boolean expression
```

A place is a geographical elementary area. The granularity of a place depends on the simulation, the only constraint is that inside a place there is no restriction neither for moves, nor for perception (restrictions due to the other agents, like collision problems, excepted). A place can represent a room, a town or any other part of the environment, and inside a place the position of an agent can be managed discretely or continuously depending on needs.

A path denotes an oriented transition between two places. It is defined by the places that it links, and a condition that must be satisfied if an agent wants to use this path. The edge is oriented and the condition to go from some place *a* to a place *b* is not necessarily the same than the one to go from *b* to *a*.

This formalism allows to describe, for example, that a door between two rooms must be opened if we want to

go from one room to the other, or that an agent must be able to swim to cross a river between two fields. In this last case, our approach allows, depending on needs, to choose to model or not the river with a place. It depends on whether the crossing of the river has a meaning in the simulation (see figures 1 and 2).



Figure 1: River is modelled. The paths are:  $(a,r, \text{“agent can swim”})$ ,  $(r,a,true)$ ,  $(b,r, \text{“agent can swim”})$ ,  $(r,b,true)$

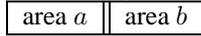


Figure 2: River is not modelled. The paths are:  $(a,b, \text{“agent can swim”})$ ,  $(b,a, \text{“agent can swim”})$ .

The environment is the place where the agents are situated. It plays the role of a reference for the agents (then in this context the environment can not itself be an agent). Each agent is located in a place and can not be in a path. If the path must be put into concrete form, this must be done using a place, like we have seen it with the river example. Then a place is mainly characterised by the set of the agents that belongs to it.

The relative position of the agents inside a place (when this has a meaning) will be managed by the place itself, and is a parameter of the simulation.

## Agents

The agents involved in the simulations we are interested in, are situated in a place of the environment. It is the environment that is in charge of the creation or removal of an agent in the simulation, even if the decision of these creations or removals is the result of the behaviours of the present agents.

We call *agent*, every entity that has some relevance in a simulation, that is, that can have an influence over the simulation. Among these agents, we distinguish two special classes: the *passive agents* and the *(pro-)active agents*. We use the terms of *inanimate* and *animate* agents too. It is for the latter that the notion of behaviour as a meaning.

In a rather natural and classical approach, agents are defined by a set of properties, a property being a pair (*name*, *value*). However, we will refine this definition (see Figure 3) and precise some particular properties imposed to our agents. We spend no time on the *name* property which allows to have a symbolic reference of the agent, but we rather insist on what characterize the agents: their abilities expressed by interactions.

Our agents (passive or active) are, at first, characterised by the actions (in the following we rather use the term “*interaction*” which denotes the way an action is coded) they can suffer. A *tree* agent could be cut, a *door* agent could be opened or painted, a *sheep* agent could be sheared, etc. We name *can-suffer* this property, the associated value is the list of interactions that the agent can suffer (that is for

which he can be a target). The interactions are presented in the next section.

We must now study the particular case of the active agent. It is easy to guess that these agents have the possibility to interact with their environment, that is with the other agents (seen through their “passive” facet). These abilities are expressed by a collection of interactions they can perform, and defined in a property; we name *can-perform* this property.

However, this property remains a declaration of abilities. In order for an active agent to have an impact over the simulation, he must be provided with a behaviour engine that takes at every moment the decision of the action undertaken by the agent, and then of the used interaction. This decision depends on the context. This engine is influenced/directed by the existence of goals for the agent. The section is dedicated to the presentation of this engine.

Confusion must not be made between “active” or “animate” agent and the modelling of “living” entity. Thus, if in a simulation there is a machine which produces regularly some objects *o*, this must be modelled by an active agent whose goal would be the production of agents corresponding to *o* and whose behaviour would be the satisfaction of this goal.

We can point out another particular property: the memory of the active agent. It represents the knowledge base for all the information gathered by the agent concerning the environment: the topology of the environment, the other agents (their position and state), etc. This memory is a degraded environment insofar as it corresponds to the data the agent knows about the environment. This knowledge can be incomplete, for instance the agent does not know that others exist. It can even be wrong, for instance because the agent is not necessarily aware when other agents act and modify the state of some entities.

To come back on what we said at the beginning of this paragraph, an element must be considered as being represented by an agent in a simulation (ie. has an influence over the simulation), if and only if either it is active and the list of the interactions it can perform is not empty, and there exists at least one possible target for one of these interactions, or it is passive and the list of the interactions it can suffer from is not empty, and at least one active agent can perform one of these interactions.

It results from this definition that the interactions play an essential role in our simulations. Agents are different because they perform or suffer different interactions. Moreover the interactions define the “laws” of the simulated environment, and then play a central role in knowledge representation. We now define this notion.

## Interactions

Interactions are the backbone of our simulation model, we could even speak of *interaction oriented simulations*. These interactions are the basis of the knowledge representation in the simulations. They define the laws of the modelled world, that is the actions that can be performed

```

agent      := passive-agent | active-agent
passive-agent := { ( "name", Symbol),
                  ("can-suffer", {interaction* } ),
                  property* }
active-agent := passive-agent U
              { ("can-perform", {interaction* } ),
                ("goals", goal* ),
                ("memory", (degraded) environment),
                ("engine", engine) }

```

Figure 3: Definition of an agent

in the simulations. They are central since the agent’s engine uses them to build the agent’s behaviour.

These interactions are the units of knowledge that describe the laws of the simulated world. They represent a declarative knowledge. A consequence is that an interaction must not, very special case excepted, be attached to one simulation but must represent a rather universal knowledge. This constitutes a difficulty in regards with the representation of these interactions, but allows to reuse them from one simulation to another one.

We characterize an interaction by an *actor* and a *target*. The *actor* is instantiated by an active agent who can perform this interaction and the *target* is any agent who can suffer from this interaction.

An interaction is defined in a rather classical way as presented in figure 4. The *name* is a unique identifier. The other three parts are:

- the *condition*, it tests the current context of execution of the interaction and consists mainly of tests on values of target or actor properties.
- the *guard*, it checks general conditions for the interaction applicability, typically it defines that to be fired an interaction requires that the distance between the target and the actor must be less than some given value.

The guard is separated from the condition since it corresponds to the knowledge due to the geographically situated feature of the simulations. In a non situated context, one would have only the condition and action parts. The guards are at the origin of the moves in the plan, and this is these moves that are indeed specific to situated problems. Thus, we do not express explicitly in an interaction that the agent has a move to do in order to fire it, we want the agent to plan it when required by a guard.

- the *action*, it describes the consequence of the interaction, it can be a change in the state of the actor and/or of the target (ie. a change of the value of a property), and/or the activation of an environment action (like the creation of an agent).

Some interactions does not naturally obey to this schema of interaction between a target and an actor. This is the case, for example, for the “*sleep*” action. However,

in this case it suffices to consider that the actor and the target are the same agent: the actor decides to make the target (himself) sleep, and thus he changes the state of the target. Such action can then be represented with the same interaction model.

More generally the consequence of an action is a change in the state of the target or actor. Thus to *open* an object (door, chest, window, etc.) makes it changing from *opened* state to *closed* one. The precise essence of the target is of no importance here, this knowledge must then be represented in a “universal” way by the interaction:

$$\begin{cases}
 \text{condition} & = \text{“target.opened} = \text{false”} \\
 \text{guard} & = \text{“distance(actor, target)} < 1\text{”} \\
 \text{action} & = \text{“target.opened} = \text{true”}
 \end{cases}$$

Such an action can be used by an engine to generate a plan such that: “*to push a button in the next room I must open this obstacle*” (or more precisely the knowledge would be *the obstacle must be opened*, and when this is not satisfied the given plan is produced). Whether this obstacle is a door or a window or anything else that is openable, the plan remains valid.

A problem arises when considering more “specific” agents. For instance, let us consider the case where the obstacle is a *lockable* door. To push the button, the above plan is still valid with respect to the knowledge that must be used and then with respect to the behaviour engine. The difference exists only in the condition for the execution of the action. This lockable door requires that, in its particular context, something like *target.lock = false* must be satisfy too. Then, from an abstract point of view the plan is still valid, but the *open* interaction must be understood as “make the door change from *closed* to *opened* state when it is *unlocked*”. The problem is then how the same abstract plan (ie. “open the door to push the button”) can receive different solutions (just “open” or “unlock and then open”) depending on the target (whether it is lockable or not). To have two different interactions, one named *open-when-lockable* (or anything else) and the other named *open* is not relevant. As a first consequence this leads to a multiplication of interactions and implies that the active agents must be finely tuned. Moreover, the agent engine must take into account the different possible

```

actor           := the agent who performs the interaction
target         := the agent who suffers the interaction

interaction     := name, condition, guard, action)
name           := Symbol
guard         := d op Integer
d             := distance between actor and target
op            := = | > | < | ≤ | ≥
condition     := test_property | predicate_primitive(args)
test_property  := { actor | target }.property_name op Value
predicate_primitive := primitive
action        := affect_property | primitive
affect_property := { actor | target }.property_name = Value
primitive     := { actor | target }.primitive_name(args)
primitive_name := Symbol
property_name := Symbol

```

Figure 4: Definition of an interaction

cases, although they conceptually represent the same action (*open* here). Thus it is more than probable that we would fall again into one of the major pitfall of the script approach for designing agent's behaviour.

Therefore, our proposition consists in the possibility to specify at the target level (ie. the agents having the considered interaction as a “*can-suffer*” one), the specific process. One can notice that only the nature of the target requires a change in the manipulation, not in the plan. During an interaction between such a target and an actor, the target “tells” to the actor the particular knowledge to be used while interacting with this target. For the actor (active agent), there is still only one generic interaction. Thus, the *can-suffer* property of a *lockable-door* agent contains the *open* interaction, with a specialization of the condition like: *target.locked = true*. Thus, when an actor tries to interact, using *open*, with this door, he gets the full condition “*target.opened = false and target.locked = false*”. This leads the actor to (try to) unlock the door before it can open it. This can be seen as a kind of inheritance for interaction. It offers to the game designer the possibility to add new targets that specifies an existing one in order to take into account some particularity of the simulated world (for instance a *lockable door* that specifies a *door*). And this specification does not require the possible actors to be modified, at least their engine must not be changed. This flexibility eases the design of simulations and the ability to reuse interactions and agents from one simulation to another.

## THE AGENT ENGINE

This section details the engine of the active agents. We first present the structure of the “mind” of the agent and the dependences between the different elements and second the planning and behaviour engine.

## The Agent Structure

The decision cycle applied by the agent is presented in the figure 5. Continuously the agent perceives its environment. The acquired information are forwarded to an update module that can influence the memory and the currently established plan. An action is then chosen and the agent tries to execute it in the environment, and so on. To apply this cycle the agent is provided with a “mind”.

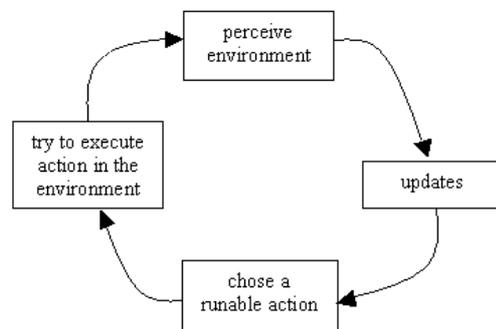


Figure 5: Agent decision cycle.

The “mind” of our active agents is made of several modules: a knowledge or beliefs base, a new information management module, a planning engine, an action selection module and an execution module. The articulation between these parts is illustrated in Figure 6.

### The Knowledge Base.

The knowledge of the agent can be divided in two. On one side, the knowledge about the actions the agent can do and on the other side the knowledge about the environment he belongs to.

The first is defined by the set of all the interactions that the agent can perform. These interactions represent the absolute knowledge of the agent about an environment independently of any given particular context. They are the

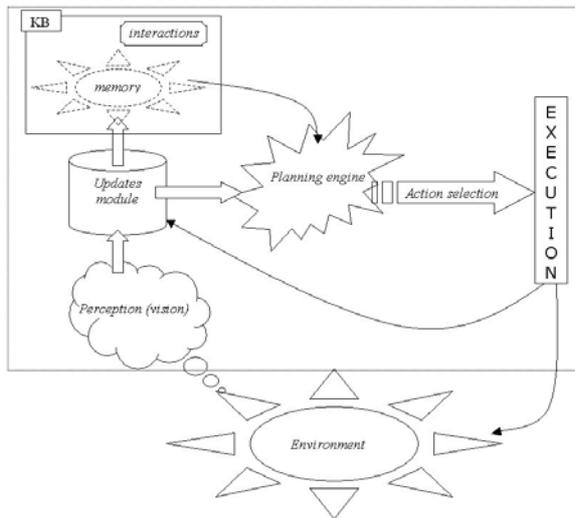


Figure 6: Different elements of agent mind.

basis of the behaviour engine to build plans.

The second corresponds to a base of beliefs and is called the *memory* of the agent. It is a contextual knowledge. It evolves according to the information perceived by the agent. It consists in the knowledge concerning the geography of the environment and in the information about the other agents. The memory is like a degraded environment and corresponds to the perception that the agent has of the environment. In the memory, some of the information can be marked *unknown*. Every known information is timestamped, this helps to estimate a confidence in the data: the older an information concerning the position of a mobile agent is, the less confident it is. The information in the memory are used by the behaviour engine to determine the one among the *can-perform* interactions that must be applied in order to achieve goals. These information are beliefs and not absolute knowledge, consequently when the agent tries to perform an action for which it believes all the conditions are satisfied, it is necessary to check if it is indeed the case in the environment.

#### Perception.

An innate and absolute knowledge of the environment in which the agent evolves will not produce realistic behaviour. Then it is necessary to provide the agent with a way to perceive new information while he is acting. Actually we content on a simple “visual” perception. The agent perceives the information of the environment that are inside its field of view (whose shape and radius can be changed at will). The perceived information are forwarded to the new information management module that is in charge to manage their influence. Since the perception module is only in charge of the perception and not of the treatment of the new information, it is easy to extend it to new kind of perceptions such as sound.

#### The update module.

As we previously say, the new information management module is in charge of the new perceived information. It is a kind of short term memory. It operates on two levels: first the memory in order to update the beliefs, and second the planning engine in order to adapt, if needed, the current computed plan through a partial re-planning. This is detailed in the following.

#### Planning, selection, execution.

These points will be more detailed in the next section. The engine is in charge of the resolution of the objectives of the agent. It uses the *can-perform* interactions of the knowledge base to build a plan of actions according to the memory. The built plan is valid according to the memory but can be wrong in the environment, this is checked at the execution step. This plan determines at every moment which actions the agent can undertake, an action selection strategy is then applied to choose the next effectively fired action. This strategy can be changed from one agent to the other to obtain different behaviours and then different individuality, even if the planning engine is the same. Once the action is chosen, the agent tries to execute it. Either the beliefs of the agent were right and the action is effectively performed in the environment, or they were wrong and the action can not be done and the agent must update its knowledge.

### The Planning

The planning algorithm we use is a kind of backward chaining. To fulfil its goal, among all the interactions that it can perform, the (active) agent searches those that can help to achieve it, and then selects one. If the conditions of this action are satisfied (according to the agent memory), the (inter)action can be fired and the plan is done. Otherwise, the non satisfied conditions become new goals that need to be planned.

#### The goals.

There exist two kinds of goals. First the *interaction-goals*, they correspond to an (inter)action that the agent wants to execute. The target of this interaction can be less or more precisely given: from a named agent to any agent that can suffer the interaction, as shown in the next table:

goal	type of target
eat(apple_12)	a given precise apple
eat(an apple)	any apple
eat(*)	any eatable (ie. “who can-suffer from eat”) agent

Second, the *condition-goals*, they correspond to a condition that the agent wants to bring to true. For instance:

`actor.energy > 100` “having actor energy to be greater than 100”

### Planning tree.

In a rather classical way, the plan produced by the backward chaining can be viewed as a tree. The nodes are made of the different goals and subgoals encountered during the resolution. Some are interaction-goals, others are condition-goals. Thus this tree is an AND-OR tree. AND-nodes correspond to condition-nodes (for condition-goals) and OR-goals to interaction-nodes (for interaction-goals).

*Condition-nodes and interaction-nodes:* A condition-node has sons only if its condition is not satisfied. These sons are interaction-nodes built from the interactions whose action part offers a way to satisfy the condition (or to approach this satisfaction, for example by increasing the energy for the above given condition-goal example). The tree leaves are the satisfied condition-nodes (ie. whose conditions are satisfied).

The interaction-node's sons are built from the conditions that can be found in the condition and guard parts of the interaction: from these, condition-nodes are built. These sons are always built. An interaction-goal is said to be satisfied when all its sons are satisfied, the associated interaction is then declared runnable.

These correspond to the general cases, however since the simulations take place in situated environment, moves must be taken into account. They must receive particular considerations as discussed in (Devigne, Mathieu, and Routier 2004), this leads to introduce *move-nodes*

*Move-nodes and exploration-nodes:* To *move* or to *explore* the environment correspond for the agent to execution of interactions. The associated nodes must then be present in the planning tree as particular cases of interaction-nodes.

The exploration case can be reduced to the move case. To explore the agent must indeed make move towards a chosen location. The existence of the exploration-nodes are justified by the need to choose the targeted position before making the move. The agent must then apply its own exploration strategy to make its choice. Thus in the following we will only concentrate on the move case.

One problem is: what are the condition-nodes sons of a move-node? This problem amounts to ask what are the conditions that must be satisfied to make a move possible. To a move corresponds a computed path that is a sequence of elementary paths presented in the "Environment" section. With these elementary paths come conditions. A move is possible if these conditions are satisfied. With these conditions we create condition-nodes that become the sons of the considered move-node.

### A classical backward chaining.

The planning tree is built according to the algorithm presented (in broad lines) in Figure 7. Every calculus are based on the memory (ie. beliefs base) of the agent. It is in particular the case when the agents checks a condition or computes a path for a move. Therefore, the computed plan is valid according to the agent memory, but can be wrong once it faces up to the reality of the environment.

For the exploration-nodes, the principle is roughly the same once the exploration strategy has provided a place to reach.

Every details are of course not presented in this algorithm. In particular if the same (sub)goal occurs more than one time during the planning, the corresponding node is not expanded twice, it is shared by its fathers. The tree is then an oriented graph.

But, actually, our algorithm builds the plan in an incremental way as we will see in section on replanning. Indeed, according to perceived information, the plan is adapted: the plan's tree is locally modified and not fully rebuilt.

**A small example** To illustrate the different points described in the previous paragraph we will consider a very small and simple example (for instance, we do not consider the *open* interaction). We consider a world with two places/rooms separated by a door *d* (see figure 8), the path between these two rooms has the condition "*d.locked=false*". Four interactions define the laws: *unlock*, *take*, *move*, *push* (see Table 1). In the world are an active agent *a* that can perform these 4 interactions, and three passive agents, the door *d* that can suffer *open*, a key *k* that can suffer from *take* (and can be used to unlock *d*) and a button *b* that can suffer from *push*. The goal of the agent is to push on *b*. The figure 8 presents the planning in two different situations.

<i>unlock:</i>	$\left\{ \begin{array}{l} \text{condition} = \text{"target.locked = true"} \\ \text{guard} = \text{"actor.own(target.key)"} \\ \text{action} = \text{"target.locked = false"} \end{array} \right.$
<i>take:</i>	$\left\{ \begin{array}{l} \text{condition} = \text{true} \\ \text{guard} = \text{"distance(actor, target) < 1"} \\ \text{action} = \text{"actor.own(target) = true"} \end{array} \right.$
<i>push:</i>	$\left\{ \begin{array}{l} \text{condition} = \text{true} \\ \text{guard} = \text{"distance(actor, target) < 1"} \\ \text{action} = \text{"target.pushed = true"} \end{array} \right.$
<i>move:</i>	$\left\{ \begin{array}{l} \text{condition} = \text{conditionsfoundinpath} \\ \text{guard} = \\ \text{action} = \text{"distance(actor, target) < 1"} \end{array} \right.$

Table 1: Definitions of the interactions (adapted - but without distortion - to shorten the example)

As one can see, different plans are obtained depending on the context. Moreover in this case, this is because the agents are situated in the environment that two different plans exist. Indeed, it is because the actor must move near the button in order to push it that the state of the door is of importance. It results that the notions of neighbourhood and distance have a direct influence on the produced plan.

### A partial replanning.

Active agents evolve in a dynamic environment. They establish a plan according to their knowledge, that can prove to be incorrect and then they can then be brought to adapt the computed plan according to new perceived information. These information can be of several types:

```

actor = the agent that builds the plan

this = interaction-node
expand()
  for each condition c in this.condition and this.guard
    newNode = createConditionNode(c)
    this.sons.add(newNode)
    newNode.expand()

this = condition-node
expand()
  If this.condition.isSatisfied()
    then finished
  else
    // gets the action that allows to solve condition
    BA = this.condition.getBackwardAction()
    // the list of can-perform candidates for BA
    LI = actor.getCanPerform(BA)
    for each I in LI
      if I is "moveTo"
        then this.sons.add(createMoveNode())
      else
        // list of known agents that can suffer I
        lAgents = actor.getKnownAgents(I)
        for each a in lAgents
          if actor.execute(I,a) satisfied this.condition
            newNode = createInteractionNode(I,a)
            this.sons.add(newNode)
            newNode.expand()
          end if
        end if
      if this.sons.isEmpty()
        then this.sons.add(create-exploration-node)

this = move-node
expand()
  path = actor.computePath()
  If path = null // no path found
    then this.sons.add(createConditionNode("false"))
  else
    // list of conditions≠"true" on paths
    conditionsList = path.getPathsConditions()
    If conditionsList.isEmpty()
      then this.sons.add(createConditionNode("true"))
    else for each condition c in conditionsList
      newNode = createConditionNode(c)
      this.sons.add(newNode)
      newNode.expand()

```

Figure 7: Node's expansion algorithm.

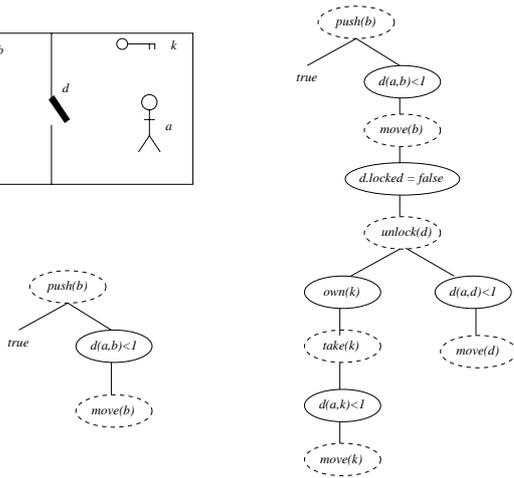
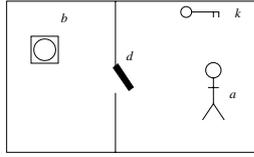


Figure 8: An active agent  $a$  is situated in an environment where are also 3 passive agents: a door  $d$ , a button  $b$  and a key  $k$ . The goal of  $a$  is to push on  $b$ .  $a$  must build a plan to achieve it. A plan can be drawn as a tree, nodes due interaction-goals are drawn with dashed lines and nodes due to condition-goals with solid lines. Depending on the execution context, different plans can be obtained. Left is the tree obtained when  $d$  is not locked and right is the case where  $d$  is locked.  $a$  must adapt its plan to the context.

- *a new information*: the agent learns that a so far unknown information exists. It is the case when the agent sees a new place, meets another agent for the first time, gets a new goal, discovers a condition on a path of the graph, etc.
- *a modification of an existing information*: it concerns mainly modifications about the state of known agents, a property value change or a position change. The position change information covers three situations:

known→known:	we thought agent at a position and we see it at another
known→unknown:	we thought agent at a viewed position and it is not there
unknown→known:	we did not know where the agent were and we now see it

To each of these situations is associated an event that describes what must be modified (changed or added) in the agent memory. The table 2 lists these events, one can easily guess what they are according to the previous paragraphs.

These events are transmitted to the update module who is in charge to take them into account and to consider their impact. First, the memory of the agent must be modified: either a new knowledge is added, or an existing data must be corrected. Second, because of the changes, it is possible that the currently established plan must be adapted. It is the new information management module that is in

<i>new information</i>	NewGoal, NewPlace, NewPathCondition, NewAgent, NewInteraction
<i>modification</i>	AgentModified, AgentMovedKK, AgentMovedKU, AgentMovedUK

Table 2: List of events for new information

charge of forwarding these information to the planning engine.

However, a new information concerns only a portion (even none) of the planning tree. Therefore, it is neither reasonable, nor efficient, to rebuild a new plan for every new information. Indeed, even if it is established that, in theory, no efficiency gain can be guaranteed while using plan reuse rather than new plan generation (Nebel and Koehler 1993), in practice improvements can be expected. Indeed, our context of dynamic simulations corresponds to the case where the agent perceives very frequently slight changes of its knowledge base.

In particular, this is due to the fact that the agent engine uses uncertain information: the planning is based on the information that are in the memory. But, since simulations occur in open dynamic environments, then the built plans are correct with respect to the memory of the agent, but can be wrong once confronted with the real environment, at the execution step. Therefore only partial and local adaptations can be expected in most of cases. Our experiments confirm that.

Our approach is then to top-down propagate events from root to leaves in the planning tree. Each node checks if it is concerned by each event, first because it is the good type and second because additional conditions are satisfied. Checking these conditions is very fast, therefore propagation costs not too much time and in particular less than a replanning when events have no impact. Thus, only the appropriate nodes are updated (collapsed, re-unfolded, adding or removal of subnodes, etc.).

It would probably be a bit tedious to enumerate all the cases and conditions for which a node is affected by an event. Thus we give only two examples. For instance, a new information that affects the graph topology or agent positions can (but not systematically) have an impact on move-nodes. Indeed a new path can “appear” or at the opposite a computed path can become blocked, in these cases, the move-node should be re-expanded. In a similar way, a new met agent can affect a condition-nodes. For instance this can be the case if this agent can be the target of an interaction that helps to solve the condition (that is one of the interactions stored in the  $LI$  list seen in the algorithm presented in Figure 7). In this case a new interaction-subnode must be added and unfolded. Other cases are similar.

Using this principle, there is no need to recompute the plan at each step. It is indeed probable that only few new information occur each step, and this is not necessarily of importance for the agent. But this principle helps the agent to remains reactive too and to adapts as soon as it

is necessary and relevant. Moreover this partial and local tree modification leads the plan to be incrementally built, change after change.

## CONCLUSION

The design of simulations of behaviours that will be perceived and evaluated as believable by a human external observer is not an easy problem. The solutions proposed by the reactive systems are not generic and reusable enough.

We endeavour to propose a general model for the simulation of behaviours. The dynamics of this model relies on the description of interactions that can be performed or suffered by some agents that are situated in a dynamic environment. An individual generic behavioural engine uses these interactions to propose a plan of actions to the deliberative agents.

A natural application of this work are computer games. Targeted games could be action and role-playing games, where one needs complex believable non-player characters to increase the quality of the simulated world and the interactions of the player with it.

Our current work concerns the application of this model to team of agents (Devigne, Mathieu, and Routier 2005). It is clear that it is a challenge for simulations, and games in particular, to be able to design groups of characters that act together to fulfil common objectives.

## AUTHOR BIOGRAPHIES

**DAMIEN DEVIGNE** is Ph.D. student at the University of Lille, in the SMAC research team (<http://www.lifl.fr/SMAC>) LIFL (UMR CNRS 8022) laboratory. He works on simulation of teams of cognitive agents in a geographical environment. E-mail: [devigne@lifl.fr](mailto:devigne@lifl.fr), personal web-page: <http://www.lifl.fr/~devigne>.

**PHILIPPE MATHIEU** is professor at the University of Lille. He receives his Ph D. in 1991. He is leader of the SMAC research team of the LIFL laboratory. He is specialized in multi-agent systems, modelization of behaviours and game theory. E-mail: [mathieu@lifl.fr](mailto:mathieu@lifl.fr), personal web-page: <http://www.lifl.fr/~mathieu>.

**JEAN-CHRISTOPHE ROUTIER** is assistant professor at the University of Lille. He receives his Ph D. in 1994. He is member of the SMAC research team of the LIFL laboratory. His research centers of interest cover agent platforms, agent-oriented software engineering, and recently the design of believable behaviours for simulations (like games). E-mail: [routier@lifl.fr](mailto:routier@lifl.fr), personal web-page: <http://www.lifl.fr/~routier>.

## REFERENCES

- Allbeck, J.; and N. Badler. 2003. "Representing and Parameterizing Agent Behaviors," in *Life-like Characters: Tools, Affective Functions and Applications.*, ed. by H. Prendinger, and M. Ishizuka, 19–39.
- Brooks, R. A.; and al.. 1999. "The COG Project: Building a Humanoid Robot," in *Computation for Metaphors, Analogy, and Agents*, vol. 1562 of *LNCS*, 52–87. Springer.
- Devigne, D.; P. Mathieu; and J.-C. Routier. 2004. "Planning for Spatially Situated Agents in Simulations," in *Proceedings of the 2004 IEEE/WIC/ACM International Joint Conference IAT'04*, 385–388.
- Devigne, D.; P. Mathieu; and J.-C. Routier. 2005. "Team of cognitive agents with leader : how to let them autonomy," in *Proceedings of 2005 IEEE Symposium on Computational Intelligence and Games*.
- Laird, J. E.; and M. van Lent. 2000. "Human-level AI's Killer Application: Interactive Computer Games," in *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence.*, 1171–1178. AAAI.
- Magerko, B.; J. Laird; M. Assanie; A. Kerfoot; and D. Stokes. 2004. "AI Characters and Directors for Interactive Computer Games," in *Proceedings of the 2004 Innovative Applications of Artificial Intelligence Conference*, ed. by A. Press.
- Nareyek, A.. 2000. "Intelligent Agents for Computer Games," in *Computers and Games, Second International Conference, CG 2000. LNCS 2063.*, 414–422.
- Nareyek, A.. 2004. "Artificial Intelligence in Computer Games - State of the Art and Future Directions," *ACM Queue*, 1(10), 58–65.
- Nebel, B.; and J. Koehler. 1993. "Plan modification versus plan generation: A complexity-theoretic perspective," in *Proceedings of of the 13th International Joint Conference on Artificial Intelligence (IJCAI)*, 1436–1441.
- Niederberger, C.; and M. Gross. 2003. "Hierarchical and Heterogenous Reactive Agents for Real-Time Applications," in *Proceedings of the Eurographics 2003, Computer Graphics Forum, Conference Issue*, ed. by P. Brunet, and D. Fellner, 323–331. Blackwell Publishing, UK.
- Ponsen, M.; and P. Spronck. 2004. "Improving Adaptive Game AI with Evolutionary Learning.," in *Proceedings of Computer Games: Artificial Intelligence, Design and Education (CGAIDE 2004)*, ed. by S. N. Quasim Mehdi, Norman Gough, and D. Al-Dabass, 389–396.
- Spronck, P.; I. Sprinkhuizen-Kuyper; and E. Postma. 2004. "Enhancing the Performance of Dynamic Scripting in Computer Games," in *Entertainment Computing - ICEC 2004*, ed. by M. Rauterberg, no. 3166 in *Lecture Notes in Computer Science*, 296–307. Springer-Verlag.
- Tozour, P.. 2002. "The Perils of AI Scripting," in *AI Game Programming Wisdom*, ed. by S. Rabin, 541–547. Charles River Media.

# ILMG GAME: LEARNING ARRANGEMENTS AND SIMULATION SCENARIOS

Robert W. Grubbström  
Department of Production Economics  
Linköping Institute of Technology  
SE-58183 Linköping, Sweden  
E-mail: [rwg@ipe.liu.se](mailto:rwg@ipe.liu.se)

Galina Merkurjeva\*  
Jana Bikovska\*\*  
Department of Modelling and Simulation.  
Riga Technical University  
Kalku Street 1, Riga, LV – 1658, Latvia  
E-mail: [gm@itl.rtu.lv](mailto:gm@itl.rtu.lv)\*, [jana@itl.rtu.lv](mailto:jana@itl.rtu.lv)\*\*

Jens Weber  
Management Simulation Systems AB  
Repslagaregatan 13, Norrköping, Sweden  
E-mail: [jens.weber@ilmg.com](mailto:jens.weber@ilmg.com)

## KEYWORDS

Business simulation game, simulation scenarios, learning arrangements.

## ABSTRACT

The modern logistics simulation game ILMG (International Logistics Management Game) developed by Robert W. Grubbström (Grubbström, Bikovska, 2004) is a business simulation game that simulates a number of companies competing with different products in different markets. The paper presents advanced ILMG features related to development of the game simulation scenarios and learning arrangements. Some advantages and shortcomings as well as the most preferable application areas and circumstances under which game could be used are analysed as well. The game allows participants to become managers of the virtual company and to try out the consequences of various logistics decisions and to explore their links to other core (marketing, financial, etc.) decisions. Game manager or/and an educational course leader needs to create a simulation scenario and provide correspondent learning arrangement according to the course goals and its participants' academic background.

## INTRODUCTION TO THE INTERNATIONAL LOGISTICS MANAGEMENT GAME

Nowadays simulation games are successfully used in education and professional training. Due to such characteristics as an interactivity, dynamic nature, and accessibility, they are more attractive for players and also more effective in many cases than other training approaches. Nevertheless, the success of each individual training process very much depends on how that simulation game is developed and used in training.

The above-mentioned ILMG is a computer simulation and network business game covering different business areas such as marketing and distribution, production

and purchasing, locations and inventories - that is, about logistics and what it means for the company's success. So, it is especially suited to help participants appreciate the general management of a complete company, its main functions and how they interact. The scenario, the objectives of a game are set by game management. Thus, a game can be constructed to focus on functions or aspects of special interest to individual management courses.

In general a business simulation game is a contrived situation which imbeds players in a simulated business environment, where they must make management-type decisions from time to time, and their choices at one time generally affect the environmental conditions under which the subsequent decisions must be made. Further, the interactions between decisions and environment are determined by a refereeing process that is not open to argument from the players.

The ILMG simulates a number of companies or "corporations" competing with different products in different markets. These companies establish corporate units (which may have different functions) in one or more regions. Some units may be production plants, others may be units specialised in selling products (distribution center). Products may be transported between the regions in different transport modes either regular or express. Each corporation is represented by a team of players having access to the Game via an Internet terminal (Grubbström and Bikovska, 2004). The number of companies participating in the game is arbitrary and companies compete in different regions of the world. The game has a strong focus on logistics, but it is not only logistics; it is the meaning of logistics in the context of successful business.

In general computer simulations provide the following advantages that could be successfully used in the learning process (Fripp, 1993):

- Help understanding of complex problems;
- Increase learning motivation;

- Support team building;
- Provide a risk free environment for experiential learning;
- Provide a variety of situations;
- Partly close the gap to reality;
- Facilitate active learning.

Extensive use of simulation games has advantages when course objectives include (Mitchell):

- Allowing participants to experience more realistically the roles and responsibilities of a top decision-makers who are trying to position an organization in a tough, competitive environment;
- Allowing participants to experience the uncertainties and surprises by the unpredictable actions of competitors;
- Facilitating effective aspects learning;
- Promoting participants' emotional arousal and involvement.

The following aspects have to be taken into account while implementing those games in practice:

- Learners background has to be taken into account;
- Complexity has to be adjusted to learning environment;
- Game should be "suitable" for the particular learning problems and goals defined;
- Terminology has to be close to one used by learners.

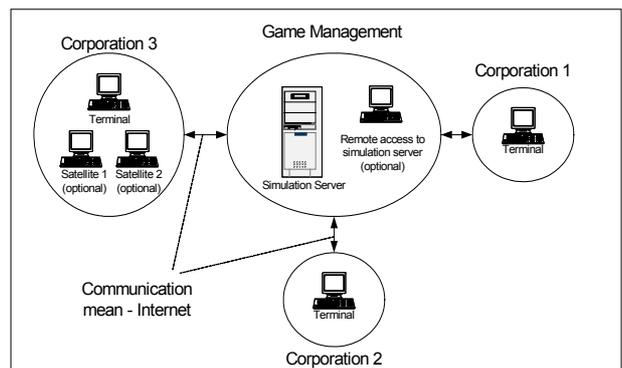
Besides above mentioned the ILMG has the following advanced features that make it unique among many others business simulation games:

- Internet-based that provides possibility of distance learning. Internet is a communication channel between educator and student. Game is accessible anywhere, whether participants are located next door or on the other side of the globe. Number of participants is arbitrary.
- Simulation scenario generation provides flexible game environment. It is possible easily make different games scenarios very simple or complex. Continuous time in the game is like in real life.
- Many kinds of reports and overviews such as balance sheets, profit and loss accounts, sales report, are provided to help participants to evaluate performance of their corporations.
- All data can be easily imported to Excel spreadsheets where different kind of calculations can be made in order to gain information to support decision-making.
- Different means of communication between game Management and corporations, between corporations like Chat, Memos are provided that is useful when the game is used in distance learning courses.

- Educational comments can be made by Management of the game or course leader to provide participants with helpful information and Web links concerning decision-making in various spheres of activity of the company.
- Distribution decision authorities between game participants within one team when participants play different roles such as a senior manager in the top of hierarchy, middle managers and operational managers who monitor the day-to-day activities of the corporation.

Game Management runs the game on remote server (see, Figure 1). Corporations represented by a group of participants are joining the game session from individual terminals via Internet. Each corporation installs and runs a program called **ILMG**, which communicates in a variety of ways with the Game Management program called as **Game Control Centre**. Corporations can use so called Satellites that allow distribute authorities.

Figure 1: Structure of the game



The **Game Control Centre** is the game management system and, unless this program is running, the rest of the game will not function. Using the Game Control Centre, it is possible to change the parameters of the game from the default values, either at the start or during it.

The **ILMG** program runs by the corporations taking part in the game. Typically, about 3-5 people are needed per team to play the game so that decisions have to be taken independently rather than through a consensus compromise based on participants own goals and interests. Corporations compete each with other in different regions of the world with specific characteristics. Each person can have a satellite screen and makes certain decisions, or each company can be based at a terminal. At the commencement, all corporations begin with the same opening position (all recourses are the same for all companies). The main role of the companies is to take decisions, regarding: the structure of the company (where to produce goods), markets and prices, production levels, transportation requirements to service markets, investment in

Table 1: Corporation decisions

	Purchasing	Production	Marketing	Distribution	Finance
Strategic		Location choice Investment into production capacity Process improvement	Market entry Product development Product improvement	Allocation market -> factory Investment into warehouses	Loan Liquidation of assets
Operational	Purchasing orders	Production batch release	Price Advertising	Shipping orders	Repayment of loan Stock exchange transactions

machinery and warehousing, the buying and selling of securities, liquidity actions (e.g. selling assets and issuing new stock), the ordering of market surveys, the distribution of decision authority around the satellites. Different kinds of reports are generated during the game and corporations can value their performance.

Companies take action to enter sales market – this has a cost and time lag attached to it. Plants are located in regions and, to sell in a market, you must have a plant in the region. Plants may not be factories or warehouses, and could represent a sales department. Data can be downloaded into Excel spreadsheets. This data should show demand amplification, and provide some idea of the financial implications of the phenomenon. It may also give an insight into the dynamic behaviour within a supply chain network. It is recommended that the program to be run at least for twelve quarters that corresponds to 3 years of a real life.

The Game Management sets an environment for the Game by deciding on a number of basic issues such as the number of regions available and their names and several other characteristics concerning available products, etc. It also decides on the initial characteristics of an individual corporation joining the Game. The corporations take decisions of different kinds concerning production, marketing, transportation, and several other issues.

**LEARNING ARRANGEMENTS AND SIMULATION SCENARIOS**

The ILMG can be considered as virtual environment that is controlled by Game Management in a multitude of ways. One of the main tasks of the Game Management or course leader is to create a simulation scenario and learning arrangement according to particular course needs and participants’ background knowledge. It means, decision environment and reports could be made as complex as in real life. The principle of scenarios application is shown in Figure 2.

Usually scenario describes behaviour of the system, process of parameters changing and conditions of system functioning (Figure 2). It depicts how the system components interact with each other. The synthesized scenario allows reflecting adequately the process of behaviour of the system, to develop strategy of the organization and realization of measures of fluctuation of a situation, to generate strategic plans of action, to lead the qualitative analysis of consequences of actions, and also to predict prospective loss, possible damage and undertaken risk (Kononov, et. 1999).

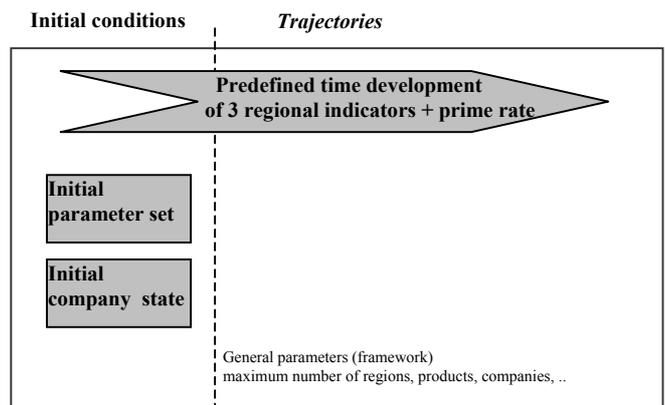


Figure 2. Scenarios in ILMG

Game Management has different means to create scenarios and control the game (Table 3).

First, there is *a set of parameters* that has to be set up. The parameters determine several basic properties of the Game and of its different functions such as different rates, costs and coefficients like market-related parameters (price effect parameters, advertising parameters etc.), financial parameters (prime rate, tax rate etc.) and many others.

Second, the *initial scenario* has to be decided. It covers the following properties of the game: number of available regions, products, securities, account types, initial corporation structure. Note, in the beginning of the game all corporations have the same initial

conditions (the same amount of cash on their accounts, the same operational corporate units and etc.).

Table 3: Scenario based simulation in ILMG

	<i>Tasks for a game manager</i>	
	<i>Before the seminar</i>	<i>During the seminar</i>
<i>Learning arrangement</i>	Plan and design seminar	Facilitate learning: announcements, educational comments, etc.
<i>Simulation scenario</i>	Design scenario: select general framework, initial conditions, preset values, edit parameters	Apply scenario
<i>Simulation model</i>		Control simulation: control timing, edit variables in case of erroneous decisions

Third, **preset parameters** have to be set up. There are time development of prime rate, regional productivity levels, regional wage levels, and business cycle indices.

And at last, has to decide about the **timing of quarters**. Game Manger can decide how long one quarter of the game will last and how many of them have to be run in the particular teaching course. If it's necessary Game Manager can define breaks between periods of the game.

Complexity of a modelled business environment basically depends on the actions described above. All parameters can be changed before the start of the game and during the game session as well. After scenario is decided learning arrangements should be prepared. Before the seminar Game Manager has to plan and design it. During the seminar in order to facilitate learning he should issue different announcements and educational comments. Educational comments can be placed to the game for educational purpose and participants can easily access them. Comments usually contain some useful information for participants regarding different theoretical aspects or some Web link to this information.

## GAME RUNNING EXPERIENCES

Below some educational experiences are presented. In November-December 2004 the ILMG game has been running during three weeks at the Linköping University, Campus Norköping (Sweden) within the Industrial Management course for MSc-students in Electrical Engineering, and Transportation and Communication Engineering. Students were divided in 3 separate groups with six corporations in each group. In total about 70 students participated in the game.

3 game sessions for 4 hours each (in total 12 hours) were planned for every group.

Game handouts (see, Appendix) were issued to participants the day before the first session in order to review the initial conditions of the game. Those handouts contained description of the initial state of the corporation presented by its balance sheet and other useful information necessary to start decision-making.

In introduction to ILMG the main aspects of the game were considered, i.e. decisions that participants could make, reports and overviews provided, the game timing and schedule. Overview of the ILMG software was given as well as basic principles of taking decisions were given.

The following sessions were proposed in the schedule:

- 1<sup>st</sup> session with 4 periods (30 min each with break between the second and the third period);
- 2<sup>nd</sup> session with 8 periods (20 min each with some breaks in between);
- 3<sup>rd</sup> session with 4 periods (15 min each).

Let's note that in the beginning of the game run periods are longer to give participants sufficient time to get familiar with the game software and to make strategic decisions which normally take more time then operational level decisions.

During the game manager follows to all participants activities online as well as compare performance of all companies and in case of necessity to give guidelines how problems can be solved. For instance, on the computer screen the current Profit & Loss Account for all companies could be analysed. Since corporations main task (like in real life) is to make profit, game manager can easily assess which company is the most successful in this particular moment.

It could be the situation when all corporations except one are profitable and this one has tremendous loss due to very high unit production cost. In this case the game manager can take a further look to another report or overview what is available in the game, for instance, to Key Indicators including Liquidity Ratio, Gearing, Return on Total Capital, Return on Equity, Inventory Turnover Rate, Capacity Utilisation, etc. For example, the Capacity Utilisation is about 10% and corporation produces enough items to satisfy a demand. In this case an existing capacity of corporate units should be checked. It could be found that corporation has very high production overheads and they have to sell their production equipment (to make decision Sell Assets) or abolish the unit (this is strategic decision)

After the last period (that corresponds to 8 years of the real life) students are given time to prepare a short presentation about corporate performance. At the end of the game debriefing session is given where students

are involved to the discussion of the following questions:

- How do you feel after the game is over? How did you like it?
- What happened with your corporation during the game?
- What did you learn?
- What next? How would you play the game differently using what you just learned?

Game participants took part in this discussion by answering the questions and presenting the results of their corporation operations to others.

## CONCLUSIONS

The paper describes modern Internet-based simulation game that uses scenario-based approach. In particular, the following strong advantages or strength of the ILMG game were defined: the game requires a multi-disciplinary approach, requires members of the corporation to work as a team, it takes a total enterprise perspective rather than focusing on a particular function, the game provides users friendly interface and etc. The following opportunities were found: the functionality of the game is learnt quickly, the game enables efficient teaching of virtual rooms and quick feedback to enable students to understand cause and effect relationships and etc. Advantages of simulation games and the ILMG unique features are given in the paper. Game management and learning scenarios generation problems are discussed in the paper. Example of learning scenario is proposed as well.

Though the ILMG is fully operational and interest for it increasing there is still a lot of work to do. The most important items are: scenario developing, storing and distributing; designing educational comments and to relate ILMG to theoretical studies in forecasting, inventory control, transport planning, investment analysis, accounting, etc. Judging by the commentary and positive ideas that are coming from the ILMG users the game developers considering the possibility to continue its development and add new features.

## REFERENCES

- Berglund, Jan E., Grubbström, Robert W., *Företagsspel - lek och verklighet (Management Games – Fiction and Reality)*, Stockholm: Bonniers/Aldus 1968.
- Fripp, John. *Learning through simulation: A guide to the design and use of simulations in business and education*. New York: McGraw\_Hill, 1993.
- Grubbström, R.W., Bikovska, J., *The International Logistics Management Game Internet Version 3.0, Participants' Manual*, December 2004.
- Kononov D.A., Kulba V.V., Kovalevsky S.S., Kosjachenko S.A. *Development of scenario spaces*

and the analysis of dynamics of behaviour of social and economic system. Preprint, Moscow, 1999. In Russian.

Mitchell, Rex C. *The Complementary Benefits of Cases and Simulations*,  
[www.exchangesjournal.org/print/print\\_1103.html](http://www.exchangesjournal.org/print/print_1103.html)

## AUTHOR BIOGRAPHIES

**ROBERT W. GRUBBSTRÖM**, Ekon. dr, Tekn. lic., Tekn. dr hc, Dr econ. sc. hc, Professor at Linköping Institute of Technology, Department of Production Economics. E-mail: [rwg@ipe.liu.se](mailto:rwg@ipe.liu.se).

**GALINA MERKURYEVA**, Dr.tech.sc.,Dr.sc.ing., Professor at Riga Technical University, Department of Modelling and Simulation. E-mail: [gm@itl.rtu.lv](mailto:gm@itl.rtu.lv).

**JANA BIKOVSKA**, M.sc.ing, Ph.D student, Leading Specialist at the Department of Modelling and Simulation, Riga Technical University. E-mail: [jana@itl.rtu.lv](mailto:jana@itl.rtu.lv)

**JENS WEBER**, Dipl. Ing., Managing Director of Management Simulation Systems AB. E-mail: [jens.weber@ilmg.com](mailto:jens.weber@ilmg.com).

## APPENDIX

### *Initial Conditions of the Game*

You are the managers of the corporation and you have to decide what to produce and where to sell produced products. You are also responsible for the financial department of your corporation.

There are two types of products available for production: **Standard Bike and Sports Bike**. You can decide to produce and sell bikes in three different regions: **Scandinavia, Baltic States and Western Europe**.

*The initial state for each company is the following:*

Company has initial cash amounting to €40,000,000 and Securities Type 1 amounting to €1,000,000 as its initial assets. It has one operational plant in Scandinavia valued at €5,000,000. The Corporation has an initial bank loan amounting to €10,000,000. The Corporation's Ordinary Share Capital is €41,000,000. Market in Scandinavia is established and two products are adopted.

Market investigation shows a price between 1000 and 4000 is realistic. The volume is depending on industries price level, marketing effort, product development and Business Cycle index in the different markets.

ASSETS		LIABILITIES	
<b>Current Assets</b>	<b>41 000 000</b>	<b>Current Liabilities</b>	
Current Bank Account (Debit)	40 000 000	Current Bank Account (Credit)	
Interest Receivable		Accrued Interest	
Short Term Investments	1 000 000	Trade Creditors	
Trade Debtors		Current Tax Due	
Materials-on-Order		Bank Loans	10 000 000
Materials-in-Transit		Provision for Stock Withdrawals	
Raw Materials Stock		<b>Total Liabilities</b>	<b>10 000 000</b>
Work-in-Progress		<b>EQUITY</b>	
Finished Goods		Ordinary Share Capital	41 000 000
Goods-in-Transit		Profit/Loss	0 000 000
<b>Fixed Assets</b>	<b>5 000 000</b>	Opening Balance 1.1.01	-5 000 000
Buildings and Machinery	5 000 000	Retained Fee for Period	
Warehouses		Dividends	
Plant under Construction		<b>Total Equity</b>	<b>36 000 000</b>
<b>TOTAL</b>	<b>46 000 000</b>	<b>TOTAL</b>	<b>46 000 000</b>

Figure 1: Balance Sheet of January 1<sup>st</sup>, 2001

You have available several data related to Financial Parameters, Regional Parameters, Products Technical Parameters, and Transport Parameters as showed in the Tables below.

Table 1 - Initial Conditions period 1

Region	BCI	P	W
Scandinavia	97	100	300
Baltic states	100	100	330
Western Europe	100	90	250

BCI =Business Cycle Indices  
 BCI-f =Business Cycle Indices-forecast  
 P =Regional Productivity Indices  
 W =Wage Levels

Table 2 – Financial Parameters

Prime Rate (%)	2.5
Bank Loan Rate Factor	2
Standard Loan Period (Years)	5
St Loan Repaym Freq (Quart=0, Ann=1, Month=2)	0
Tax Rate (%)	30

Table 3 – Regional parameters (Examples)

Item	Scandinavia	Baltic States	Western Europe
Unit Projecting Cost (€1000)	2500	1800	2000
Construction Time (Days)	40	50	45
Unit Abolishment Cost (€1000)	300	200	300
Abolishment Time (Days)	30	30	30
Nom Capac Expansion per Inv (Monthly Mach Hrs/€1000000)	3000	3200	2800
Capacity Expansion Fixed Cost (€1000)	100	100	100
Capacity Expansion Lead Time (Days)	20	30	25
Warehouse Capac Expansion per Investm (Units/€1000000)	1000	800	1000

Table 4 - Market-Related Parameters

Region	Market Introduction Time (Days)	Market Introduction Fee (€1000)
Scandinavia	15	200
Baltic States	10	150
Western Europe	15	200

Table 5 – Material Move Tariff

From \ To	Scandinavia			Baltic States			Western Europe		
	Time (Days)	Fixed Cost (€)	Variable Cost (%)	Time (Days)	Fixed Cost (€)	Variable Cost (%)	Time (Days)	Fixed Cost (€)	Variable Cost (%)
Scandinavia	2	500	5	4	400	5	6	600	5
Baltic States	4	400	5	2	500	5	4	400	5
Western Europe	6	600	5	6	500	5	4	500	5

Table 6 – Regular Transport Tariff (Express Transport Tariff)

From \ To	Scandinavia			Baltic States			Western Europe		
	Time (Days)	Fixed Cost (€)	Variable Cost (%)	Time (Days)	Fixed Cost (€)	Variable Cost (%)	Time (Days)	Fixed Cost (€)	Variable Cost (%)
Scandinavia	1	300	5	2	500	5	2	500	5
Baltic States	2	500	5	1	300	5	2	500	5
Western Europe	2	500	5	2	500	5	1	230	5

Table 7 – Technical Coefficients of Products

Item	Scandinavia		Baltic States		Western Europe	
	Standard Bike	Sports Bike	Standard Bike	Sports Bike	Standard Bike	Sports Bike
Material per Unit (Tons/Unit)	0.01	0.02	0.01	0.02	0.01	0.02
Machine Hours per Unit	5	10	5	10	5	10
Work Hours per Unit	10	20	10	20	10	20
Setup Cost (€)	50	50	50	50	50	50

**Planned Seminars:** 1<sup>st</sup> - Introduction and Strategic Decisions, 2<sup>nd</sup> - Tactical and Operational Decision, 3<sup>rd</sup> - Operational Decisions and Debriefing Session.

# Creating and Visualising an Intelligent NPC using Game Engines and AI Tools

N.P. Davies, Q.H. Mehdi and N. Gough

Research Institute for Advanced Technologies  
School of Computing and Information Technology  
University of Wolverhampton  
Wolverhampton, UK  
E-mail: N.P.Davies2@wlv.ac.uk

## ABSTRACT

Our research is focussed on the creation of human-like Artificial Intelligence (AI) based on the Belief Desire Intention (BDI) paradigm that employs rational decision making processes to achieve behaviour in computer game characters that will appear more realistic than current reactive techniques. To test the agent behaviour we need to develop an architecture that incorporates various systems such as an advanced graphics engine, AI environments, and GameBots.

This paper presents a framework that would help solving problems such as predictable and repetitive behaviour. The architecture and usefulness of this system will also be detailed here.

## INTRODUCTION

Computer games developers have traditionally devoted the majority of time and processing power to high performance graphics, as this is the most appealing feature to a potential game player (Namee. 2004). However, this development strategy is producing diminishing returns as modern computer games are now capable of near photo-realistic graphics (Valve Software. 2004). In comparison, AI has been neglected as a development priority, and has evolved to work within the confines of limited processor allocation. This has resulted in AI based on efficient reactive techniques rather than complex reasoning systems. Agents do not 'think' (reason) about actions, but rather perform a set response to a set event. In some cases this has resulted in poor AI, where characters can either appear to be 'stupid' by not being able to react to unpredicted human interaction, or 'cheat' when the developer supplies extra information and abilities to the agents unavailable to human players. Some techniques have been developed that can successfully approximate human-like intelligence very efficiently and effectively (Woodcock 2000), but even these techniques can produce repetitive behaviour as agents do not have the ability to adapt. With graphics now maturing, developers are starting to release more processing power for the inclusion of sophisticated AI that will overcome some of these problems.

The architecture we intend to implement will be based on a simplified view of human cognition known as Belief,

Desire, Intention (BDI) as postulated by Bratman (1987). The architecture is capable of modelling expert human behaviour, and has been used in applications such as Air Traffic Control (Rao & Geofgeff. 1995), and Military Simulations (Wallis et al. 2002), where agents are endowed with military tactics and used as opponents in air combat training. More recently, the architecture is gaining interest from the computer game industry with some success (Norling. 2004). By adopting a goal-based approach to reasoning based on human-like decision-making processes, agents will assess and react to high-level goals in a natural way. This will introduce variability to agents' behaviours, as they will be able to react to an environmental change in many different ways, which do not have to be explicitly specified by designers. There are still many unexplored research areas, including the role of memory and emotion on decision-making processes, the effect of physical conditions of an agent e.g. fatigue, and the aspect of team coordination with social hierarchies where tasks need to be distributed.

## GAME ENGINE DEVELOPMENT

The integration of high performance graphics and human like AI for computer games and simulation would be very useful for visualising agent behaviour. A system for the creation and rendering of virtual scenes and graphical environments was developed by Davies et al (2004). The initial system was used for the creation of quasi-accurate 3D scenes that incorporated animated characters. The implementation used a custom graphics engine developed in DirectX, graphics developed in 3D Studio Max, and a tool for building maps via a 2D plan in C#. It was possible to create some impressive results in a relatively short space of time and investment. Our implementation loads and orientates 3D models and renders them to screen. Animated characters are also incorporated running pre-created animation sequences. Users navigate around the scene using mouse and keyboard input to view the action from different locations and angles. However, it becomes increasingly difficult to make significant progress when requirements become more sophisticated, and our graphics engine started to exhibit limitations quite rapidly. The refresh rates for visualisations start to fall below 30fps when more than three characters are present resulting in jerky animations. The engine does not support dynamic

animation or many other important features. It is possible to incorporate these features, however, significant allocations of time and resources would be required. Even the basic engine we developed is still quite complex. According to Lewis and Jacobson (2002), the development of highly accurate graphics and virtual reality systems is now mainly the domain of generously funded research establishments and military installations with six figure budgets. Therefore, we have investigated alternatives to building our own custom graphics engine.

With the development of modern computer game engine technology in the commercial sector, it is now possible to create impressive results without having to expend resources developing custom graphics engines. Game companies with decades of development experience, and access to some of the most innovative developers and substantial investments in time and money are creating some of the most sophisticated simulation environments available. Prior to game engines, content and code were integrated into a single system. To modify a game, changes had to be made directly to the source code and the game recompiled, which made it virtually impossible for anyone but the game developers to use the game technology for anything other than its original purpose. The original DOOM™ (ID Software) game broke this mould and created the first modular game engine. This technique splits the technical aspects of the graphics rendering pipeline, which is the domain of a small group of specialist developers, from the game specific aspects which can be left to teams of content writers and game developers. This results in the 'game' being considered the graphic models, animation sets, sound, AI and Physics, and enables specialists to concentrate on their particular area of expertise. To allow this process, developers invest heavily in the creation of tools and scripting languages that content developers use to create the game. A recent trend is for developers to release their development tools to the gaming community, allowing the game to be modified after it has been released. This can extend the shelf life of games, with a huge amount of new content being provided on a regular basis by external sources. These modifications range from the creation of new graphics to complete games types (LudoCraft, 2005). It is possible for these tools to be used by the research community which allows access to sophisticated game engine features inexpensively. The use of game engines is not a complete solution to our problems, but it is adequate for the research we are undertaking. While extremely sophisticated, game engines are not real world simulators. We have to work within the limits imposed by the game engine developers. For example, if we wanted a feature such as Kinematics animation that the engine does not provide it would be difficult, but not impossible, for us to incorporate it. However, even though there are limitations, the benefits and functionality far out way negatives.

We intend to use the game engine Unreal Tournament (Epic Games), which provides a high performance real time 3D graphics platform. The game is extendable via a suit of tools provided by the developer, including a C like language Unreal Script which can be used to customise physics and behaviours, and a graphical map builder that is used design game levels. The game consists of 'maps' which are created by level designers and specify the location of graphics, objects, and a network of nodes or waypoints. The internal game characters play as team mates or opponents to human players and are known as 'robots', which are abbreviated to 'bots'. The bots use the map waypoints embedded in the game maps to navigate around the game. The original game contains multiple game types including team based and individual scenarios, and allows human-human, agent-agent, and human-agent teams as shown in Figure 1. The main team based game is called 'Capture the Flag', where teams on two competing sides work together to capture the oppositions flag and return it to their base whilst also protecting their own flag. Other game types include DeathMatch, where the goal is to kill as many opponents as possible.

The game is fundamentally violent in nature, however, it is possible to customise the game in such a way to minimise the violent aspects via the scripting language replacement graphics. There has been substantial interest in the academic community in the use of Unreal Tournament in research including Sioutis et al (2003) which has arisen partly because of the extension GameBots/JavaBots (Marshall et al., 2004) technology shown in Figure 1, which is an extension to the Unreal Engine written in Unreal Script that sits between the game engine and an external client that receives and distributes messages via a TCP/IP link. This has the benefit that AI can be written in any language or development environment that allows a network connection.

Using this information sent out by the game, we need to implement a suitable AI system. It is relatively straightforward to implement algorithms such as the path-planning algorithm A\* in the game using waypoint positions within the game map, and to script more complex behaviours in set pieces. The current game AI is quite standard, for example, in Death Match games, bots work on a finite state machine reasoning with states such as roam, flee, and engage enemy. Different skill levels are achieved by increasing the bots speed, accuracy and field of view. The AI is developed in this way for a number of reasons. Games are processor intensive. Clock cycles are at a premium when other factors such as physics and graphics are also competing for processor time. As far as the game developer is concerned, as long as the behaviour appears variable and reasoned, it is irrelevant how it is achieved. Performance and optimisation are the dominant factor. Another issue is the fact that for many computer games human level sophistication is simply not required. The

game domains are relatively discreet requiring only a limited array of behaviours. We intend to create more sophisticated behaviour in more complex game types, and to produce a more robust cognitive architecture. This will require us to model human like behaviour by capturing expert knowledge for a particular game type, and convert this information into a knowledge base that an AI agent can utilise. However, the information received will determine the AI that can be produced. Similar work was carried out by Norling (2002) using the Quake engine. A number of limitations were reported by them where certain behaviours cannot be modelled due to the deficiencies in information sent from the game server. Specifically this related to the behaviour of throwing a grenade at an archway so it would bounce off and hit a chasing bot. As the game engine did not send information identifying when an agent was under an archway, this could not be modelled. With the use of Unreal, which is more customisable, we may be able to address these deficiencies. In effect we have an environment that is partially observable where the agent cannot see everything that is happening everywhere in the environment, and dynamic, the environment is constantly changing. There are many factors happening simultaneously, which cannot be anticipated. Actions are non-deterministic, and it may be that a goal will not be achieved even though an agent is committed to achieving it. For example, an agent may decide to increase its health by searching for a health pack. It cannot guarantee that this action will be successful, and other events may occur that intervene. On route to collecting the health pack, the agent may sustain further damage and die. The AI engine needs to account for this using information sent from the game server, and adapt its behaviour accordingly. The agent will have to build an internal representation of the environment, plan a course of actions which will be limited to available commands, execute the plan, and observe the results. This scenario is getting close to a real-life open system used in robotics and its associated technologies, and would support research into many areas including reactive systems, deliberative systems, cognition, memory and learning, planning etc.

The Unreal Tournament and GameBot system specifies messages into two types; server messages and client messages. Messages from the server provide sensory information detailing what a bot can 'see' in the game, as well as information about the bots physical status e.g. health, armour etc. Other information is also sent, such as if the agent has taken damage, and the direction a shot came from. Messages from the client take the form of commands that instruct the agent to perform tasks such as rotate, walk, and shoot. The GameBots system allows the client to have access to the internal native AI system, and allows instructions such as GETPATH which returns a list of waypoints between two locations, and RUNTO that will direct the bot to a specified waypoint. Another benefit of

the GameBots / JavaBots system is that it allows for an extendable gaming environment.

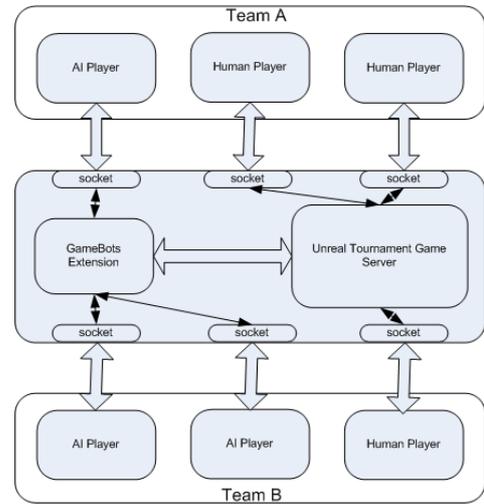


Figure 1: GameBot system

## BDI AI ARCHITECTURE

To develop an AI agent that will have the ability to recognise situations, formulate plans based on the current situation, and execute those plans, while executing plans, the agent should be able to monitor the environment in order to ensure the plan is still relevant. The agent has to be embodied with the ability to adapt and change plans in order to add variability to behaviour, which will make the agent appear less predictable. In addition to plan modelling, a model for temporal characteristics that influence the decision-making process such as emotional, social and physical characteristics will be created. This includes elements of team working, where agents will collaborate with each other by allocating tasks. Finally, the application needs to execute plans in a timely manner on a resource bound machine.

The AI architecture identified as suitable candidate that exhibits some of these features is the Belief-Desire-Intention architecture (BDI) as originally postulated by Bratman (1987), shown in Figure 2. In this architecture, an agent is characterised by its beliefs about the state of its environment, goals (desires) that it wishes to achieve in this environment, and a set of plans and partial plans it can use in order to satisfy its desires. As the model for the AI agent needs to be executed on a machine with limited resources, the concept of intention is introduced which limits the deliberation an agent is required to perform. This information will be limited to information regarding tasks the agent has committed itself to achieving. An intention is formed when the agent commits to a particular goal, and retrieves a plan from the plan library that

contains a particular sequence of steps to perform in order to achieve a goal. The steps themselves may be atomic actions, or they may be sub goals, which can be satisfied by other plans. The beliefs, goals and intentions of an agent are maintained by the BDI reasoning engine. This engine will help to drive the agent, update beliefs, monitor and update goals, form intentions, and select plans to achieve goals.

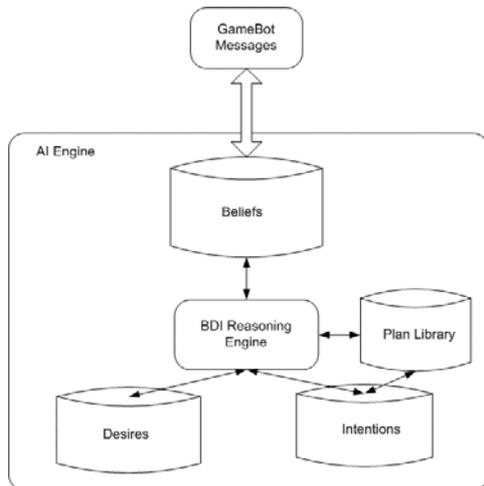


Figure 2: BDI Architecture (Bratman 1987)

The plan library is an important feature of this architecture that specifies how goals can be achieved. Although the plans are fixed sequence of actions, they do not have to be fully specified. For any particular goal, there may be multiple plans to achieve that goal, and while any plan may be fully specified as a sequence of actions, a plan may instead consist of a sequence of sub goals or a combination of actions and sub goals. All plans will eventually decompose into a sequence of atomic events, which will take the form of GameBot messages. If a plan contains sub goals, reasoning to decompose the plan into atomic events can be delayed until the behaviour is required. For example, in the Unreal Game, a plan may be to kill opponent agents. This can be broken into sub-goals of 'locate weapon', 'locate enemy' and finally 'kill enemy'. With this high level plan in place, the agent narrow deliberate to achieve the first stage of the plan, which may involve a path planning or roaming behaviour. Once this part of the plan is completed, the next stage will become relevant, and deliberation commenced upon that stage. While this does not achieve the full range of adaptability that humans display when reasoning about their goals, it does allow considerable flexibility in the agent's planning, and its resulting behaviours, and also allows a balance between reactive and deliberative planning.

In theory, the BDI agents will use a utility-based decision making strategy, where decisions are made to maximize the expected utility of the selected course of action. However, in practice on resource bound machines this is

either impractical or unrealistic. In the some systems the default behaviour is simply to select the first applicable plan, regardless of expected utility. In real-world situations, while either of these strategies may be used on occasion, it is argued that neither is commonly used, particularly when the person is operating within their area of expertise (Norling 2003). The DBI paradigm does not incorporate all the required features we are intending to incorporate into our AI, such as physical attributes including emotional states, memory, and learning. However, there is much research ongoing into incorporating these aspects.

## BDI TOOLS

The development of a BDI architecture is not a trivial task, and much as the development of commercial game engines can be used by researchers, there are many mature systems based on BDI available ranging from open source to commercial applications. Examples of open source projects are the Java Agent Development Environment (JADE) (TiLab 2005) with the BDI extension JADEX (Braubach & Pokahr. 2005). JADE is a mature platform that is FIPA (Foundation for Intelligent Physical Agents) compliant. FIPA are an organisation that is producing standards for software agent development. The JADE environment is based on the Java environment, and allows the creation of agents via a GUI interface. JADEX is a BDI add-on to JADE that is currently in development. The use of JADEX may be a viable option to base our AI on in the future, however, as development is currently at a BETA stage it was considered JADEX is still not mature enough to consider. Another example of an open source BDI project is Jason, which is an interpreter for the agent oriented BDI programming language AgentSpeak. The environment is also constructed over in the Java environment, and allows the development of agents which can be distributed over a network connection. An alternative to pure BDI environments is the cognitive AI architecture SOAR (University of Michigan) which is a mature environment that has been in development for over 20 years. It attempts to provide a general cognitive architecture for developing agents that can exhibit human-like intelligence. It addresses factors such as knowledge complexity and rational decisions making within these complex domains. An advantage in using this technology is that it has started to address factors such as memory, learning and emotion. The platform has been used in a diverse set of academic applications including training simulations and robotic control. The environment is not platform independent like the Java solutions, however, there are environment available to the major operating systems. However, the platform we have chosen to implement our AI on is the leading edge commercial BDI development environment JACK (Agent Oriented Software). This system provides a comprehensive toolkit, and extensive documentation. Like other system outlined above it is designed as an extension

to the Java language, which makes it platform independent. Agents can be developed via a GUI application, or by the use of keyword extensions to Java. Another aspect that makes JACK favourable is the maturity of the system for use in computer games. Work has already been completed by (Norling) into integrating JACK and Quake, and more recently, (Soitis) have developed an interface between JACK and GameBots, which may be released as open source in the future. There is also interest in developing an extension to JACK called Co-Jack, which is attempting to incorporate moderating factors such as emotion and physical conditions such as fatigue, and has been initiated by the Ministry of Defence (Lucas. 2003.)

### THE PROPOSED ARCHITECTURE

Figure 3 shows the proposed architecture which is based on the work carried out by Sioutis et al. 2003. The architecture uses the JACK BDI programming language and links to the Unreal Tournament game engine via the use of GameBots/JavaBot technology. This allows a structured environment in which to investigate the development of human like artificial intelligence. The architecture is divided into two main sections, with a third section linking the two together. The section to the left of the dotted line represents the commercial game engine, Unreal Tournament, where a custom game can be defined via databases containing custom graphics, and a second database containing game modifications and rules written in Unreal Script. To the right of the dotted line is the BDI reasoning system JACK which receives messages from the game server and uses it to update a Belief data set, describing the current world state. Based on the beliefs, a set of goals and plans will be formulated from a plan library which at the lowest level will take the form of GameBots commands such as Rotate and Walk. The middle linking section contains the GameBots technology that distributes messages and receives commands over a TCP/IP network connection. The architecture is not limited to the use of Unreal Tournament. If an alternative game were used, only the middle link layer would need to be rewritten to account for an alternative set of messages. The GameBot/UT link will then be developed and incorporated. This part of this system has been developed by Sioutis et al (2003), which we may adopt in the future.

### APPLICATION EXAMPLE

To begin the development of our artificial intelligence we first need to specify and design a game type where human like reasoning would be of benefit. It has been demonstrated that games such as a FPS Death Match can be developed with simple rules and function satisfactorily. Games that would benefit from human like AI will be of a slower pace, and give the game player time to perceive and interact with the intelligence. Games that will exhibit this

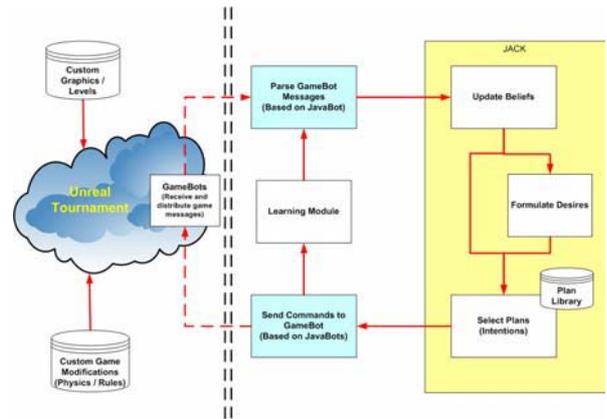


Figure 3 : Unreal Tournament / GameBots / JACK architecture

type of behaviour would include, but not limited to, tactical combat games such as hostage situations or squad based combat missions. Unreal Tournament lends itself to this type of game, and a modification would be feasible to produce. Once the game has been specified, a knowledge base will be developed outlining the tactics that can be employed by the agent. This task will account for a significant amount of development time.



Figure 4: PacMan style game with integrated BDI controlled NPCs

As an initial proof of concept we have developed a simple PacMan game shown in Figure 4, which is linked to JACK through the glue language TCL. PacMan was chosen as it shares many of the features that are common in FPS games such as Unreal Tournament. Agents are required to navigate around an environment based on waypoint nodes, and are blocked for navigating certain paths due to obstructions such as walls. Agents can also pick up key objects such as pills which can change other agents' states. Agents can incorporate team tactics to capture an enemy, and an agent can evade capture from the teams of agents. The game is open and dynamic, and much of the knowledge base will be applicable to further developments using a more sophisticated games engine.

## CONCLUSION AND FURTHER WORK

In this paper, the benefits and limitations of developing human like artificial intelligence and high performance graphics engines have been outlined. The benefit of using commercial game engines including Unreal Tournament and its associated tools is to allow the designer to concentrate on AI modelling and simulation rather than devoting time for developing a custom graphics engine that may not be as sophisticated as a commercial engine. The limitations of using commercial game engines have also been identified; however, these limitations have not had a significant effect on the development and implementation of the system. The BDI architecture as proves to be a suitable system paradigm for the requirements of our AI system.

In conclusion a system that includes the use of JACK, GameBot/JavaBot and Unreal Tournament offers a good development platform for testing AI agents in virtual worlds.

Further work will include the development of GameBot / JACK interface that will allow sophisticated techniques such as emotion and physical limitations to be incorporated into the system.

## REFERENCES

- Agent Oriented Software Pty. Ltd. JACK Intelligent Agents. <http://www.agent-software.com>
- Bratman, M. (1987) "Intentions, plans and practical reasoning." Harvard University Press, Cambridge, Massachusetts.
- Braubach, L., Pokahr, A., 'JADEx' <http://vis-www.informatik.uni-hamburg.de/projects/jadex/> Accessed 10.03.2005
- Davies, N., Mehdi, Q., Gough, N. E. (2004), 'Crime Scene Reconstruction With Integrated Animated Characters', Proceeding of European Simulation Multiconference. Networked Simulation and Simulated Networks, Magdeburg, Germany, 388-393
- Epic Games Inc. Unreal Tournament. <http://www.unreal.com>
- Lewis, M., Jacobson, J., (2002), 'Game Engines in Scientific Research', Communications of the ACM, Volume 45, Issue 1. New York, USA
- Lucas A., Human Variability in Computer Generated Forces, presented at Behavior Representation in Modeling and Simulation (BRIMS'03), 2003.
- LudoCraft Ltd, AirBuccaneers. <http://ludocraft oulu.fi/airbuccaneers/> accessed 10.03.2005
- Magerko, B., Laird, J., Assanie, M., Kerfoot, A., Stokes, D., (2004) 'AI Characters and Directors for Interactive Computer Games' AAAI Press; Proceedings of the 2004 Innovative Applications of Artificial Intelligence Conference, San Jose, CA
- Marshall, A. N., Gamard, S., Kaminka, G. J., Manojlovich, Tejada S., Gamebots, viewed 10th Mar 2005, <http://planetunreal.com/gamebots/>
- McBreen, H., Shafe, P. Jack, M., Wyard, P. (2000) "Experimental assessment of the effectiveness of synthetic personae for multi-modal E-Retail applications" in: Proc. 4th Int. Conf. on Autonomous Agents (2000), pp. 39-45, Barcelona, Spain
- Namee. M. B., (2004), Proactive Persistent Agents: "Using situational intelligence to create support characters in character-centric computer games". PhD These, University of Dublin, Trinity College
- Norling, E. (2004) "Folk psychology for human modelling: extending the BDI paradigm". In : Int. Conf. on Autonomous Agents and Multi Agent Systems (AAMAS), New York, 2004.
- Rao, A., Georgeff, M., (1995) "BDI agents: from theory to practice" Tech. Rep. 56, Australian Artificial Intelligence Institute, Melbourne, Australia
- Sioutis, C, Ichalkaranje, N & Jain, LC 2003, 'A framework for interfacing BDI Agents to a real-time simulated environment', Design and Application of Hybrid Intelligent Systems, A Abraham, M Koppen & K Franke (eds), proceedings of the 3rd International Conference of Hybrid Intelligent Systems (HIS'03), Melbourne, Australia, December 2003, IOS-Press, Amsterdam, The Netherlands, pp. 743-748.
- TiLab, Java Agent Development Environment (JADE) <http://jade.tilab.com/> accessed 10/3/2005
- University of Michigan 'SOAR' <http://sitemaker.umich.edu/soar> Accessed 10.03.05
- Urlings, P., Tweedale, J., Sioutis, C., Ichalkaranje, N., (2003) "Intelligent agents as cognitive team members" in: Proc. 10th Int. Conf. On Human Computer Interaction, Crete-Greece, June 2003
- Valve Software (2004) Half Life 2, <http://www.half-life.com/>
- Wallis, P., Ronnquist, R., Jarvis, D., Lucas, A., (2002) "The automated wingman – using JACK intelligent agents for unmanned autonomous vehicles", in IEEE Aerospace Conference, Big Sky MT
- Woodcock, S., (2000) "Flocking: a simple technique for simulating group behaviour" in Game Programming Gems, Charles River Media

# DYNAMIC HYBRID STRATEGY MODELS FOR NETWORKED MULTIPLAYER GAMES

Aaron McCoy, Seamus McLoone, Tomás Ward  
Department of Electronic Engineering  
National University of Ireland, Maynooth  
Maynooth, Co. Kildare, Ireland  
E-mail: amccoy@eeng.nuim.ie

Declan Delaney  
Department of Computer Science  
National University of Ireland, Maynooth  
Maynooth, Co. Kildare, Ireland

## KEYWORDS

Distributed Interactive Applications, Hybrid Strategy Model, Networked Multiplayer Computer Games, Dead Reckoning, Packet-Reduction, User-Modelling.

## ABSTRACT

Two of the primary factors in the development of networked multiplayer computer games are network latency and network bandwidth. Reducing the effects of network latency helps maintain game-state fidelity, while reducing network bandwidth usage increases the scalability of the game to support more players. The current technique to address these issues is to have each player locally simulate remote objects (e.g. other players). This is known as dead reckoning. Provided the local simulations are accurate to within a given tolerance, dead reckoning reduces the amount of information required to be transmitted between players. This paper presents an extension to the recently proposed Hybrid Strategy Model (HSM) technique, known as the Dynamic Hybrid Strategy Model (DHSM). By dynamically switching between models of user behaviour, the DHSM attempts to improve the prediction capability of the local simulations, allowing them to stay within a given tolerance for a longer amount of time. This can lead to further reductions in the amount of information required to be transmitted. Presented results for the case of a simple first-person shooter (FPS) game demonstrate the validity of the DHSM approach over dead reckoning, leading to a reduction in the number of state update packets sent and indicating significant potential for network traffic reduction in various multiplayer games/simulations.

## INTRODUCTION

Networked multiplayer computer games are one of the most important areas of an already burgeoning computer games industry. In a formal domain such applications belong to a class more commonly known as Distributed Interactive Applications (DIAs). These systems typically involve many users simultaneously interacting in a simulated virtual environment. Game designers are constantly seeking to scale such applications to more and more simultaneous users whilst still maintaining a high quality of interactivity and responsiveness. However, a number of technical problems combine to make delivery of such an

experience difficult (Singhal and Zyda 1999; McCoy et al. 2003). One such problem is latency, which is the time it takes for information to propagate across the network to all participants. Another closely related issue is the problem of network bandwidth. Within a DIA we refer to these problems as the information updating issue. Several methods have been devised to reduce the quantity of data that needs to be transmitted between participants. The standard for Distributed Interactive Simulation (DIS) defines one such method known as dead reckoning (IEEE 1995), a form of client predictive contract mechanism.

Recently, an alternative technique known as the Hybrid Strategy Model (HSM) has been introduced, which offers an improvement over the performance of dead reckoning (Delaney et al. 2003; Delaney 2004). It is a hybrid predictive contract technique, which dynamically switches between a short-term dead reckoning model and a longer-term user behavioural model, allowing quasi-deterministic modelling of an entity's dynamics. This can reduce the number of communicated update packets required for remote modelling of entities when compared to the use of a pure dead reckoning model alone.

This paper demonstrates how to apply the HSM technique when behavioural models must be recalculated during run-time as users pursue a dynamic goal. The original exposition of the HSM method was only ever demonstrated with fixed navigational goals. In section two of this paper we describe predictive contract mechanisms as used for information updating in DIAs. Existing solutions to this issue, including dead reckoning, are outlined. In particular the HSM technique is summarized. Section three describes our extension of this proposed hybrid switching technique, referred to as a Dynamic Hybrid Strategy Model (DHSM), and its implementation is discussed in section four. Several test environments were developed to compare this new technique with existing dead reckoning techniques. These environments are described in section five. Example simulation results are presented in section six for both the DHSM and dead reckoning techniques. Finally, the paper ends with the conclusions and suggestions for future research.

## PREDICTIVE CONTRACT MECHANISMS

The most common solution to the information updating issue within DIAs involves a client-side predictive contract mechanism called dead reckoning (IEEE 1995). All participating clients agree to maintain the same low-order local models of the dynamics of all other participating entities. This is the contract. Each participant also maintains a model of its own entity dynamics, which it continuously compares to its actual dynamics. When these differ by a pre-defined error-threshold amount, update information is broadcast to all other participants, who then proceed to update their models for that entity based on this newly received state information. Convergence algorithms are typically incorporated to allow a natural transition to occur between the modelled and actual motion when update data arrives.

Alternative methods have also been explored, and these include relevance/area-of-interest filtering and multicast groups (Rak and van Hook 1996), packet bundling (Liang et al. 1999), data compression (Van Hook et al. 1994), time management, priority scheduling and visibility culling (Faisstnauer et al. 2000). A comprehensive overview of such consistency maintenance measures can be found in (Delaney 2004). In addition, various enhancements have also been proposed for use with the standard dead reckoning algorithm, including adaptive error-thresholds (Lee et al. 1999; Shim and Kim 2001), multi-step dynamic curve fitting (Singhal and Cheriton 1994) and pre-determination of likely error-threshold deviations (Zhang et al. 2004).

### The Hybrid Strategy Model

The Hybrid Strategy Model (HSM) as proposed in (Delaney et al. 2003) is a predictive model of the following form:

$$M = px + (1 - p)\Gamma$$

where  $x$  is any conventional dead reckoning model,  $\Gamma$  is a long-term model of entity behaviour and  $p$  is a binary weighting factor governed by:

$$p = 1, \text{ for } \|E - \Gamma\| \geq \theta \\ = 0, \text{ otherwise}$$

where  $\theta$  represents a distance measure threshold between the actual behaviour  $E$  and the long-term model.

The model given by  $M$  is used by participating clients in a DIA. The parameters and initial entity state used by the model are updated every time the state deviates from the true state by a predefined threshold amount  $T_m$ .

It has been shown that this approach leads to a reduction in the number of packets that need to be transmitted in a DIA for a given consistency. However up to now this approach has only ever been applied to DIAs where the long-term models are fixed navigational behaviours in which an entity is traversing an environment. To apply the HSM when a user's behaviour is reacting to a changing goal (originally defined as a dynamic goal in Delaney's original paper) is non-trivial. In such cases, suitable models of user behaviour are required which are clearly complex functions of, among other things, the environment, user experience and intention, and user ability. Nonetheless, for a restricted application such as an FPS many of these dependencies can be modelled adequately for HSM purposes in certain situations. In the following two sections, our approach to applying the HSM technique to dynamic goals in a realistic FPS is demonstrated.

## THE DYNAMIC HYBRID STRATEGY MODEL

A Dynamic Hybrid Strategy Model (DHSM) consists of a set of  $i$  candidate prediction models given by:

$$M_{candidate} = \{m_1, \dots, m_i\}$$

In addition, a default model is chosen to provide initial prediction for the DHSM (typically a short-term dead reckoning model would be chosen as default, though this is not a necessity):

$$m_{initial} = m_{default}, \quad \{m_{default} \in M_{candidate}\}$$

During each simulation step, the DHSM has a currently selected *active* model (beginning with the initial model) that is used to perform actual prediction and entity-state-update (ESU) packet generation for a user. However, the DHSM also runs prediction for each candidate model in parallel with the active model and records each associated prediction error for every simulation step:

For  $m_{current} \in M_{candidate}$

- Predict  $P(m_{current})$
- Compute Error  $E(m_{current})$

Endfor

When the current prediction error for the *active* model exceeds the allowable error-threshold, an ESU packet is required to be generated and transmitted. At this point, the average prediction error (APE) since the *last* transmitted ESU packet is computed for each candidate prediction model. This assigns a score to each model based on overall performance between consecutive ESU packet transmissions. The model that exhibits the best score (i.e. lowest APE) is then chosen to be the current *best* model.

If  $E(m_{active}) \geq ErrorThreshold$

For  $m_{current} \in M_{candidate}$

```

- Compute avg. error  $APE(m_{current})$ 
Endfor
 $m_{best} = \text{Argmin}(APE(m_{current}), \forall m_{current})$ 
If  $m_{active} \neq m_{best}$ 
-  $m_{active} = m_{best}$ 
Endif
Endif

```

The currently selected active model is switched over to the determined best model if necessary (i.e. if we are not already using the best model), and an ESU packet is generated and transmitted. Appended onto the end of the packet is an identifier number indicating which candidate model is now in use. When remote clients receive this ESU packet, they can use this ID to switch their own local models of prediction in accordance with the new active model. This process is repeated every time the allowable error-threshold is exceeded, resulting in dynamic switching between prediction models based on their current performance in relation to one another.

## DHSM IMPLEMENTATION

The DHSM implementation presented in this paper utilizes a candidate set of just two prediction models: *dead reckoning* and *shortest-path*.

### Dead-Reckoning (DR)

This is the standard first-order, one-step short-term dead-reckoning prediction algorithm as originally detailed in the Standard for Distributed Interactive Simulations (DIS) (IEEE 1995). It is given by the following equation of motion:

$$P = P_0 + V_0 \Delta t$$

where  $P_0$  and  $V_0$  refer to initial position and velocity respectively and  $\Delta t$  is the time increment for the current prediction step.

### Shortest-Path (SP)

This is a simple dynamic first-order, one-step behavioural model that assumes a user will move along the shortest-path from the current position to the target position (i.e. their dynamic goal) with constant speed. It is given by the following equation of motion:

$$P = P_0 + (T - P_0) \left( \frac{\|V_0\| \Delta t}{\|T - P_0\|} \right)$$

where  $P_0$  and  $V_0$  refer to initial position and velocity respectively,  $\Delta t$  is the time increment for the current prediction step and  $T$  refers to the current target (dynamic goal) position.

The design of this shortest-path model was driven by previous work that we have done in analysing the behaviour of users within various DIAs (and in particular networked multiplayer computer games) (McCoy et al. 2004a; McCoy et al. 2004b). It is grounded in the notion that within these networked games, users often exhibit tendencies to move directly closer to their target of interest when attempting to engage, thereby maximising their probability of disabling an opponent. During these time periods, correlation can often be observed between both sets of data, indicating a tight coupling of behaviour for both players. It is this coupling of data that we are exploiting with the shortest-path model in order to define a user's behaviour in terms of their dynamic goal. This allows us to reduce the transmission of state data required to accurately predict the user using a remote model.

For the implementation of the DHSM, we have also included an ancillary *line-of-sight* score function used to weight the selection of the shortest-path model over the dead reckoning model. It states that unless there is a *direct line-of-sight between a user and their dynamic goal*, the DHSM will not switch from using the DR model to the SP model, even if the SP model would have provided a better average prediction error since the last received ESU packet. The motivation behind this is the assumption that if a user cannot see the dynamic goal, the probability of rapidly changing behaviour occurring is far greater, thus decreasing the chance that the SP model will provide any benefit for short-term prediction over that provided by the standard DR model.

## DATA COLLECTION

We utilize the Torque Game Engine for performing experiments and data collection (Marshall et al. 2004). During a user's interaction within a test environment, data is sampled at a rate of 20Hz and consists of position, velocity, and forward facing direction vectors for each player (human or AI controlled). Discrete events are recorded for the cases of a player being disabled, firing their weapon or (in the case of a human user) interacting directly with the control device (i.e. mouse and/or keyboard). For each game tick, a trace can be performed to determine if there is an unobstructed line-of-sight (LOS) between any two players. This trace information is combined with the forward facing direction vectors in a post-processing step to determine *onscreen* LOS status for each sampling time.

Figure 1 shows plan (top-down) views of the three environments used for the experiments. In each case, a single test subject was asked to play against a single *non-reactive* computer-controlled opponent (BOT). The goal here was to disable the BOT a desired number of times as fast as possible. Each environment consisted of a unique *path network* that was used by the BOT for constant circumnavigation. The motivation behind this particular set of test scenarios was to ensure that the user's dynamic goal (in this case the BOT) was known

in advance, thus negating any possible requirement for performing some kind of target identification procedure (as would be the case if there were multiple independent dynamic goals for a user to choose from).



Figure 1: Test environment plan views showing path networks

## RESULTS AND ANALYSIS

Presented below in Tables 1-3 are simulation results for packet number, average prediction error and standard deviation error respectively for a collection of three datasets using a variety of increasing error-thresholds (to put the error-thresholds into perspective, the height of a player within our test environments is approximately 2.3 units). Each dataset was recorded using a different test subject and test environment (the particular environment is noted directly above the results in the tables). The term ‘DR’ refers to the standard first-order, one-step dead reckoning prediction model and is used as the base comparison with which to compare results. The term ‘DHSM’ refers to the Dynamic Hybrid Strategy Model using both the standard first-order, one-step dead reckoning prediction model and the first-order, one-step shortest-path prediction model (as outlined previously). The term ‘% Red’ refers to the percentage reduction (or increase) in the associated variable (number of packets sent, average prediction error or standard deviation error). A negative value indicates better performance with respect to the Dynamic Hybrid Strategy Model in direct comparison with the standard dead reckoning model. A heartbeat timeout of 5 seconds was set for both the DR and DHSM models (meaning if the error-threshold has not been exceeded for 5 seconds or more, an ESU packet is automatically generated). This reflects a typical timeout value that might be used within a DIA system (Singhal et al. 1999). Finally, ideal network conditions were assumed (i.e. no network latency or packet loss).

From inspection of the results, it is noted that in almost every case, the DHSM offers a reduction in the number of packets sent that ranges from small bandwidth savings (in the region of 1% or lower packet reduction) to very large bandwidth savings (in the region of 20% and sometimes higher packet reduction). It is important to note here that a quantitative comparison of inter-dataset results is not productive, as each test conducted is independent of any other. Despite this fact however, comparatively speaking it would appear that the percentage savings are partly dependent on the particular test subject and test environment, and

Table 1: Packet number results for several datasets

Threshold	Number of Update Packets Sent (PacketNum)								
	Dataset 1 (Environment 1)			Dataset 2 (Environment 2)			Dataset 3 (Environment 3)		
	DR	DHSM	% Red	DR	DHSM	% Red	DR	DHSM	% Red
0.5	1071	1087	1.49	618	624	0.97	478	494	3.35
1	724	720	-0.55	419	412	-1.67	340	325	-4.41
1.5	565	549	-2.83	336	323	-3.87	268	241	-10.07
2	485	467	-3.71	274	264	-3.65	237	210	-11.39
2.5	442	420	-4.98	256	244	-4.69	208	181	-12.98
3	391	375	-4.09	228	216	-5.26	184	154	-16.30
3.5	364	349	-4.12	214	203	-5.14	172	143	-16.86
4	327	315	-3.67	185	178	-3.78	165	133	-19.39
4.5	311	298	-4.18	173	167	-3.47	160	126	-21.25
5	284	272	-4.23	170	161	-5.29	149	124	-16.78
5.5	265	256	-3.40	156	150	-3.85	143	110	-23.08
6	249	244	-2.01	150	143	-4.67	136	107	-21.32
6.5	239	232	-2.93	146	139	-4.79	131	105	-19.85
7	233	223	-4.29	138	130	-5.80	127	101	-20.47
7.5	220	214	-2.73	129	120	-6.98	120	97	-19.17
8	211	211	0	124	115	-7.26	114	94	-17.54
8.5	207	200	-3.38	121	115	-4.96	109	93	-14.68
9	201	197	-1.99	123	113	-8.13	108	92	-14.81
9.5	194	191	-1.55	115	110	-4.35	108	92	-14.81
10	202	192	-4.95	111	104	-6.31	107	90	-15.89

Table 2: Average prediction error results for several datasets

Threshold	Average Prediction Error per Simulation Step (E)								
	Dataset 1 (Environment 1)			Dataset 2 (Environment 2)			Dataset 3 (Environment 3)		
	DR	DHSM	% Red	DR	DHSM	% Red	DR	DHSM	% Red
0.5	0.2535	0.2568	1.30	0.2751	0.2786	1.27	0.254	0.2657	4.61
1	0.4111	0.4256	3.53	0.4509	0.4595	1.91	0.3938	0.4151	5.41
1.5	0.5755	0.5947	3.34	0.6148	0.6288	2.28	0.54	0.5751	6.50
2	0.7449	0.7586	1.84	0.7809	0.7867	0.74	0.6856	0.6817	-0.57
2.5	0.9104	0.9283	1.97	0.9842	0.9921	0.80	0.8251	0.8895	7.81
3	1.0529	1.0597	0.65	1.1595	1.1723	1.10	0.9804	1.0412	6.20
3.5	1.2112	1.2192	0.66	1.3419	1.3792	2.78	1.1159	1.1702	4.87
4	1.364	1.3671	0.23	1.4808	1.5128	2.16	1.3394	1.3686	2.18
4.5	1.5533	1.5738	1.32	1.6675	1.6836	0.97	1.5256	1.4827	-2.81
5	1.7816	1.7489	-1.84	1.8471	1.7998	-2.56	1.7058	1.4778	-13.37
5.5	1.9759	1.9094	-3.37	2.045	2.0221	-1.12	1.811	1.6177	-10.67
6	2.119	2.1186	-0.02	2.1262	2.1933	3.16	1.9017	1.7369	-8.67
6.5	2.3202	2.2692	-2.20	2.3813	2.3934	0.51	2.0685	1.9731	-4.61
7	2.4628	2.4195	-1.76	2.6539	2.6244	-1.11	2.3996	2.0513	-14.51
7.5	2.74	2.5452	-7.11	2.7276	2.7212	-0.23	2.468	2.2653	-8.21
8	2.8036	2.676	-4.55	2.9641	2.979	0.50	2.6943	2.2987	-14.68
8.5	2.9346	2.8045	-4.43	3.07	3.0749	0.16	2.7637	2.3358	-15.48
9	3.1314	3.0776	-1.72	3.3125	3.2983	-0.43	2.8355	2.3022	-18.81
9.5	3.2789	3.1337	-4.43	3.4683	3.5024	0.98	2.9132	2.4025	-17.53
10	3.497	3.3846	-3.21	3.6011	3.6299	0.80	3.0654	2.4462	-20.20

Table 3: Standard deviation error results for several datasets

Threshold	Standard Deviation Error per Simulation Step (D)								
	Dataset 1 (Environment 1)			Dataset 2 (Environment 2)			Dataset 3 (Environment 3)		
	DR	DHSM	% Red	DR	DHSM	% Red	DR	DHSM	% Red
0.5	0.1892	0.1887	-0.26	0.1861	0.1849	-0.64	0.1829	0.1822	-0.38
1	0.3402	0.3392	-0.29	0.3431	0.3373	-1.69	0.3289	0.3211	-2.37
1.5	0.4929	0.4946	0.34	0.5004	0.4913	-1.82	0.4809	0.459	-4.55
2	0.6534	0.6462	-1.10	0.6469	0.6352	-1.81	0.6383	0.6017	-5.73
2.5	0.8106	0.7975	-1.62	0.8129	0.7998	-1.61	0.7939	0.7734	-2.58
3	0.9625	0.9387	-2.47	0.9556	0.9369	-1.96	0.9371	0.8948	-4.51
3.5	1.1214	1.1073	-1.26	1.1245	1.1062	-1.63	1.0779	1.0151	-5.83
4	1.2799	1.2559	-1.88	1.2485	1.2537	0.42	1.2713	1.1864	-6.68
4.5	1.4389	1.4297	-0.64	1.4292	1.4322	0.21	1.4345	1.3156	-8.29
5	1.5846	1.5434	-2.60	1.5834	1.5586	-1.57	1.5892	1.4029	-11.72
5.5	1.7745	1.7205	-3.04	1.7257	1.6928	-1.91	1.7644	1.5378	-12.84
6	1.894	1.88	-0.74	1.8699	1.8588	-0.59	1.9044	1.6508	-13.32
6.5	2.0589	1.985	-3.59	2.0575	2.0248	-1.59	2.0685	1.8111	-12.44
7	2.1962	2.1392	-2.60	2.2371	2.2154	-0.97	2.1523	1.9146	-11.04
7.5	2.3595	2.2902	-2.94	2.3027	2.282	-0.90	2.3568	2.0622	-12.50
8	2.5071	2.4264	-3.22	2.4868	2.4522	-1.39	2.4912	2.1919	-12.01
8.5	2.6202	2.5558	-2.46	2.6211	2.5912	-1.14	2.611	2.3198	-11.15
9	2.7764	2.7084	-2.45	2.7888	2.7221	-2.39	2.7297	2.3558	-13.70
9.5	2.8996	2.8537	-1.58	2.9252	2.8624	-2.15	2.8617	2.4948	-12.82
10	3.0724	3.0582	-0.46	3.0397	3.016	-0.78	3.0601	2.6697	-12.76

definitely dependent in this case on the particular dynamic prediction model being used (i.e. shortest-path). A more advanced and accurate prediction model would almost certainly yield even better results, particularly if it were tailored for a specific user’s behavioural traits and habits. It is also worth noting that the reduction in number of packets sent does not appear to be linearly dependent on the error-threshold (i.e.

increasing the error-threshold does not guarantee a higher reduction in number of packets sent over the corresponding standard dead reckoning prediction model).

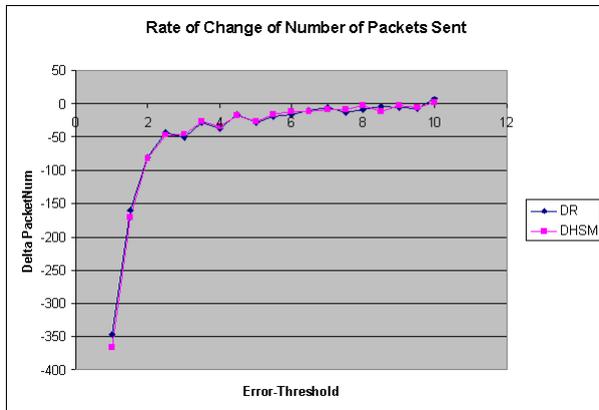


Figure 2:  $\Delta$ PacketNum vs error-threshold (Dataset 1)

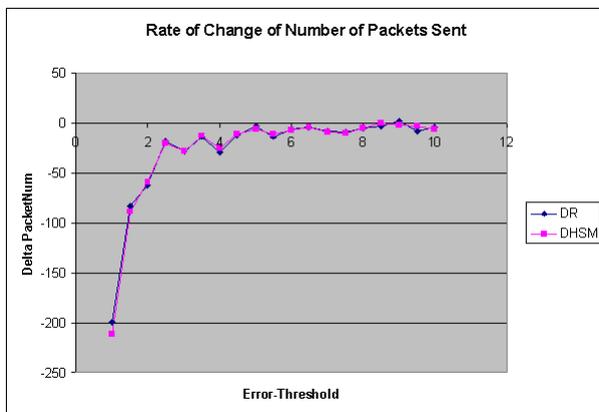


Figure 3:  $\Delta$ PacketNum vs error-threshold (Dataset 2)

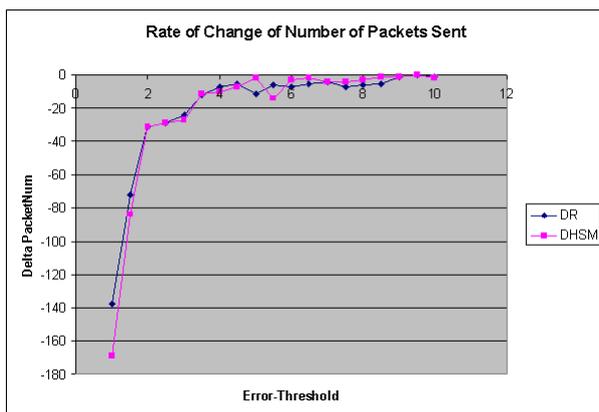


Figure 4:  $\Delta$ PacketNum vs error-threshold (Dataset 3)

Further inspection of the results highlights the fact that the reduction in packet numbers (and hence bandwidth savings) for the DHSM often comes at the cost of slightly reduced prediction accuracy for tight error-thresholds ( $\leq 4$ ). Conversely, at higher error-thresholds ( $\geq 5$ ) the DHSM often provides improved prediction accuracy. In addition, the DHSM also offers a smaller standard deviation for the prediction error over its DR

counterpart in practically every case, implying a slightly more stable predictive capability. This is further evidenced in Figures 2-4 that present the rate of change of generated packets as a function of the increasing error-threshold for each dataset. Inspection of these plots reveals no erroneous packet generation behaviour for the DHSM when compared with the DR mechanism. On the contrary, they provide a close match for all three datasets.

## CONCLUSIONS

In this paper, we have described a novel extension to the concept of the Hybrid Strategy Model (HSM) called the Dynamic Hybrid Strategy Model (DHSM). Like the HSM, the DHSM attempts to reduce the amount of entity-state-update (ESU) packets required to maintain consistency within a Distributed Interactive Application (DIA) such as a networked multiplayer computer game. Unlike the HSM, which uses fixed long-term goals, the DHSM takes account of a user's behaviour towards a dynamic goal and attempts to exploit the shared information contained within this relationship. By switching between various candidate prediction models at appropriate times, the DHSM provides improved prediction for clients participating within the DIA.

We have provided simulation results for several different test subjects and test environments that verify the validity of our approach, showing a reduction in the number of ESU packets sent (bandwidth usage) in favour of the DHSM technique over pure dead reckoning in the majority of our specific test situations. In addition, improved average prediction error for higher error-thresholds and improved standard deviation error make this a promising technique for possible use within a prediction scheme incorporating some kind of adaptive error-threshold selection (Lee et al. 1999; Shim and Kim 2001). Despite the promising results however, a better understanding of the relationship between packet reductions, user experience and test environment topology is required, and to this end additional tests will need to be conducted utilizing more complex, real-world scenarios. These include such things as realistic network conditions, reactive computer-controlled opponents and human vs. human experiments.

Future work will involve the investigation of advanced user-modelling techniques to provide a better pool of candidate prediction models that work under more general and complex situations, where the likes of simple shortest-path type prediction will not provide enough prediction accuracy. Possible avenues of approach here include the use of neural networks (Thurau et al. 2003) and probabilistic independence networks (Smyth et al. 1996) to model the relationships between users and dynamic goals. In addition, extensions to the switching criteria and score functions used by the DHSM will also be investigated with the aim of ensuring optimal model switching.

## ACKNOWLEDGEMENT

This material is based upon works supported by Enterprise Ireland under grant no. SC/2002/129/.

## REFERENCES

- Delaney, D.; Ward, T.; and S. McLoone. 2003. "Reducing Update Packets in Distributed Interactive Applications using a Hybrid Approach." In *Proceedings of the 16<sup>th</sup> International Conference on Parallel and Distributed Computing Systems (PDCS 2003)* (Reno, USA, Aug. 13-15). 417-422.
- Delaney, D. 2004. *Latency Reduction in Distributed Interactive Applications Using Hybrid Strategy-Based Models*. Ph.D. dissertation Department of Electronic Engineering. National University of Ireland, Maynooth. November 2004.
- Faisstnauer, C.; Schmalstieg, D.; and W. Purgathofer. 2000. "Scheduling for Very Large Virtual Environments and Networked Games Using Visibility and Priorities." In *Proceedings of the 4<sup>th</sup> IEEE International Workshop on Distributed Simulation and Real-Time Applications (DS-RT'00)* (San Francisco, California, Aug. 25-27). 31-38.
- IEEE. 1995. *IEEE Standard for Distributed Interactive Simulation – Applications Protocols*. Volume IEEE Standard 1278.1-1995. The Institute for Electrical and Electronics Engineers, Inc., New York, March 26 1996.
- Lee, B.S.; Cai, W.; Turner, S.J.; and L. Chen. 1999. "Adaptive Dead Reckoning Algorithms for Distributed Interactive Simulation." *International Journal of Simulation* 1. No. 1-2.
- Liang, L. A. H.; Cai, W.; Lee, B. S.; and S.J. Turner. 1999. "Performance Analysis of Packet Bundling Techniques in DIS." In *Proceedings of the 3<sup>rd</sup> IEEE International Workshop on Distributed Interactive Simulation and Real-Time Applications* (College Park, Maryland, Oct. 23-24). 75-82.
- Marshall, D.; McCoy, A.; Delaney, D.; McLoone, S.; and T. Ward. 2004. "A Realistic Distributed Interactive Application Testbed for Static and Dynamic Entity State Data Acquisition." In *Proceedings of the Irish Signals and Systems Conference 2004* (Belfast, N.I., Jun. 30 – Jul. 02). 83-88.
- McCoy, A.; Delaney, D.; and T. Ward. 2003. "Game-State Fidelity across Distributed Interactive Games." *ACM Crossroads*, 9.4, 4-9.
- McCoy, A.; Delaney, D.; McLoone, S.; and T. Ward. 2004a. "Investigating Behavioural State Data-Partitioning for User-Modelling in Distributed Interactive Applications." In *Proceedings of the 8<sup>th</sup> IEEE International Symposium on Distributed Simulation and Real Time Applications* (Budapest, Hungary, Oct. 21-23). 74-82.
- McCoy, A.; Delaney, D.; McLoone, S.; and T. Ward. 2004b. "Towards Statistical Client Prediction – Analysis of User Behaviour in Distributed Interactive Media." In *Proceedings of the 5<sup>th</sup> Game-On International Conference (CGAIDE 2004)* (MS Campus, Reading, UK, Nov, 8-10). 144-149.
- Rak, S. J. and D. J. van Hook. 1996. "Evaluation of Grid-based Relevance Filtering for Multicast Group Assignment." In *Proceedings of the 14<sup>th</sup> DIS Workshop on Standards for the Interoperability of Distributed Simulation* (March 1996). 739 – 747.
- Shim, K. H. and J. S. Kim. 2001. "A Dead Reckoning Algorithm with Variable Threshold Scheme in Networked Virtual Environment." In *Proceedings of the 2001 IEEE International Conference on Systems, Man, and Cybernetics* Vol. 2 (Oct. 7-10). 1113-1118.
- Singhal, S. and D. Cheriton. 1994. "Using a Position History-Based Protocol for Distributed Object Visualisation." Technical Report STAN-CS-TR-94-1505. Department of Computer Science, Stanford University.
- Singhal, S. and M. Zyda. 1999. *Networked Virtual Environments: Design and Implementation*. ACM Press SIGGRAPH Series. Addison-Wesley, Reading, Massachusetts.
- Smyth, P.; Heckerman, D.; and M.I. Jordan. 1996. "Probabilistic Independence Networks for Hidden Markov Probability Models." *Neural Computation* Vol.9, No.2. 227-269.
- Thurau, C.; Bauckhage, C.; and G. Sagerer. 2003. "Combining Self Organizing Maps and Multilayer Perceptrons to Learn Bot-Behaviour for a Commercial Game." In *Proceedings of the 4<sup>th</sup> International Conference on Intelligent Games and Simulation (Game-On 2003)* (London, UK, Nov. 19-21). 119-124.
- Van Hook, D. J.; Calvin, J. O.; and D. C. Miller. 1994. "A protocol independent compression algorithm (PICA)." *Advanced Distributed Simulation Memorandum 20PM-ADS-005*. MIT Lincoln Laboratories, Lexington, MA.
- Zhang, X.; Gracanin, D.; and T. P. Duncan. 2004. "Evaluation of a Pre-Reckoning Algorithm for Distributed Virtual Environments." In *Proceedings of the 10<sup>th</sup> International Conference on Parallel and Distributed Systems (ICPADS'04)* (Newport Beach, California, Jul. 7-9).

## AUTHOR BIOGRAPHIES

**AARON MCCOY** received his B.Sc. Degree from the National University of Ireland, Maynooth in 2002, specializing in computer science and theoretical physics. His main areas of interest are distributed systems, networked virtual environments and the use of artificial intelligence in interactive games. He is currently studying for his Ph.D. at the Department of Electronic Engineering, NUI Maynooth, in the area of user modelling in Distributed Interactive Applications.

**DECLAN DELANEY** received his electronic engineering degree from University College Dublin in 1991 and his Masters degree in 1999. At present, he lectures in the Department of Computer Science at NUI Maynooth and is director of the Masters of Science in Software Engineering program. His current research interests focus on techniques for reducing the effects of latency in distributed networked applications. He is a chartered member of the IET.

**SEAMUS MCLOONE** received his B.Eng. honours degree in Electrical and Electronic Engineering from Queen's University Belfast in 1996 and his Ph.D. in September 2000. At present, he is employed as a lecturer in the Department of Electronic Engineering at NUI Maynooth. His main research interests include multiple model approaches to nonlinear system identification and reducing/masking the effects of latency in distributed interactive applications.

**TOMÁS WARD** received the B.E., the M.Eng.Sc. and Ph.D. degrees from University College Dublin, Ireland in 1994, 1996 and 1999 respectively. In 2000, he became a lecturer in the Department of Electronic Engineering at NUI Maynooth. His main research interests lie in the areas of near-infrared optical and terahertz medical imaging, and distributed environments for multi-user interaction.

# A Preliminary Investigation into Eye Gaze Data in a First Person Shooter Game

Alan Kenny, Hendrik Koesling, Declan Delaney  
Department of Computer Science  
National University of Ireland, Maynooth  
Maynooth, Co. Kildare, Ireland  
E-mail: akenny@cs.nuim.ie

Seamus McLoone and Tomás Ward  
Department of Electronic Engineering  
National University of Ireland, Maynooth  
Maynooth, Co. Kildare, Ireland

## KEYWORDS

Eye Tracking, Networked Games, Psycho-Perceptual Measures

## ABSTRACT

This paper describes a study carried out in which the eye gaze data of several users playing a simple First Person Shooter (FPS) game has been recorded. This work shows the design and implementation of a simple game and how the execution of the game can be synchronized with an eye tracking system. The motivation behind this work is to determine the existence of visual psycho-perceptual phenomena, which may be of some use in developing appropriate information limits for distributed interactive media compression algorithms. Only 2 degrees of the 140 degrees of human vision has a high level of detail. It may be possible to determine the areas of the screen that a user is focusing on and render it in high detail or pay particular attention to its contents so as to set appropriate dead reckoning limits. Our experiment shows that eye tracking may allow for improvements in rendering and new compression algorithms to be created for an online FPS game.

## 1. INTRODUCTION

Networked interactive computer games are an important class of Distributed Interactive Application (DIA). These games generate a large quantity of data that must be communicated across the physical network between participating game nodes to allow the game state to be replicated at remote nodes. However, network bandwidth is a limited resource, being particularly scarce at typical end-user locations. A key research area in distributed computer games is, therefore, concerned with devising new techniques and methodologies that reduce the amount of data that needs to be generated. In this way the available bandwidth is exploited fully, without loss of consistency for the participants. One popular technique introduced with the Distributed Interactive Simulation (DIS) standard (IEEE 1993) and used in games such as Doom and Tribes is dead reckoning (Pantel and Wolf 2002). This compresses user dynamics information by down-sampling the user trajectory. Other techniques employed in DIAs include the hybrid strategy model (Delaney et al. 2003), area of

interest management, data compression and dynamic load balancing (Singhal and Zyda 1999).

More recently in distributed media such as video or music, psycho perceptual phenomena have been employed to reduce the quantity of data that needs to be communicated without impacting the quality of the end-user experience. Examples include the mp3 standard for music and the MPEG or DivX standards for video. However, in DIAs such as online computer games the exploitation of psycho-perceptual parameters has received limited attention. This paper is part of a larger body of work that hypothesises that knowledge of player psycho-perception can be exploited to reduce the amount of data that needs to be transmitted between users in distributed games. Visual information processing is one of the most important components of how a user perceives a gaming experience. The contribution of this paper is that it describes how eye movement data can be recorded and synchronized with game play in an action game developed using the Torque games engine.

Eye tracking is one area of research that has yet to be explored in this context, to the best of our knowledge, and has the potential to yield interesting and useful psycho-perceptual results. Eye movements in static scenes, such as a computer screen, are not performed continuously, but consist of "jumps" (saccades) and "rests" (fixations). Eye-tracking results are represented as lists of fixation data. This data contains, for example, the fixation position and duration, the current pupil size and the start and end times of each fixation. Previous work on decision making in soccer match (Koesling and Höner 2003) has shown distinct pre- and post-decisional phases of user action: once a target player has been chosen, the visual field narrows around this target for post-decisional action planning. The underlying action theory for decision-making is known as the Rubicon Theory and the point that separates the pre- and post-decisional phases is known as the Rubicon point (Gollwitzer and Bayer 1999). It is our goal to determine if similar results could be obtained with respect to computer games. This would enable developers to determine entities and regions of the game environment on which players focus their attention. Such information could be used to determine scene rendering priorities, level of rendering detail and load balancing,

as well as providing a parameter for assigning priorities to entity extrapolation in compression techniques such as dead reckoning or the hybrid strategy model.

The remainder of this paper is structured as follows: section 2 describes the experiment and how the data acquisition was achieved. The data is analysed and discussed in detail in section 3. Finally conclusions are made along with some proposals for future work in section 4.

## 2. METHODOLOGY

The system consisted of an SR Research EyeLink®2 eye tracker, a control Personal Computer (PC) and a display PC. The EyeLink®2 is a binocular video-based eye-tracking system with a sampling rate of up to 500Hz. During experiments, subjects wear the eye-tracker headset while they view stimuli on a computer monitor screen (Figure 1). Small infrared (IR) cameras on the headset transmit information about the subjects' head and pupil positions to the eye tracker. From this data the eye tracker calculates and records the positions subjects look at on the screen. The control PC hosts the eye tracker operational software and is used to control and calibrate the eye tracker. The experiment is run on the display PC, which is connected to the control PC via an Ethernet connection. The display PC for these experiments was a laptop with an external monitor, mouse and keyboard. Four IR markers were attached to the external monitor in order for the eye tracker to compensate for head movements. The advantage of using an external monitor, keyboard and mouse is that it provides the operator with real-time control over the experiment, so that the eye tracker software can be started and stopped without interfering with the subjects controls.



Figures 1: Experiment Set-up

A First Person Shooter game was created using the Torque Game Engine (<http://www.garagegames.com>). Torque is an industrial game engine that has been used to produce many games (Marshall et al. 2004) including the award winning Tribes 2. The objective of the game was to get as high a score as possible. A user could

increase their score by destroying enemy bots or by collecting tokens, but in order to achieve a high score more difficult bots had to be attacked. Each game lasted two minutes, with a brief period for synchronisation at the start and end of the game. Six volunteers participated, each playing the game twice; the first run was considered a learning experience. This resulted in a total of twelve data sets. Through the use of questionnaires it was discovered that each volunteer had a good level of computer expertise. Three volunteers had keen interest in video games and were quite proficient at them; the remainder had relatively little experience.

One of the greatest challenges encountered in performing the experiment was the synchronisation between the eye tracker software and the FPS game. These software packages do not have easily accessible interfaces and the communication and processing of data is dictated by the real-time constraints of the experiment. Once the eye tracker program is initialised and the calibration procedures have finished, the external Torque application is executed. A "STARTING GAME" message is sent to the eye tracker file so that the recorded eye data can be synchronized with the application time. In addition, just prior to the game play commencing and finishing, a set of animated points are displayed on screen, each lasting approximately 900ms. The user is asked to fixate on these points. By locating this pattern in the eye tracker data, the data relating to the start and end of game play can be isolated.

Three sets of data were recorded for each trial: the eye tracker data, the Torque recording and a custom Torque log file. Eye tracker data was recorded every 4ms. The Torque recording is a playback of game events based on network samples and each recording was later recorded as an AVI file using FRAPS. FRAPS is a utility specially designed for recording game footage (further information on FRAPS can be found at <http://www.fraps.com/>). The Torque log records entity position every 500ms in addition to all events associated with items, weapons, damage and kills.

## 3. ANALYSIS

In Figure 2 a sample image is shown, which superimposes the computer game footage with markers representing eye fixations. The darker cross shows the left eye fixation point and the dark triangle shows the right eye fixation point. In order to generate such superimposed video, the eye data relating to the game play session had to be extracted by hand using the synchronization mechanism mentioned in section 2. Perl and Matlab® scripts were then developed to strip out and recombine the information from the two separate data.



Figure 2: Eye Fixation Data

Observations of initial results from this experiment are summarized in the following paragraphs.

The crosshair in a FPS game effectively creates a natural fixation point. This coupled with the fact that the user effectively controls the worldview with the crosshair, results in the majority of fixations taking place near the centre of the screen, where ‘near centre’ is taken to be the inner 400x300 rectangle from the 800x600 resolution screen. In essence the control of the natural fixation point around the point of most activity in the game results in the user using this as a virtual eye. Nearly 88% of all fixations fall within this region as shown in Figure 3. This results in the users spending 86% of fixation time and 82% of the game time within the near centre region. Interestingly the regions of the screen representing the user’s health, message box and score received very little eye fixations, only 2% of all fixations fall within these regions of which one user contributed 53%.

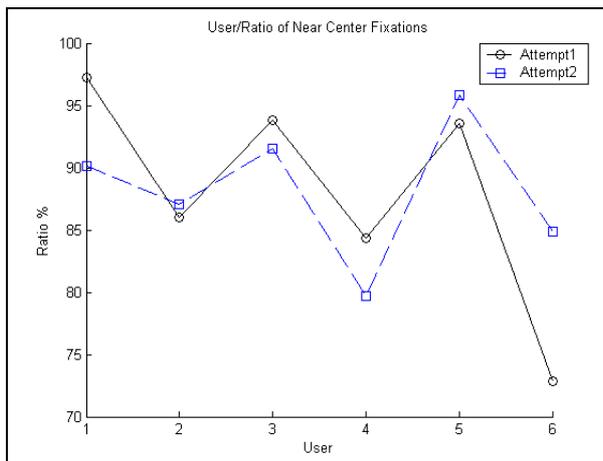


Figure 3: Ratio of Near Centre Fixations to Total Fixations for the Twelve Data Sets

This phenomenon could be exploited by graphics rendering algorithms (O’ Sullivan and Dingliana 2001).

Areas within the near centre region could be rendered in high quality with the periphery being rendered in a lower quality, as the human peripheral vision is less aware of the lower quality graphics. It has been shown in previous research that the human eye only focuses on a small area of a screen in high detail. This result, based on human perception capabilities, can be used to further compress real time MPEG video streams, which have appropriately-paced content and require high resolution gain in the eye gaze window (Komogortsev and Khan 2004). Research in level of detail (LOD) management has also shown that a user’s subjective opinion of a scene is not hampered by lowering the detail of non-important objects in that scene (Brown et al. 2003). This would reduce processing requirements in graphically intense FPS titles, such as Doom 3 and Half-life 2, allowing developers to create graphically detailed titles that reach a wider audience.

Additionally, the human eye is more likely to notice inconsistencies (i.e. due to latency, jitter and packet loss) that occur around the points of fixations. As most fixations take place within this near centre region for a FPS game it is likely that any inconsistencies that occur within this region are more likely to be noticed than those that occur outside this region. In networked online games this could be exploited by ‘area-of-interest’ algorithms to reduce the update packets that need to be exchanged between participants.

Previous work on a set of soccer videos has shown that there exists up to 600ms of delay from when a person makes a decision about executing a task as indicated by eye movement and when the task is actually executed (Koesling et al. 2001). In those experiments users had only to make a single decision while viewing a series of six-second video clips. However, in the experiments described here the nature of a FPS game means that the time between user decisions can be very short. This makes it difficult to consistently identify the Rubicon point. As a consequence, it may be difficult to utilize eye data in order to pre send information during online multiplayer games. However, other game genres that do not require twitch actions may be more suitable to using eye data for predicting users’ actions. Further experiments to investigate the decision-making process in games would have to be conducted to test this hypothesis.

Our results also indicate that, in general, users exhibit fixation durations that are comparable to those of other media (Sibert and Jacob 2000). In Figure 4 the average left eye fixation duration for each session is shown (a similar graph can be derived from the right eye fixations). As can be seen the average fixation duration for most users falls within the range of 300ms and 750ms, with only two outliers. These outliers belong to an extremely experienced FPS player, which may explain their occurrence. The results appear to contradict the information displayed in the

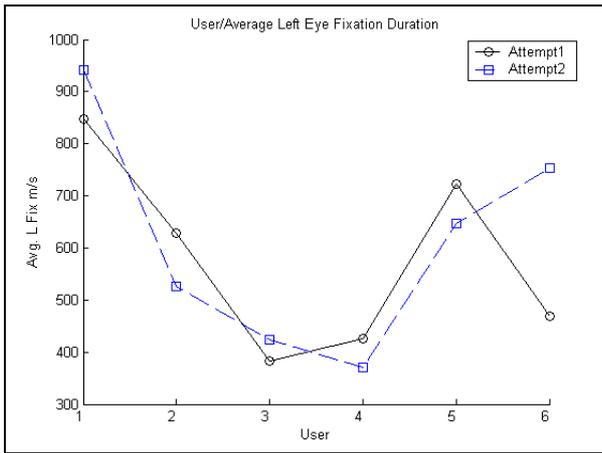


Figure 4: Average Left Eye Fixation Duration

superimposed videos, where users seem to have relatively long fixation durations (fixations of up to ten seconds being measured). This can be explained by the fact that most users (except the previously noted outliers) have more fixations under 300ms than between 300ms and 600ms or over 600ms as seen in figure 5. The average fixation duration minus the fixations under 300ms is nearly 1150ms, which more accurately represents the data that was observed in the videos.

Another point of interest is the tendency for higher game scores to result in longer fixation durations and consequently a lower number of fixations as shown in figure 6. It can also be seen that players didn't necessarily perform better on their second attempt. While the scoring mechanism used within the game may not perfectly represent task proficiency all subjects exhibited higher scores for games with longer average fixation durations. More accurate measures of task proficiency would have to be developed in order to verify this. There may also be a correlation between the high percentages of near centre fixations, the duration of

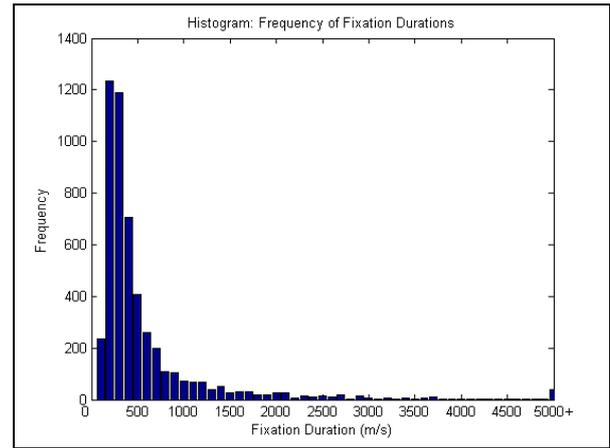


Figure 5: Frequency Distribution of Fixation Durations

the average fixation and the user's score. Using a player's score as an indication of their abilities, the three best players had an average of 91% near centre fixations and average fixation duration of just over 700ms. Additionally the worst players had an average near centre fixation of 84% and average fixation duration of just less than 470ms. Eye movement research has already suggested that high frequency of fixations in a region coupled with long durations indicates high task difficulty and can be taken as an indication of information complexity, similarly a smaller number of saccades indicates higher mental load (Pan et al. 2004). In relation to our FPS game this may be explained as the more proficient players having greater control over the virtual eye. As the task of scoring involved disabling enemies and this is accomplished by focusing the crosshair on the target it may indicate that the better a player becomes at supplementing their own eye movement with virtual eye movement the more accurately they are able to focus on the activity around the crosshairs. Further research would have to be carried out to determine if this is the case.

Table 1: Statistics Summary

User	Score	Near Centre Fix. %	Time in Near Centre Region %	Avg. L. Fix. ms	Num. Fix. Under 300 ms	Num. Fix. Between 300 ms – 600 ms	Num. Fix. Over 600 ms
1.1	3350	97	99	847	62	100	125
2.1	3700	85	95	628	194	93	84
3.1	1200	93	96	383	368	131	83
4.1	2100	84	88	426	302	155	91
5.1	2500	93	97	723	148	76	103
6.1	1100	72	86	468	249	141	78
1.2	5800	90	98	941	97	60	97
2.2	2800	87	92	526	220	140	87
3.2	1500	91	96	424	321	144	79
4.2	1200	79	85	370	368	167	81
5.2	2150	95	96	647	157	113	91
6.2	1900	85	90	754	165	58	88

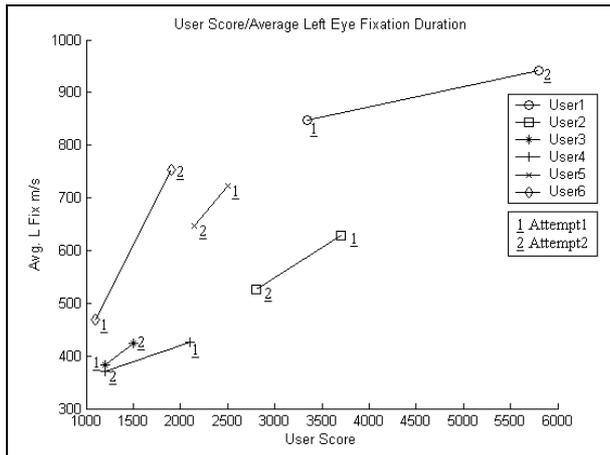


Figure 6: Average Left Eye Fixation Versus Score

#### 4. CONCLUSIONS AND FUTURE WORK

In this paper a system for synchronising an eye tracker with a commercial game development engine has been described. Preliminary results suggest that the system successfully co-registers eye gaze information with the visual scene presented to the user during their game play. Analysis of the data indicates that users of a FPS spend an extremely large proportion of time focused around the centre of the screen, that there is some indication of the Rubicon theory taking place within games and that there may be a correlation between task proficiency and the duration of eye movement.

Future work will investigate psycho-perceptual phenomenon that can be used to filter the information that needs to be transmitted in DIAs such as networked computer games to maintain an adequately consistent global shared state. Specifically if the point of the Rubicon can be accurately and continuously determined then some information about user actions could be sent before they take place. While our results have shown that it may be difficult to achieve this in a fast action FPS game, other game genres with slower interactions or an FPS with more controlled interactions may prove easier to analyse. Future experiments will aim to more accurately highlight the Rubicon effect within games.

#### REFERENCES

Brown, R., Copper, L. and Pham, B. 2003. "Visual Attention-based Polygon Level of Detail Management." *First International Conference on Computer Graphics and Interactive Techniques, Melbourne, Australia*.

Delaney, D., Ward, T. and McLoone, S. 2003. "Reducing Update Packets in Distributed Interactive Applications using a Hybrid Model." *16th International Conference on Parallel and Distributed Computing Systems, Reno, USA*.

Gollwitzer, P.M., and Bayer, U. 1999. "Deliberative versus implemental mindsets in the control of action." *Dual-process theories in social psychology (pp. 403-422)*. New York: Guilford.

IEEE 1993. IEEE Standard for Distributed Interactive Simulation - Application Protocols, in IEEE Std 1278-1993, I.C. Society, Editor. 1993, IEEE: New York.

Koesling, H. and Honer, O. 2003. "Bend it like Beckham – Mindsets and Visual Attention in Decision-Making in Soccer." *Proc. 12<sup>th</sup> European Conference on Eye Movements*, Dundee, UK.

Komogortsev, O., and Khan, J. 2004. "Predictive Perceptual Compression for Real Time Video Communication." *Proc. 12<sup>th</sup> ACM International Conference on Media, New York, USA*.

Marshall, D., Delaney, D., McLoone, S. and Ward, T. 2004. "Exploring the Spatial Density of Strategy Models in a Realistic Distributed Interactive Application." *International Symposium on Distributed Simulation and Real-Time Applications*, Budapest, Hungary.

O'Sullivan, C. and Dingliana, J. 2001. "Collisions and Perception." *ACM Transactions on Graphics, Volume 20, Issue 3*.

Pan, B. et al. 2004. "The Determinants of Web Page Viewing Behaviour: An Eye Tracking Study." *Proc. Eye Tracking Research and Applications Symposium, San Antonio, Texas*.

Pantel, L. and Wolf, L. 2002. "On the Suitability of Dead Reckoning Schemes for Games." *First Workshop on Network and System Support for Games*, Braunschweig, Germany.

Sibert, L. and Jacob, R. 2000. "Evaluation of Eye Gaze Interaction." *Proc. SIGCHI conference on Human Factors, The Hague, Netherlands*.

Singhal, S. and Zyda, M. 1999. "Networked Virtual Environments: Design and Implementation." *ACM Press SIGGRAPH Series*. Addison-Wesley, Reading, Massachusetts.

#### AUTHOR BIOGRAPHIES

**ALAN KENNY** was born in Mullingar, Ireland and went to NUI, Maynooth, where he studied computer science and software engineering and obtained his degree in 2004. Having spent a short stint working as a web developer he returned to the University to study a PhD in the area of psycho-perceptual measures in networked games. His e-mail address is: akenny@cs.nuim.com and his Web page can be found at <http://www.cs.may.ie/~akenny>.

**HENDRIK KOESLING** graduated in computer science and physics (M.Sc.) in 1997 and received a PhD in computer science from Bielefeld University, Germany, in 2003. He worked as a lecturer and researcher in the Neuroinformatics Group and the CRC360 "Situated Artificial Communicators" at Bielefeld University until 2004. He currently holds a postdoctoral research position at the Computer Science Department at NUI Maynooth. His research interests are in visual perception/attention and cognitive processes in humans, based on the analysis and modeling of eye movements.

**DECLAN DELANEY** received his electronic engineering degree from University College Dublin in

1991 and his Masters degree in 1999. He has worked abroad as both a software engineer and project manager. At present he lectures in the Computer Science Department in NUI Maynooth. His current research interests focus on techniques for reducing latency in distributed networked applications and on the use of small group PBL in software engineering education. He is a chartered member of the IEE.

**SEAMUS McLOONE** received his B.Eng. honours degree in Electrical and Electronic Engineering from Queen's University Belfast in 1996 and his Ph.D., entitled Nonlinear Identification using Local Model Networks, in September 2000. At present, he is employed as a lecturer in the Department of Electronic Engineering in the National University of Ireland (NUI), Maynooth. His main research interests include multiple model approaches to nonlinear system identification and reducing and masking the effects of latency in distributed interactive applications (DIAs).

**TOMAS WARD** received the B.E., the M.Eng.Sc. and Ph.D degrees from University College, Dublin, Ireland in 1994, 1996 and 1999 respectively. He worked as Research Coordinator in the Rehabilitation Engineering Department at the National Rehabilitation Hospital from 1996-1999. From 1999-2000 he was an Assistant Lecturer in the Department of Computer Science at the National University of Ireland, Maynooth. In 2000 he became a lecturer in the Department of Electronic Engineering at NUI Maynooth. His main research interests lie in the areas of near-infrared optical and terahertz medical imaging, the development of therapeutic tools in medical rehabilitation, emotion synthesis in artificial speech and distributed environments for multi-user interaction.

**The  
2005 High Performance  
Computing & Simulation  
(HPC&S)  
Conference 2005**



# FAST PATTERN DETECTION USING PARALLEL NEURAL PROCESSORS AND IMAGE DECOMPOSITION

HAZEM M. EL-BAKRY

*University of Aizu, Aizu Wakamatsu, Japan 965-8580  
hazemelbakry1@yahoo.com*

**ABSTRACT-** In this paper, an approach to reduce the computation steps required by fast neural networks for the searching process is presented. The principle of divide and conquer strategy is applied through image decomposition. Each image is divided into small in size sub-images and then each one is tested separately using a fast neural network. The operation of fast neural networks based on applying cross correlation in the frequency domain between the input image and the weights of the hidden neurons. Compared to conventional and fast neural networks, experimental results show that a speed up ratio is achieved when applying this technique to locate human faces automatically in cluttered scenes. Furthermore, faster face detection is obtained by using parallel processing techniques to test the resulting sub-images at the same time using the same number of fast neural networks. In contrast to using only fast neural networks, the speed up ratio is increased with the size of the input image when using fast neural networks and image decomposition.

**KEYWORDS:** Fast Neural Networks, 2D-FFT, Cross Correlation, Image decomposition, Parallel Processing.

## I. INTRODUCTION

The human face is a complex pattern. Finding human faces automatically in a scene is a difficult yet significant problem. It is the first step in fully automating human face recognition system. Face detection is the fundamental step before the face recognition or identification procedure. Its reliability and time response have a major influence on the performance and usability of the whole face recognition system. For web indexation applications, the processing time must be kept as low as possible as the number of images on the web increases continuously [8]. Among other techniques [6], neural networks are efficient face detectors [2,5].

The main objective of this paper is to reduce the detection time using neural networks. Compared to conventional neural networks, fast neural networks based on cross correlation between the input image and the weights of neural networks in frequency domain have shown a significant reduction in the number of computation steps required to detect an object (face/iris) in the image under test [1,2]. In section II, fast neural

networks for face detection are described. A faster searching algorithm for face detection that reduces the number of the required computation steps through image decomposition is presented in section III. Accelerating the new approach using parallel processing techniques is introduced in section IV.

## II. FAST NEURAL NETWORKS FOR HUMAN FACE DETECTION

In this section, a fast algorithm for object/face detection based on two dimensional cross correlations that take place between the tested image and the sliding window (20x20 pixels) was described. Such window is represented by the neural network weights situated between the input unit and the hidden layer. The convolution theorem in mathematical analysis says that a convolution of  $f$  with  $h$  is identical to the result of the following steps: let  $F$  and  $H$  be the results of the Fourier transformation of  $f$  and  $h$  in the frequency domain. Multiply  $F$  and  $H$  in the frequency domain point by point and then transform this product into spatial domain via the inverse Fourier transform. As a result, these cross correlations can be represented by a product in the frequency domain. Thus, by using cross correlation in the frequency domain a speed up in an order of magnitude can be achieved during the detection process [1-3].

In the detection phase, a sub image  $I$  of size  $m \times n$  (sliding window) is extracted from the tested image, which has a size  $P \times T$ , and fed to the neural network. Let  $W_i$  be the vector of weights between the input sub image and the hidden layer. This vector has a size of  $m \times n$  and can be represented as  $m \times n$  matrix. The output of hidden neurons  $h(i)$  can be calculated as follows:

$$h_i = g \left( \sum_{j=1}^m \sum_{k=1}^n X_i(j,k) I(j,k) + b_i \right) \quad (1)$$

where,  $g$  is the activation function and  $b(i)$  is the bias of each hidden neuron ( $i$ ). Eq.1 represents the output of each hidden neuron for a particular sub-image  $I$ . It can be obtained for the whole image  $Z$  as follows:

$$h_i(u,v) = g \left( \sum_{j=-m/2}^{m/2} \sum_{k=-n/2}^{n/2} X_i(j,k) Z(u+j, v+k) + b_i \right) \quad (2)$$

Eq.2 represents a cross correlation operation. Given any two functions f and d, their cross correlation can be obtained by:

$$f(x,y) \otimes d(x,y) = \left( \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f(x+m, y+n) d(m,n) \right) \quad (3)$$

Therefore, Eq.2 may be written as follows:

$$h_i = g(Z \otimes X_i + b_i) \quad (4)$$

where,  $h_i$  is the activity of the hidden neuron (i) when the sliding window is located at position (u,v) and (u,v)  $\in [P-m+1, T-n+1]$ .

Now, the above given cross correlation can be expressed in terms of the Fourier Transform:

$$Z \otimes X_i = F^{-1}(F(Z) \bullet F^*(X_i)) \quad (5)$$

Hence, by evaluating this cross correlation between the input image and the weights of the hidden layer, a speed up ratio can be obtained compared to conventional neural networks. Also, the final output of the neural network can be evaluated as follows:

$$O(u,v) = g \left( \sum_{i=1}^q w_o(i) h_i(u,v) + b_o \right) \quad (6)$$

$O(u,v)$  is the output of the neural network when the sliding window located at the position (u,v) in the input image Z, and  $w_o$  is the vector of weights between the hidden and the output layer.

The complexity of cross correlation in the frequency domain can be analyzed as follows [1]:

1- For a tested image of  $N \times N$  pixels, the 2D-FFT requires a number equal to  $N^2 \log_2 N^2$  of complex computation steps. Also, the same number of complex computation steps is required for computing the 2D FFT of the weight matrix for each neuron in the hidden layer.

2- At each neuron in the hidden layer, the inverse 2D FFT is computed, so, q backward and (1+q) forward transforms have to be computed. Therefore, for an image under test, the total number of the 2DFFT to compute is  $(2q+1)N^2 \log_2 N^2$ .

3- The input image and the weights should be multiplied in the frequency domain. Therefore, a number of complex computation steps equal to  $qN^2$  should be added.

4- The number of computation steps required by fast neural networks is complex and must be converted into a real version. It is known that the two dimensions Fast Fourier Transform requires  $(N^2/2) \log_2 N^2$  complex multiplications and  $N^2 \log_2 N^2$  complex additions. Every complex multiplication is realized by six real floating point operations and every complex addition is implemented by two real floating point operations. So, the total number of computation steps required to obtain the 2D-FFT of an  $N \times N$  image is [1]:

$$\rho = 6((N^2/2) \log_2 N^2) + 2(N^2 \log_2 N^2) \quad (7)$$

which may be simplified to:

$$\rho = 5(N^2 \log_2 N^2) \quad (8)$$

Performing complex dot product in the frequency domain also requires  $6qN^2$  real operations.

5- In order to perform cross correlation in the frequency domain, the weight matrix must have the same size as the input image. So, a number of zeros  $= (N^2 - n^2)$  must be added to the weight matrix. This requires a total real number of computation steps  $= q(N^2 - n^2)$  for all neurons. Moreover, after computing the FFT2 for the weight matrix, the conjugate of this matrix must be obtained. So, a real number of computation steps  $= qN^2$  should be added in order to obtain the conjugate of the weight matrix for all neurons. Also, a number of real computation steps equal to N is required to create butterflies complex numbers ( $e^{jk(2\pi n/N)}$ ), where  $0 < K < L$ . These  $(N/2)$  complex numbers are multiplied by the elements of the input image or by previous complex numbers during the computation of FFT2. To create a complex number requires two real floating point operations. So, the total number of computation steps required for fast neural networks becomes [1]:

$$\sigma = ((2q+1)(5N^2 \log_2 N^2) + 6qN^2 + q(N^2 - n^2) + qN^2 + N) \quad (9)$$

which can be reformulated as:

$$\sigma = ((2q+1)(5N^2 \log_2 N^2) + q(8N^2 - n^2) + N) \quad (10)$$

6- Using a sliding window of size  $n \times n$  for the same image of  $N \times N$  pixels,  $(q(2n^2 - 1)(N - n + 1)^2)$  computation steps are required when using traditional neural networks for object/face detection process. The theoretical speed up factor  $\eta$  can be evaluated as follows [3]:

$$\eta = \frac{q(2n^2 - 1)(N^2 - n^2 + 1)}{(2q + 1)(5N^2 \log_2 N^2) + q(8N^2 - n^2) + N} \quad (11)$$

### III. A NEW FASTER ALGORITHM FOR HUMAN FACE DETECTION BASED ON IMAGE DECOMPOSITION

In this section, a new faster algorithm for face detection is presented. The number of computation steps required for fast neural networks with different image sizes is listed in Table 1. From this table, we may notice that as the image size is increased, the number of computation steps required by fast neural networks is much increased. For example, the number of computation steps required for an image of size (50x50 pixels) is much less than that needed for an image of size (100x100 pixels). Also, the number of computation steps required for an image of size (500x500 pixels) is much less than that needed for an image of size (1000x1000 pixels). As a result, for example, if an image of size (100x100 pixels) is decomposed into 4 sub-images of size (50x50 pixels) and each sub-image is tested separately, then a speed up factor for face detection can be achieved. The number of computation steps required by fast neural networks to test an image after decomposition can be calculated as follows [7]:

- 1- Assume that the size of the image under test is (NxN pixels).
- 2- Such image is decomposed into  $\alpha$  (LxL pixels) sub-images. So,  $\alpha$  can be computed as:

$$\alpha = (N/L)^2 \quad (12)$$

- 3- Assume that, the number of computation steps required for testing one (LxL pixels) sub-image is  $\beta$ . So, the total number of computation steps (T) required for testing these sub-images resulting after the decomposition process is:

$$T = \alpha \beta \quad (13)$$

To detect a face of size 20x20 pixels in an image of any size by using fast neural networks after image decomposition into sub-images, the optimal size of these sub-images must be computed. From Table 1, we may conclude that, the most suitable size for the sub-image which requires the smallest number of computation steps is 25x25 pixels. A comparison between the speed up ratio for fast neural networks and fast neural networks after image decomposition with different sizes of the tested images is listed in Table 2 (n=20, q=30). The speed up ratio is increased with the size of the input image when using fast neural networks and image decomposition. This is in contrast to using only fast neural networks.

### IV. SIMULATION OF THE FAST FACE DETECTION PROCESS (AFTER IMAGE DECOMPOSITION) USING PARALLEL PROCESSING TECHNIQUES

In the previous section, a new algorithm for face detection based on decomposing the image under test to many sub-images has been presented. Then, for each sub-image, a fast neural network has been used to detect the presence/absence of human faces. Here, to further reduce the running time as well as increase the speed up ratio of the detection process, a parallel processing technique is used. Each sub-image is tested using a fast neural network simulated on a single processor or a separated node in a clustered system. The number of operations ( $\omega$ ) performed by each processor / node (sub-images tested by one processor/node) =

$$\omega = \frac{\text{The total number of sub - images}}{\text{Number of Processors / nodes}} \quad (14)$$

$$\omega = \frac{\alpha}{Pr} \quad (15)$$

where, Pr is the Number of Processors or nodes.

The total number of computation steps ( $\gamma$ ) required to test an image by using this approach can be calculated as:

$$\gamma = \omega \beta \quad (16)$$

As shown in Table 3, using a symmetric multiprocessing system with 16 parallel processors or 16 nodes in either a massively parallel processing system or a clustered system, the speed up ratio (with respect to conventional neural networks) for human face detection is increased. A further reduction in the computation steps can be obtained by dividing each sub-image into groups. For each group, the neural operation (multiplication by weights and summation) is performed for each group by using a single processor. This operation is done for all of these groups as well as other groups in all of the sub-images at the same time. The best case is achieved when each group consists of only one element. In this case, one operation is needed for multiplication of the one element by its weight and also a small number of operations ( $\epsilon$ ) is required to obtain the over all summation for each sub-image. If the sub-image has  $n^2$  elements, then the required number of processors will be  $n^2$ . As a result, the number of computation steps will be  $\alpha q(1+\epsilon)$ , where  $\epsilon$  is a small number depending on the value of n. For example, when n=20, then  $\epsilon=6$  and if n=25, then  $\epsilon=7$ . The speed up ratio can be calculated as:

$$\eta = O((2n^2 - 1)(N - n + 1)^2 / \alpha(1 + \epsilon)) \quad (17)$$

Moreover, if the number of processors =  $\alpha n^2$ , then the number of computation steps will be  $q(1+\epsilon)$ , and the speed up ratio becomes:

$$\eta = O((2n^2 - 1)(N - n + 1)^2 / (1 + \epsilon)) \quad (18)$$

Furthermore, if the number of processors =  $q\alpha n^2$ , then the number of computation steps will be  $(1 + \epsilon)$ , and the speed up ratio can be calculated as:

$$\eta = O(q(2n^2 - 1)(N - n + 1)^2 / (1 + \epsilon)) \quad (19)$$

In this case, as the length of each group is very small, then there is no need to apply cross correlation between the input image and the weights of the neural network in frequency domain.

## V. CONCLUSIONS

A faster neural network approach has been introduced to identify frontal views of human faces. Such approach has decomposed the image under test into many small in size sub-images. A simple algorithm for fast face detection based on cross correlations in the frequency domain between the sub-images and the weights of the neural net has been presented in order to speed up the execution time. Furthermore, simulation results have shown that, using a parallel processing technique, large values of speed up ratio could be achieved. Moreover, by using fast neural networks and image decomposition, the speed up ratio has been increased with the size of the input image. The proposed approach can be applied to detect the presence/absence of any other object in an image.

## REFERENCES

- [1] H. M. El-Bakry, "Human Iris Detection Using Fast Cooperative Modular Neural Networks and Image Decomposition," *Machine Graphics & Vision Journal (MG&V)*, vol. 11, no. 4, 2002, pp. 498-512.
- [2] H. M. El-Bakry, "Automatic Human Face Recognition Using Modular Neural Networks " *The International Journal on Machine Graphics*, Vol. 10, No. 1, 2001, pp. 47-73.
- [3] H. M. El-Bakry, "Comments on Using MLP and FFT for Fast Object/Face Detection," *Proc. of IEEE IJCNN'03*, Portland, Oregon, pp. 1284-1288, July, 20-24, 2003.
- [4] R. Klette, and Zamperon, "Handbook of image processing operators," John Wiley & Sons, Ltd, 1996.
- [5] H. A. Rowley, S. Baluja, and T. Kanade, "Neural Network - Based Face Detection," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 20, No. 1, pp. 23-38, 1998.
- [6] H. Schneiderman and T. Kanade, "Probabilistic modeling of local appearance and spatial relationships for object recognition," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 45-51, SantaBarbara, CA, 1998.
- [7] H. M. El-Bakry, "Face detection using fast neural networks and image decomposition," *Neurocomputing Journal*, vol. 48, 2002, pp. 1039-1046.
- [8] R. Feraud, O. Bernier, J. E. Viallet, and M. Collobert, "A Fast and Accurate Face Detector for Indexation of Face

Images," *Proceedings of Fourth IEEE International Conference on Automatic Face and Gesture Recognition*, Grenoble, France, 28-30 March, 2000.

Table 1: The number of computation steps required by fast neural networks (FNN) for images of sizes (25x25 - 1050x1050 pixels).

Image size	No. Of computation steps in case of using FNN	Image size	No. of computation steps in case of using FNN
25x25	1.9085e+006	550x550	1.7524e+009
50x50	9.1949e+006	600x600	2.1130e+009
100x100	4.2916e+007	650x650	2.5096e+009
150x150	1.0460e+008	700x700	2.9426e+009
200x200	1.9610e+008	750x750	3.4121e+009
250x250	3.1868e+008	800x800	3.9186e+009
300x300	4.7335e+008	850x850	4.4622e+009
350x350	6.6091e+008	900x900	5.0434e+009
400x400	8.8203e+008	950x950	5.6623e+009
450x450	1.1373e+009	1000x1000	6.3191e+009
500x500	1.4273e+009	1050x1050	7.0142e+009

Table 2: The speed up ratio in case of using FNN and FNN after image decomposition into sub-images (25x25 pixels) for images of different sizes.

Image size	Speed up ratio in case of using (FNN)	Speed up ratio in case of using FNN after image decomposition
50x50	2.505 2	4.5871
100x100	3.6646	8.9997
150x150	3.9325	10.7600
200x200	4.0045	11.6707
250x250	4.0136	12.2228
300x300	3.9985	12.5923
350x350	3.9736	12.8565
400x400	3.9449	13.0547
450x450	3.9151	13.2088
500x500	3.8855	13.3320

Table 3: The speed up ratio in case of using FNN after image decomposition into sub-images (25x25 pixels) for images of different sizes using 16 parallel processors or 16 nodes.

Image size	Speed up ratio
50x50	73.3935
100x100	143.9953
150x150	172.1592
200x200	186.7312
250x250	195.5652
300x300	201.4760
350x350	205.7032
400x400	208.8745
450x450	211.3405
500x500	213.3126

# APPLICATIONS OF NEUROFUZZY TRAINING ALGORITHMS TO SIMULATION METAMODELLING

Galina V. Merkurjeva and Liana E. Napalkova  
Department of Modelling and Simulation  
Riga Technical University  
1 Kalku Street  
LV – 1658, Riga, Latvia  
E-mail: gm@itl.rtu.lv, liana@factm.com

## KEYWORDS

Simulation metamodelling, NeuroFuzzy training, approximation error.

## ABSTRACT

NeuroFuzzy learning algorithm for simulation metamodelling is described in the paper. Here, the simulation experimental data are used to train a fuzzy neural network-based simulation metamodel and to generate a set of relevant decision rules. Regression type model is applied to define the structure of the metamodel training set and essential decrease of an approximation error of simulation output data is received. The research results in the paper are illustrated with a range of experiments performed.

## INTRODUCTION

Simulation metamodelling is based on substitution of the current simulation model by its metamodel in order to perform further model manipulations such as conducting sensitivity analysis of the model, testing hypotheses regarding the real system, predicting of the model behaviour and optimisation of its parameters, etc. Usually, these actions lead to decreasing approximation accuracy of the investigated process necessary to make relevant conclusions and effective decisions.

Traditionally, statistical metamodels, such as regression ones, are used to approximate simulation model input-output relationships. Mostly, these models are applied to understand the behaviour of the simulation model and to perform its sensitivity analysis. Nevertheless, regressions models are not usable for extrapolation purpose. In general, regression models need a priori knowledge to define the kind of an appropriate function, and in many applications the resulting simulation metamodels don't ensure high approximation accuracy. In this sense, machine learning techniques, such as artificial neural network (ANN) - based models, could be more preferable. Moreover, learning techniques based on fuzzy computing well suit to deal with 'stochastic' data as they allow modelling soft data points.

ANN-based models provide universal means for data approximation and, at least theoretically, enable to discover relationships in simulation experiments with any accuracy degree. Nevertheless, in practice, these models require a great number of simulation experimental data. To solve this problem optimisation of the training set structure for generating a neural network is proposed in the paper. To define the training set structure dynamic or static regression type models are used. General features of this approach are described in the next section.

## ANN-BASED APPROXIMATIONS

Let's consider a set of  $K$  data points from simulation experiments specified by a set of input variables, or factors  $U = \{u_{i,k}\}, i = \overline{1, N}$ , and output variable  $y_k$ , where  $N$  is a number of input variables, and  $k = \overline{1, K}$ .

Let's suppose, that causal relationships between simulation input and output variables are represented by simulation metamodel, or approximation  $f$ , as:

$$\hat{y}_k = f(u_{i,k}, \delta), \quad (1)$$

where  $\delta$  represents an approximation error defined by an average linear error:

$$\delta = \frac{\sum_{k=1}^K |y_k - \hat{y}_k|}{K}. \quad (2)$$

In case of ANN-based approximations in (1), a set of  $K$  experimental points present a metamodel training set.

To decrease an approximation error  $\delta$ , the training set structure could be transformed into another one (see, Figure 1) by including variables (e.g. lags, quadratic or interaction factors) that could be derived from the statistical analysis of simulation experimental data. As a result, ANN-based simulation metamodel  $f'$  with an approximation error  $\delta' < \delta$  could be received:

$$\hat{y}_{k'} = f'(u_{i',k'}, \delta'), \quad (3)$$

where  $u_{i',k'} \in U'$ ,  $U' \subseteq U \cup L$  is the modified training set structure,  $L = \{l_{j,k'}\}$  is a set of derived variables,  $j = \overline{1, M}$ ,  $i' = \overline{1, N'}$ ,  $N' \leq M+N$ ,  $k' = \overline{1, K'}$ ,  $K' \leq K$ , and  $M$  is a number of derived variables,  $N'$  is a total number of variables selected.

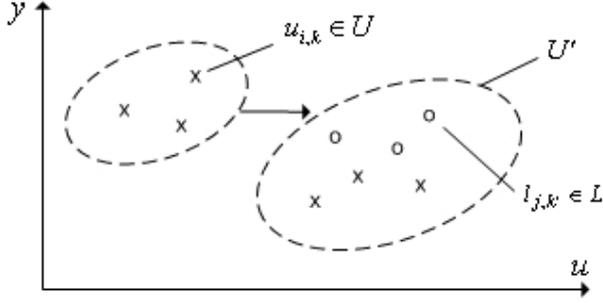


Figure 1: Transformation of the metamodel training set structure

The above-mentioned ANN-based models allow describing complex non-linear functional dependencies between simulation inputs and outputs in automatic mode. Both general and specialized ANN networks include training algorithms to set-up their parameters. However, general ANN networks cannot describe causal relationships in the simulation model because generated ANN model operates as “Black Box” also being poor from a standpoint of its verifiability. In this sense, specialized models, such as Adaptive-Network-based Fuzzy Inference System (ANFIS) could be more preferable for simulation metamodeling. It could be used to generate the decision rules in order to develop metamodeling knowledge base. In this case, simulation generated data are used to inquire and represent knowledge in the form of production rules as:

$$\begin{aligned} \text{IF} & \quad (u_1 \text{ is } A_1) \wedge (u_2 \text{ is } B_1) \\ \text{THEN} & \quad z_1 = p_1 u_1 + q_1 u_2 + w_1 \end{aligned} \quad (4)$$

where  $A_1, B_1$  are linguistic terms,  $p_1, q_1, w_1$  are coefficients of linear equation, and  $z_1$  is a real number.

To generate fuzzy productions (4), Sugeno inference multi-layer algorithm (Fuller 1995) is applied.

Layer 1 defines the degree to which the given input satisfies the linguistic terms, such as  $A_1, A_V, D_1, D_V$ .

Layer 2 computes the firing levels of the rules as:

$$\alpha_1 = A_1(u_1) \wedge D_1(u_N), \dots, \alpha_R = A_V(u_1) \wedge D_V(u_N),$$

where  $V$  is a number of linguistic terms and  $R$  is a number of rules in a rule base.

Layer 3 normalizes these firing levels by the following formulas:

$$\beta_1 = \frac{\alpha_1}{\alpha_1 + \alpha_R}, \quad \beta_R = \frac{\alpha_R}{\alpha_1 + \alpha_R}.$$

Layer 4 performs the production of the normalized firing level and the individual rule output.

At last, layer 5 computes the overall output of the system:

$$z_0 = \sum_{r=1}^R \beta_r z_r^*.$$

### NEUROFUZZY TRAINING ALGORITHM

To generate the training data set structure for ANN-based simulation metamodel, a dynamic regression is applied.

Dynamic regression models, generally called as Autoregressive Distributed Lag or ADL models (Pindyck and Rubinfeld, 1998), can be presented by the following equation:

$$y_t = \vartheta_0 + \sum_{j=1}^P \vartheta_j y_{t-j} + \sum_{i=1}^N \sum_{j=0}^R \varphi_{ij} u_{i,t-j} + \varepsilon_t \quad (5)$$

where  $y_{t-j}$  is a vector of values of an output variable observed at the time  $(t-j)$ ,  $\vartheta_j, \varphi_{ij}$  are regression coefficients,  $N$  is a number of explanatory variables,  $P$  and  $R$  are lag lengths, and  $\varepsilon_t$  is an error term. Let note, that comparing with static regression models dynamic ones follow changes of the simulation model behaviour on time and enable to increase approximation accuracy of the underlying dependencies.

The corresponding NeuroFuzzy training (NTF) algorithm (Merkuryeva and Napalkova 2004) aimed to build ANN-based simulation metamodel includes the following steps:

1. ADL model identification.
2. Generating the training set structure according to ADL model.
3. Generating NeuroFuzzy metamodel structure by fuzzy production rule set.
4. Training simulation metamodel.
5. Simulation metamodel validation.

**Step 1.** ADL model is developed from simulation experimental data.

This step is generally splitted into two actions: 1) analysis of residuals for static regression model that focuses on the test of first-order serial correlation

(Heuchemer 2000), in this action evaluation of Durbin-Watson statistics is performed; 2) transition from a static regression to a dynamic one when first-order serial correlation occurs.

To correct the initial static model lagged variables are added. To detect a lag length an autocorrelation plot or a partial autocorrelation plot is used. To test a higher-order serial correlation an analysis of Box-Ljung statistics is performed. In order to exclude insignificant variables, the regression model is evaluated by R-squared, F-statistics and t-statistics.

Let's assume that residuals are described by the second-order AR(2) process. The static regression model with an AR(2) error could be transformed into the dynamic ADL model by adding two lags of  $y_t$  and  $u_{i,t}$ ,  $i = 1, N$  :

$$y_t = \vartheta_0 + \sum_{j=1}^2 \vartheta_j y_{t-j} + \sum_{i=1}^N \sum_{j=0}^2 \varphi_{ij} u_{i,t-j} + \varepsilon_t. \quad (6)$$

**Step 2.** Training set structure  $U'$  is defined and statistical data for variables included in the training set are generated. Training set structure for the ADL model (6) is presented as follows:

$$\{ \langle y_t, y_{t-1}, y_{t-2}, u_{1,t-1}, u_{1,t-2}, \dots, u_{N,t-1}, u_{N,t-2} \rangle \}. \quad (7)$$

**Step 3.** To generate ANN-based fuzzy metamodel, the following tasks are performed: 1) the type and number of membership functions assigning to each input and output variable are defined; 2) the form of production rules (4) is defined. The set  $K'$  of data points for each input  $u_{i',k'}$  and output variable  $y_{k'}$  is divided into clusters. The number of clusters corresponds to the number of terms for each variable. Numerical values of the rules consequent and premise parameters are estimated at the next step.

**Step 4.** The ANFIS training algorithm is used to identify the rules consequent and premise parameters. This algorithm adjusts the consequent parameters in a forward pass and the premise parameters in a backward pass by using least-squares method and gradient descent method, respectively (Figure 2).

**Step 5.** The approximation error to validate the resulting metamodel is estimated using equation (2). In a case of  $\delta' \geq \delta$ , the structure of the training set  $U'$  is corrected.

Thus, ANN-based simulation metamodel approximates the target functional dependency with a piece-wise linear function and could be described by a set of production rules like:

$$\text{IF } g(u_{i'} \text{ is } A_v) \text{ THEN } \hat{y} = f'(u_{i'}), \quad (8)$$

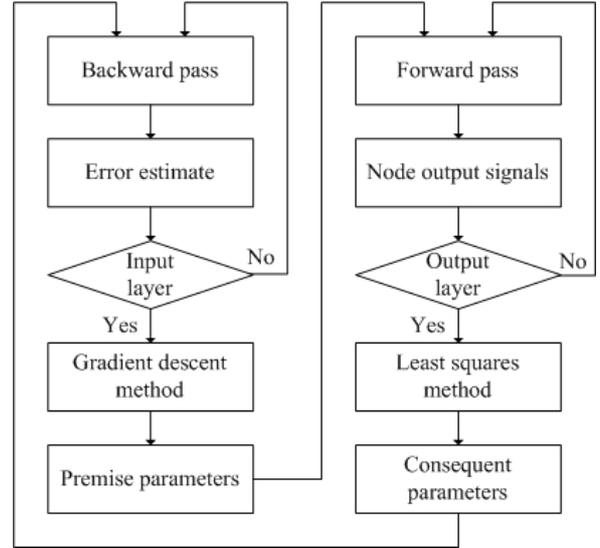


Figure 2: Adjusting production rules parameters

These rules define linear regions for a metamodel, where  $g$  is a logical function that combines propositions of the conditions; and  $f$  is a function of simulation input variables  $u_i \in U'$  that produces a simulation metamodel output variable. The next section presents application and testing results of NTF algorithm.

## NFT ALGORITHM TESTING

The simulation study in (Merkuryeva and Shires 2004) is aimed to improve business operations in the planning department in a medium-sized company. The high-level business/manufacturing simulation model is developed to analyse processing of incoming inquiries of two different types and planning production orders already confirmed by customers. The following controllable variables are defined in this model: 1) inquiries processing time, 2) planning time for orders confirmed. The time between arrivals of inquiries, customer response time, the probability an inquiry becoming confirmed or becoming an order, and order processing time at production stages are regarded as environmental variables. The following simulation output variables are analysed: 1) average lead-time, 2) total revenue, 3) utilization of planners, etc.

To uncover relationships between the enquiries processing time, the orders planning time and an average lead-time, ANN-based simulation metamodels without and with NTF algorithm were built and tested. Two tests for two types of inquiries are performed to illustrate preferability of application NTF algorithm.

In the *Test 1* performed for the first type of inquiries (i.e. Pharmaceutical ones) the training set structure  $U$  consists of two input variables  $u_{1,k}$ ,  $u_{2,k}$  and output variable  $y_k$ , denoting inquiry processing time, order planning time, and an average lead-time in the system,

correspondingly. Each input variable is described by four triangular membership functions (Figure 3).

First, to train ANN-based metamodel without NTF algorithm, 17 data points  $\{<y_k, u_{1,k}, u_{2,k}>\}$  generated from simulation experiments are used. As a result, 16 production rules, like (9), are derived that approximate the input-output behaviour of the basic simulation model:

$$\begin{aligned} \text{IF} & \quad (u_1 \text{ is } A_2) \wedge (u_2 \text{ is } B_3) \\ \text{THEN} & \quad \hat{y}_{4'} = -1549u_1 + 1037u_2 + 98.38 \end{aligned} \quad (9)$$

where  $A_2, B_3$  presents specific membership functions and coefficients present consequent parameters.

In general, the potential number of production rules that represent metamodeling knowledge base is equal  $V^N$ , where  $N, V$  is a number of inputs variables and membership functions for each input, correspondingly. Let's note, that some of that rules could have empty THEN part.

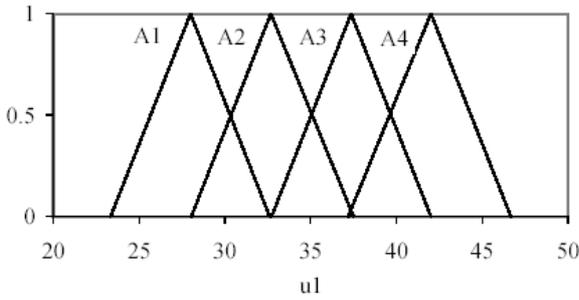


Figure 3: Triangular membership functions for input variable 'inquiry processing time'

Training process required 100 iterations. The estimate of approximation error is equal to 1.388.

Second, to build ANN-based metamodel with NTF algorithm, the regression-correlation analysis is performed. The following regression equation is received:

$$\hat{y} = 9277.03 - 21.05 * u_1 + 4.83 * u_2 + 0.62 * (u_1)^2 + 0.41 * u_1 * u_2$$

From the last equation the training set structure  $U'$  is defined and it includes four input variables as follows:

$$\{<y_{k'}, u_{1',k'}, u_{2',k'}, u_{3',k'}, u_{4',k'} >\}, \quad (10)$$

where  $u_{1',k'} = u_{1,k}, u_{2',k'} = u_{2,k}, u_{3',k'} = (u_{1,k})^2, u_{4',k'} = u_{1,k} \times u_{2,k}$  and  $K' = K$ .

Number of triangular membership functions assigned to each input variable is equal to 4. Training process consists of 100 iterations or epochs (Figure 4). As a

result, 256 production rules are generated that approximate the input-output behaviour of the basic simulation model and represent a new metamodeling knowledge base. The sample production rule is given below:

$$\begin{aligned} \text{IF} & \quad (u_{1'} \text{ is } A_2) \wedge (u_{2'} \text{ is } B_2) \wedge (u_{3'} \text{ is } C_2) \wedge \\ & \quad (u_{4'} \text{ is } D_1) \text{ THEN} \\ \hat{y}_{4'} & = 0.1004u_{1'} + 0.2574u_{2'} + 3.247u_{3'} + \\ & \quad + 8.527u_{4'} + 0.00312, \end{aligned} \quad (11)$$

where  $A_2, B_2, C_2, D_1$  are inputs membership functions, and coefficients present consequent parameters.

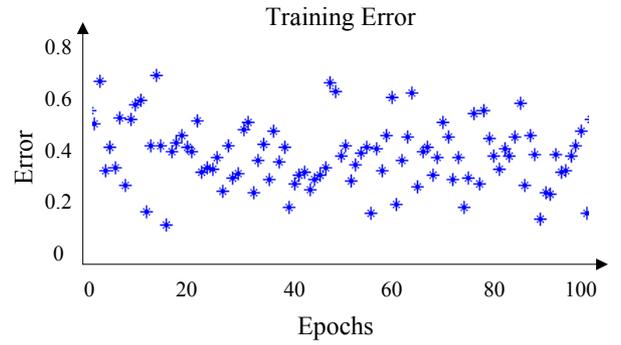


Figure 4: ANFIS training process (Test 1)

ANFIS testing results while comparing training data with metamodeling output data are illustrated in Figure 5. In this case, the estimate of approximation error  $\delta'$  according to formula (2) is equal to 0.4489. It means that  $\delta' < \delta$ .

In the *Test 2* the results of simulation experiments for the second type of inquiries (i.e. Personal care inquiries) are used. ANN-based simulation metamodels without and with NTF algorithm are built and tested.

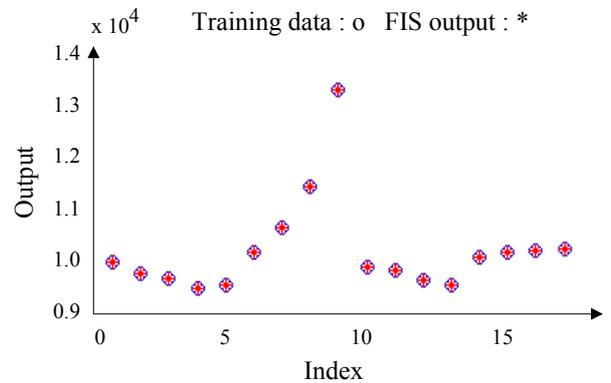


Figure 5. ANFIS testing results

Training set structure for the first metamodel built without NTF algorithm corresponds to one in the first test. Training set structure for the metamodel built with

NFT algorithm includes additional derived variables. Each input variable is described by three triangular membership functions.

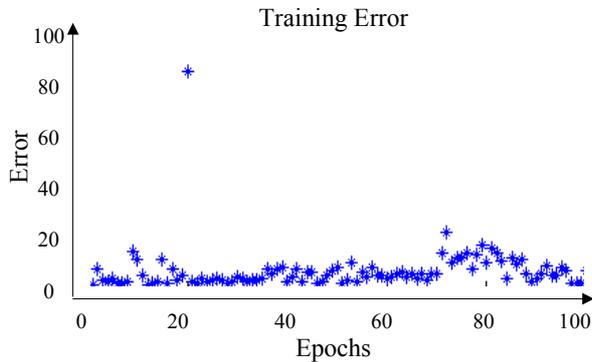


Figure 6: ANFIS training process (Test 2)

After the training process in 100 iterations (Figure 6) corresponding approximation errors  $\delta$  and  $\delta'$  are estimated and they are equal to 12.038 and 2.644, respectively, that gives  $\delta' < \delta$ . Corresponding ANFIS testing results are presented in Figure 7.

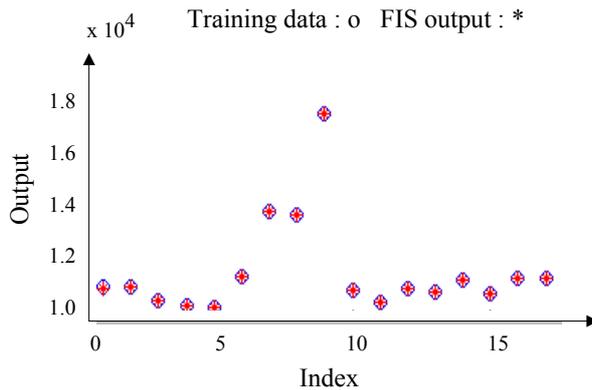


Figure 7. ANFIS testing results

Table 1: Comparative testing results

	Metamodel built without NFT ( $\delta$ )	Metamodel built with NFT algorithm ( $\delta'$ )
Test 1	1.388	0.449
Test 2	12.038	2.644

The results of above tests (Table 1) shows that, ANN-based simulation metamodels built with NeuroFuzzy training algorithm provide lower approximation error of input-output relations in basic simulation models.

## CONCLUSIONS

The developed NFT algorithm for simulation metamodeling is described in the paper. It is based on application of regression models to modify the training set structure of ANN-based fuzzy metamodels. It is indicated that application of NFT algorithm leads to increasing the degree, to which metamodel accurately approximates underlying dependencies in the basic simulation model. Moreover, this algorithm doesn't require expensive and time-consuming experiments.

## REFERENCES

- Esbensen, K. 1994. "Multivariate analysis in practice". In: CAMO, Trondheim.
- Fuller, R. 1995. "Neural Fuzzy Systems". In: Åbo Academy University.
- Heuchemer, S. 2000. "Analyse eines Regressionsmodells". In: Institut für Allgemeine Wirtschaftsforschung.
- Merkuryeva, G. and Napalkova, L. 2004. "Applications of NeuroFuzzy training algorithms to analysis of business processes". In: *Scientific proceedings of Riga Technical University*. Riga, 141-148.
- Merkuryeva, G. and Shires, N. 2004. "SIM-SERV Case study: Simulation-based production scheduling and capacity optimisation". In: *Scientific proceedings of European Simulation Multiconference*. Magdeburg.
- Pindyck, R. and Rubinfeld, D. 1998. "Econometric Models and Econometric Forecasts". In: McGraw-Hill, 4<sup>th</sup> edition.
- Tong, L. 2003. "Inference of Structural Econometric Models: An Unified Approach". In: Indiana University.

## AUTHOR BIOGRAPHIES

**GALINA MERKURYEVA** is Professor at the Department of Modelling and Simulation, Riga Technical University. She holds Doctor of Engineering (Dr.sc.ing) from Institute of Electronics and Computer Techniques of Latvian Academy of Sciences and Doctor of Technical Sciences (Dr.tech.sc.) from Institute of Control Sciences of Russian Academy of Sciences. G. Merkuryeva has professional interests in discrete-event simulation, simulation metamodeling, decision support tools development, and simulation based training.

**LIANA NAPALKOVA** holds a BSc. degree in Computer Science from Riga Technical University. Graduated from Herder School in Riga, Latvia. Currently, she is a research assistant at Department of Modelling and Simulation and develops Master thesis on neural network-based simulation metamodeling.

# ADVANCED TECHNIQUES FOR IMPROVING INDIRECT BRANCH PREDICTION ACCURACY

Adrian Florea and Lucian N. Vintan

Computer Science Department

“Lucian Blaga” University of Sibiu,

E. Cioran Street, No. 4, Sibiu-550025, Romania

Tel./Fax: +40-269-212716, E-mails: [adrian.florea@ulbsibiu.ro](mailto:adrian.florea@ulbsibiu.ro), [lucian.vintan@ulbsibiu.ro](mailto:lucian.vintan@ulbsibiu.ro)

## KEYWORDS

Indirect branches / function calls, dynamic branch and value prediction, execution-driven simulation, SPEC benchmarks, prediction accuracy, correlation information, branch arity.

## ABSTRACT

*Deep pipelines and fast clock rates are necessitating the development of high accuracy branch predictors. From microarchitectural viewpoint, in the last decade the importance of indirect branch prediction increased even though, in the computing programs the indirect jumps remain less frequent than the more predictable conditional branches. One reason refers to predicative execution that implies decreasing of conditional branches number. The dimension took by the desktop, visual or object-oriented applications development (C++, Java – characterized by a large amount of indirect calls comparative to procedural programs), represents another reason which illustrates that indirect branch prediction misses start to dominate the overall misprediction cost. Since the maximum prediction accuracy obtained by a feasible PPM predictor and reported in literature is around 90% implies the necessity of implementing new efficient indirect branch prediction schemes. Thus, we developed a hybrid predictor with arity-based selection that improves indirect branch prediction accuracy reaching in average to 93.77%, comparable with a multi-stage cascaded predictor. We also showed that a modified Target Cache structure based on confidence mechanism and indexed with extended global correlation information represents a more simple and feasible solution that could replace the more complex PPM predictor.*

## I. INTRODUCTION

Deep pipelines and fast clock rates are necessitating the development of high accuracy branch predictors. The most branch prediction research is based on two closely related correlation mechanisms (local and global). The global method exploits correlation between the outcome of a branch and the outcome of neighboring branches that are executed immediately prior to the branch. In contrast, the local method

depends on the observation that the outcome of a specific instance of a branch is determined not simply by the past history of the branch, but also by the previous outcomes of the branch when a particular branch history was observed. Yet branch prediction is a specific example of a far more general time series prediction problem that occurs in many diverse fields of science. It is therefore surprising that there has not been more cross-fertilization of ideas between different application areas. A notable exception is a paper by Mudge (Mudge et al. 1996) that demonstrates that all two-level adaptive predictors implement special cases of the PPM (markovian) algorithm that is widely used in data compression. Mudge uses the PPM algorithm to compute a theoretical upper bound on the accuracy of branch prediction, while Steven (Steven et al 1999) demonstrates how a two-level predictor can be extended to implement the PPM algorithm with a resultant reduction in the misprediction rate. Other researchers developed some more sophisticated predictors based on neural networks algorithms (Jimenez 2002, Egan et al 2003, Sez nec 2004).

A particularly difficult challenge consists in target prediction for indirect jumps and calls. Because the target of an indirect jump (call) can change with every dynamic instance of that jump, predicting the target of such an instruction is really difficult. From microarchitectural point of view object-oriented programming techniques exercise different aspects of computer architecture to support the object-oriented programming style (Calder and Grunwald 1994, Florea et al. 2004). In the last decade the importance of indirect branch prediction increased even though, in the computing programs the indirect jumps and calls remain less frequent than the more predictable conditional branches. One of the reasons refers to predicative execution that implies decreasing of conditional branches number. The dimension took by the desktop, visual or object-oriented applications development (C++, Java – characterized by a large amount of indirect branches comparative to procedural programs), and respectively, the portability trend of many of them, as well as the usage of Dynamically-Linked Libraries, represent other reasons which illustrate that indirect branch prediction misses start to dominate the overall misprediction cost. Knowing that the Pentium4 equivalent processor performance degrades by 0.45% per additional branch missprediction cycle and, additional, a very small number of static indirect

branches is responsible for more than 90% of dynamic indirect jumps (Calder and Grunwald 1994, Florea et al. 2004), results that the overall performance of architectures are very sensitive to indirect branch prediction. Since the prediction accuracy generated by classical schemes Branch Target Buffer or Last Value Predictor is less than 75% (Chang et al 1997) and, the maximum value obtained by a feasible PPM (Partial Prefix Matching) predictor, reported in literature is around 90%, implies the necessity of implementing new performing indirect branch prediction schemes (hybrid or cascaded predictors, or even adapted from value prediction – contextual, PPM predictors).

One goal of this paper is to show that a modified Target Cache structure based on confidence mechanism and indexed with extended global correlation information represents a more simpler and feasible solution that could replace the more complex PPM predictor. We developed a path based predictor derived from the native Target Cache (Chang et al. 1997), attaching a confidence counter (degree of reliability) at every Target Cache location, together with a replacement mechanism based on LRU (least recently used), confidence and both of them. The confidence mechanism performs speculation control by limiting the prediction to those that are likely to be correct. We also extend the global correlation information to identify the right context of indirect jump occurrences. The prediction accuracy generated by the new scheme proposed is only with 0.4% under the accuracy provided by a PPM predictor, but our scheme is much simpler and more feasible to be implemented in hardware.

However, the best result reported in this work consists in developing of a hybrid predictor with arity-based selection. Our new scheme improves indirect branch prediction accuracy reaching in average to 93.77%, comparable with a more complex multi-stage cascaded (Driesen and Hoelzle 1999). We classified branches according to a dynamic measure, the number of different targets encountered in a program run, or branch *arity*. The arity of a branch we determined in a profiling run. The profile information table helps to select which component predictor will predict at every moment. The component predictors are: a *LastValue* predictor (Lipasti et al. 1996, Florea et al. 2002) for monomorphic (1 target) or duomorphic (2 targets) indirect jumps, and respectively, the best contextual (markovian) predictor (Sazeides 1999, Florea et al. 2004) for polymorphic branches (more than 2 targets). Therefore, we determine first, based on laborious simulations what is the optimum search pattern when different contexts are used. Thus, we obtained that the markovian predictor has the best behavior in two different contexts: using a *short history* (the last 32 targets and a search pattern of 3) and a *rich history* (last 256 targets and a search pattern of 6).

The organization of the rest of this paper is as follows. In section II we review related work in the field of indirect branch prediction. Section III describes the implemented predictors. Section IV includes simulation

methodology and experimental results obtained using the simulator that we developed. Finally, section V suggests directions for future works and concludes the paper.

## II. RELATED WORK

Several forms of BTBs, applied to indirect branch prediction, have been studied by Lee (Lee and Smith 1984). The simplest BTB keeps the most recent target for each branch. In the case of a BTB target mispredict, the predicted target address is replaced. An improvement upon that basic BTB configuration was proposed by Calder (Calder and Grunwald 1994), where a two-bit counter is used to limit the update of the target address only after two consecutive mispredictions have occurred (BTB2b strategy). The BTB2b can produce a better branch prediction ratio for C++ applications by taking advantage of the locality exhibited by targets of virtual function calls (C++ polymorphic calls). Similar results we obtained in a previous work (Florea et al. 2004). We showed that target localities for some object oriented test programs and SPEC benchmarks ('95 and 2000) is over 90%, considering different context orders, varying from 1 to 32.

In (Kaeli and Emma 1991) and (Kaeli and Emma 1997) the authors described two mechanisms that accurately predict the targets of two special classes of indirect branches: (i) subroutine returns, and (ii) indirect jumps generated by switch statements. A Call/Return Stack (RAS) was described which uses the inherent correlation between procedure calls and returns to pair up the correct target address with the current return invocation.

In (Driesen and Holzle 1998a) it is examined a modified structure of the two-level adaptive branch predictor to predict the targets of indirect branches. Their design allows path-based correlation to be exploited by recording partial previous targets in the history register (instead of branch directions since the indirect jumps have always the same direction). Driesen performed an exhaustive search on a large number of two-level predictor configurations. The best prediction accuracy was generated using the GAp structure (Global History Register, Per-address Pattern History Table (PHT) configuration). However, the application of their results is limited since they recorded full target addresses in the history register and assumed infinite PHTs. Also, the authors explore realistic designs, varying how and when targets are recorded in the history register, the size and associativity of the PHTs, and the size of the history register. They also proposed using dual-path hybrid predictors with confidence counters to improve prediction accuracy under a fixed hardware budget. Each component was a two-level predictor with a different path length. Their findings suggest that the best dual path predictor had components with the short and a long path length. Some of our results exhibit the same conclusion. If the context would permit it could be seen a correlation between branches

situated at a large distance in the dynamic instruction stream. Whether in the current path-based predictors, the  $N$  most recent target addresses are hashed together to form an index into a table, where  $N$  is some fixed integer, Stark proposed (Stark et al. 1998), using profiling information, to select the proper value of  $N$  for each branch, thus, achieving an extremely accurate branch prediction.

In (Chang et al 1997) was proposed a structure similar to the two-level adaptive scheme, named Target Cache (TC). Its history register recorded partial targets from a selected group of branches (the group may include all branches, indirect branches, conditional branches or calls/returns). Their path-based simulation results showed the dependence of indirect branch predictability on the type of correlation.

In (Driesen and Holzle 1998b) was performed another study about the predictability of indirect branches using filtering. The proposed filtering scheme improve the prediction ratio by isolating monomorphic and low entropy branches from a main body predictor and reducing the collision factor in the later. Their *cascaded predictor* included a GAp/Dual-path hybrid predictor as their major component and strict/leaky filter, implemented as a BTB-like structure. In (Driesen and Holzle 1999) the two-stage prediction was generalized to multistage prediction. At 1.5K entries a three-stage predictor reaches 94% accuracy, the hit rate of a hypothetical two-level predictor with an unlimited, fully associative prediction table. At 6K entries, accuracy increases to 95%, the limit achieved by an idealized twelve-stage cascaded predictor with an unlimited hardware budget.

In (Kalamatianos and Kaeli 1998) the authors applied the PPM algorithm to the task of indirect branch prediction. The PPM predictor shortens a history pattern bit by bit, and looks it up successively smaller stages. Each stage is half the size of its predecessor. The bits correspond to branch targets, so this scheme tests ever shorter path lengths. Kalamatianos combines a viable implementation of the PPM algorithm with dynamic per-branch selection of the path-based correlation. The maximum prediction accuracy reported is 90.53%.

### III. INDIRECT JUMPS TARGET PREDICTION

#### Extending Target Cache structure with a confidence mechanism

The first considered predictor was a tagged Target Cache predictor inspired from that presented in (Chang et al. 1997). The Target Cache improves the prediction accuracy for indirect branches by choosing its prediction from the last  $N$  targets of the indirect branch that have already been encountered. When an indirect jump is fetched, both the PC and the *globalHR* (a history register that retains the behavior of last HRgLength conditional branches) are used to access the TC for predicting the target address. As the program executes, the TC records the target for each indirect

jump target encountered. Our proposed scheme, for set selection, uses the least significant bits of the word obtained by hashing (XOR) the indirect jump's address (PC) and the *globalHR*. The most significant bits of the obtained output form the *Tag*. In the case of a hit in TC the predicted target consists in the corresponding address belonging to that TC set (field *Adr* from figure 1). In case of a misprediction, (the Tags coincide but the target addresses differ) after the indirect branch is resolved, the Target Cache entry is updated with its real target address. It is implemented a LRU replacement algorithm. We have implemented and simulated a  $P$ -way set associative TC, where  $P=1, 2, 4, 8$  like that presented in figure 1. In the case of a miss in TC the prediction is considered wrong, it doesn't propose any value and it is added a new entry in the respective set updating with the proper *tag* and the proper *target*, accordingly with the specified replacement algorithm.

A common criticism for all the present two-level adaptive branch prediction schemes (applied either to conditional branches or indirect jumps) consists in the fact that they used insufficient global correlation information (Vintan and Egan 1999). There are situations when for the same static indirect branch and in the same *globalHR* (see figure 1) context pattern it's possible to find different targets. If each bit belonging to *globalHR* will be associated during the prediction process with its corresponding PC the current indirect branch's context becomes more precisely and therefore its prediction accuracy could be better. Next, we developed a *path based predictor*, through extending the correlation information according to the above idea (Vintan and Egan 1999). Thus, the first level of history of Target Cache predictor records the path (the conditional branches' addresses) leading to the current indirect jump. Extending the correlation information in this way suggests that at different occurrence contexts of a certain indirect jump it will access different sets from TC structure, reducing a significant amount of interferences and increasing the prediction accuracy. Compression of this complex information is possible and even necessary, taking into account the request of reasonable costs for these schemes. The hash function used is a simple XOR.

The next contribution follows up to improve indirect branch prediction accuracy by selectively ignoring some predictions. Therefore, we attached a confidence counter (degree of reliability) at every Target Cache location, together with a replacement mechanism based on *LRU*, *confidence* and on the both metrics superposition called by us *MPP* (performance potential minimum) – see figure 1. A confidence mechanism performs speculation control by limiting the predictions to those that are likely to be correct. A high confidence degree represents *the continuous correct predictability in a given history* of an indirect jump while the LRU field means its *activity degree* (how many times this branch was accessed). Both *Confidence* field and *LRU* field were implemented as saturating counter being represented on different number of bits. For exploiting

this subsection results we used some new metrics, proposed in (Deswet et al. 2002):

We call only here *prediction accuracy* ( $A_p$ ) the probability that prediction generated by a high confident state to be correct. *Usage* represents the prediction performed degree, practically the number of cases when the automaton was in a high confident state (confidence is greater than a certain threshold and TC is ensured to make a prediction), reported to the total number of indirect jumps.

$$A_p = \text{Prob}(\text{correct prediction} \mid \text{High confidence}) = \frac{HC_{\text{corr}}}{HC_{\text{corr}} + HC_{\text{ntcorr}}} \quad (1)$$

$$\text{Usage}(\text{prediction performed degree}) = \frac{HC_{\text{corr}} + HC_{\text{ntcorr}}}{\text{Total\_indirect\_jumps}} \quad (2)$$

We define the *predictors' overall performance* metric, the product:  $P = A_p \cdot \text{Usage}$  (3)

From (1), (2) and (3) results that:

$$P = \frac{HC_{\text{corr}}}{\text{Total\_indirect\_jumps}} \quad (3')$$

We determined how are influenced this metrics by a parameterized threshold.

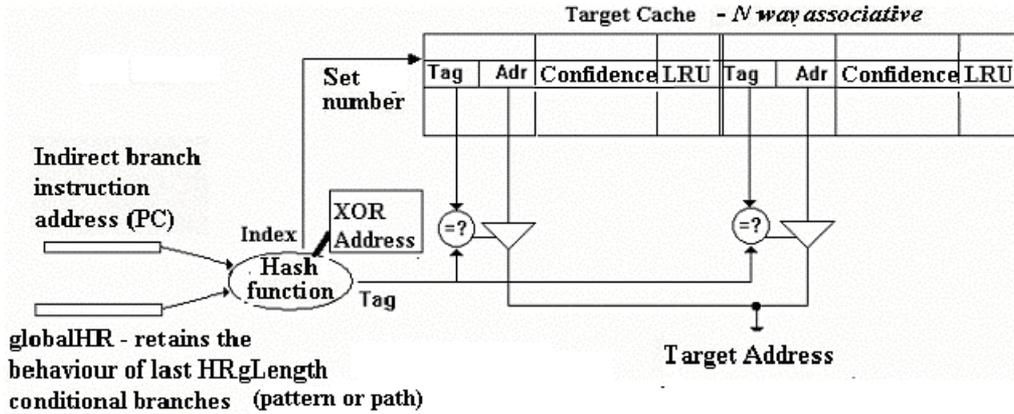


Figure 1. Extending Target Cache Structure with a Confidence Mechanism

### Arity-based selection hybrid predictor

In a previous research (Florea et. al 2004), in order to have an ultimate context predictability metric of indirect branches we measured their target address value locality. The value locality concept was first introduced by Lipasti (Lipasti et al. 1996) and it represents the likelihood of the recurrence of a previously seen value within a storage location. Accordingly, in our case, we'll say that an indirect jump (call) target value is local if it belongs to the previous  $K$  dynamic target instances of that certain jump (call). Obviously, a great target locality degree involves great prediction accuracy, too. In other words, the value locality degree obtained for  $K$  dynamic target instances represents the maximum achievable prediction accuracy using a context predictor of order  $K$ . Therefore, our approach establishes an analogy between value prediction and, respectively, indirect jumps target prediction. Statistical results based on simulations have proved that indirect jumps targets are characterized by higher degree of value locality (over 90% for  $K \geq 4$ ). The main causes for this phenomenon are: the compiler routines that resolve virtual function calls, inheritance and polymorphism from object-oriented programs, indirect calls through function pointers, switch/case statements, etc.

In the same research we find that a complete PPM predictor having an associative indirect jump value prediction table (JVPT) with 256 entries generates average prediction accuracy between 89.33% and 91.58% depending on the context used in simulation:

*short history* (the last 32 targets) and a *rich history* (the last 256 targets), the search pattern varying descending from of 4 to 1. Taking into account PPM's complexity we tried to implement a simplified PPM or a confidence-base hybrid predictor having as components two contextual predictor of different order. Knowing that only a small amount of the path information leading up to a branch is needed for prediction (Stark et al. 1998) we tried to find what the optimum search pattern is when different contexts are used. Thus, we obtained that the markovian predictor has the best behavior in two different contexts: using a *short history* (the last 32 targets and a search pattern of 3) and a *rich history* (last 256 targets and a search pattern of 6).

The values obtained (orders of Markov predictors) we used in developing of a hybrid predictor with arity-based selection. We classified branches according to a dynamic measure, the number of different targets encountered in a program run, or branch *arity*. The arity of a branch we determined in a profiling run. The profile information table, completely associative and retaining the arity of every indirect jump, helps to select which component predictor will predict at every moment. The component predictors are: a *LastValue* predictor for monomorphic or duomorphic indirect jumps, and respectively, the best contextual predictor, previously determined, for polymorphic branches. Both LastValue and contextual predictor are completely associative and are indexed in the instruction fetch stage with indirect branch address (PC).

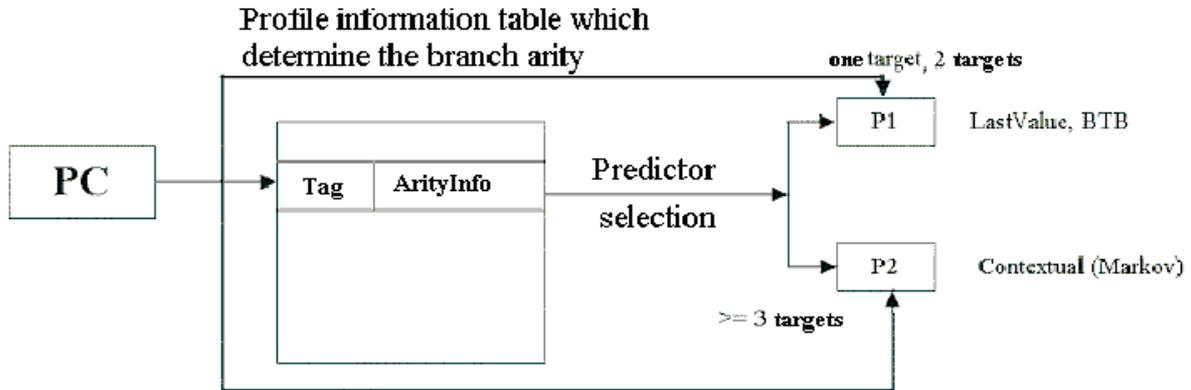


Figure 2. Arity-based Selection Hybrid Predictor

#### IV. SIMULATION METHODOLOGY AND EXPERIMENTAL RESULTS

We developed a cycle-accurate execution driven simulator derived from the *sim-outorder* simulator in the *SimpleScalar* tool set (Burger and Austin 1997). The baseline superscalar processor supports out-of-order instruction issue and execution. We modified it to incorporate the indirect jumps' predictors proposed in section 3 in order to measure target locality, and, respectively to predict targets for indirect jumps and calls.

To perform our evaluation, we collected results from different versions of SPEC benchmarks: 3 integer (*i, go, cc1*) and 4 floating point (*applu, apsi, fpppp, hydro*) SPEC'95 benchmarks. From the CINT SPEC2000 set, it was simulated 8 benchmarks (*gzip, b2zip, parser, crafty, gap, gcc, twolf* and *mcfl*). We simulated some SPEC'95 benchmarks too in order to compare their behavior with that one involved by more recently SPEC2000. In other words, we intend to discover how these different benchmarks influence the indirect jumps predictors' micro-architectural features.

The number of instructions fast forwarded through before starting our simulations is 500 million. We used the `-fastfwd` option in *SimpleScalar* / *PISA* 3.0 to skip over the initial part of execution in order to concentrate on the main body of the programs. Results are then reported for simulating each program for 200 million committed instructions.

For improving indirect branch prediction accuracy the first tentative was to modify the native Target Cache predictor (Chang et al. 1997). In a previous work (Florea et al. 2004) it was studied the potential of native TC and determined the influence of conditional branches global history about prediction. Thus, without taking into account of conditional branches global history, the prediction accuracy produced by a native

TC having 64 4-way associative sets is slightly less than 80%. Repeating the simulation process for a 256 set TC we obtained that the prediction accuracy is practically saturated. Anyway, the obtained results are smaller than those obtained using a complete PPM predictor (see section III). For benchmarks with a high number of targets generated by an indirect branch (*cc1, li*), through indexing the Target Cache with conditional branches global history the indirect branch prediction accuracy increases (in average until 16.93% and also, in *cc1* particular case even 45%). Simulation results on 7 SPEC'95 benchmarks show that the optimum accuracy it is obtained by keeping the behavior (T/NT) of last 8 conditional branches encountered (*globalHR* – global history register on *HRgLength* bits), a pattern longer than this behaving as noise. Even if, in other researches is used a “long” history (Thomas et al. 2003) or “elastic” (variable length depending on every branch) (Stark et al. 1998, Tarlescu et al. 1997), since the simulation results on the native Target Cache structure (Florea et al. 2004) were optimum for a history length of 4 (in average on 7 SPEC'95 benchmarks) or 8 (in average on the 4 benchmarks rich in dynamic indirect jumps – *apsi, cc1, li, hydro2d*) we decided to continue the simulations using a fix value for *HRgLength* (4 or 8). Figure 3 and table 1 are presenting the quantitative benefits of extending context information that indexes TC structure.

For benchmarks with the largest number of indirect jumps, by extending the correlation ( $PC_1, PC_2, \dots, PC_{HRgLength}$ ) and at the same level of complexity (Target Cache structure of the same size), the prediction accuracy increases (with **8.64%** for *HRgLength* of 4 and respectively, with **15.16%** for *HRgLength* of 8 – see table 1). Table 1 illustrates the prediction accuracy in average on *apsi, cc1, li, hydro2d* testing programs.

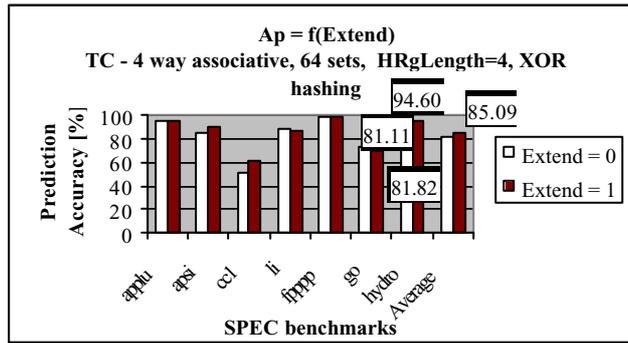


Figure 3. The Influence of Extended Correlation Information about Indirect Branches Prediction Accuracy

Table 1. Increasing the Indirect Jumps Prediction Accuracy by using a more precise Context on Benchmarks with the largest Number of Indirect Jumps, using a TC – 4 way associative; 64 sets

	HRgLength = 4	HRgLength = 8
Extend = 0	76.52%	74.56%
Extend = 1	83.13%	<b>85.86%</b> (respectively <b>88.21%</b> for a TC – 8 way associative)

Although the improvement obtained by extending the correlation information is obvious, in average, the indirect branches prediction accuracy is still lower than that generated by a complete PPM predictor (**89.33%**). However, particularly there are also very good results: prediction accuracy obtained on hydro2d – 99.98%

(HRgLength = 8; TC – 4 way associative; 64 sets, and using rich context  $PC_1, PC_2, \dots, PC_{HRgLength}$ ), is equal with that exhibits by the complete PPM predictor.

Unpleased by previously results we tried to improve indirect branch prediction accuracy by selectively ignoring some predictions.

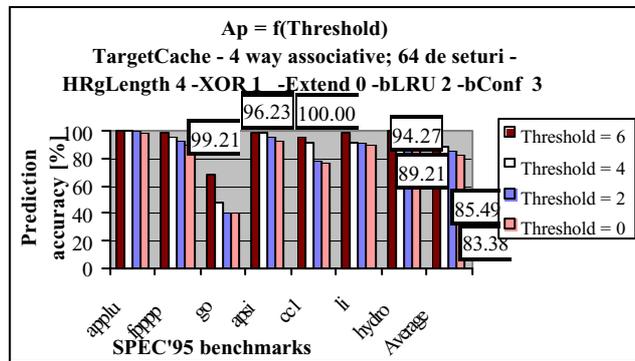


Figure 4. Prediction Accuracy ( $A_p$ ) varying the Threshold, using a Confidence Mechanism

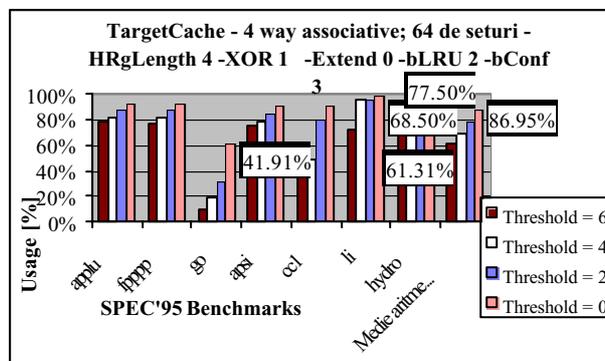


Figure 5. The Prediction performed Degree (the Indirect Jump Fraction having  $Confidence > Threshold$ )

Practically, the probability that prediction generated by a high confidence states ( $A_p$ ) to be correct significantly

increases through reducing the cases when the structure makes a prediction (between 3.57% and 11.45%

depending the *threshold*). The disadvantage is that the percentage of cases in which is made a prediction dramatically decreases. The efficiency of extending correlation information is proved in this case once again (*predictors' overall performance* increases with 5.62% when is used a rich context to identify the current

indirect branch, approaching by the PPM predictors performance). After laborious simulations the conclusion is that attaching a confidence counter the prediction accuracy is improved when this is less selective (see tables 2 and 3).

Table 2. The Influence of Associativity Degree on TC Predictor (with and without Confidence)  
(TC - 64 sets -HRgLength 4 -XOR 1 -Extend 1 -bLRU 2 -bConf 3) - I

Associativity Degree (AD)	Prediction Accuracy (without confidence mechanism)	Predictors' overall performance (with confidence mechanism) - P		
			Threshold = 1	Threshold = 2
AD = 2	78.15%	<	79.39%	>
AD = 4	82.27%	<	82.78%	>
AD = 8	82.35%	<	85.13%	>

Table 3. The Influence of Associativity Degree on TC Predictor (with and without Confidence)  
(TC - 128 de seturi -HRgLength 8 -XOR 1 -Extend 1 -bLRU 2 -bConf 3) - II

Associativity Degree (AD)	Prediction Accuracy (without confidence mechanism)	Predictors' overall performance (with confidence mechanism) - P		
		Threshold=0	Threshold=1	Threshold=2
AD = 4	86.40%	< 87.39%	85.66%	84.98%
AD = 8	86.47%	< 88.88%	> 87.17%	> 86.51%

Increasing the associativity degree greater than 8-way, the predictors' overall performance became asymptotical. Thus, for a 8-way associative Target Cache predictor having 128 sets, keeping the behavior of last 8 conditional branches  $P=88.97\%$ , only with 0.4 % under the accuracy provided by a PPM predictor (89.33%).

Since for a Target Cache 4-way associative the percentage of cases in which there are made replacements according to LRU principle is less than 1% (except *apsi* and *hydro* benchmarks) it results that, increasing the associative degree and implicit decreasing the percentage of conflict misses the influence of LRU field tend to become insignificant. Thus it might be implemented a trivial replacement algorithm (e.g. FIFO) having benefits about reducing

Target Cache structure complexity. Also, the replacement mechanism from Target Cache based upon LRU principle has proved more efficient than that based upon confidence field minimum value (with 2.43%) and even with 0.34% better than the MPP algorithm. This leads to the idea that some indirect branches are replaced before attaining a minimum confidence and later they would prove correct predicted. Furthermore, although it was expected that MPP replacement algorithm to generate the highest prediction accuracy it seems that the minimum values of this metric (MPP) – possible 0 are influenced by the lower confidences (0 – for many times).

The next two figures exhibit what is the *optimum search pattern* when different contexts are used.

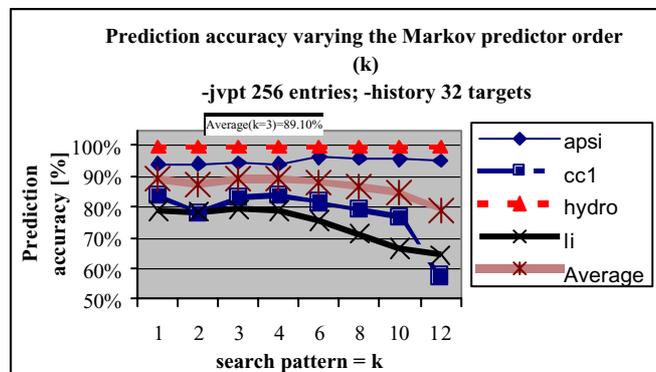


Figure 6. Obtaining the proper Order of Markov Predictor (*poor Context* – 32 targets)

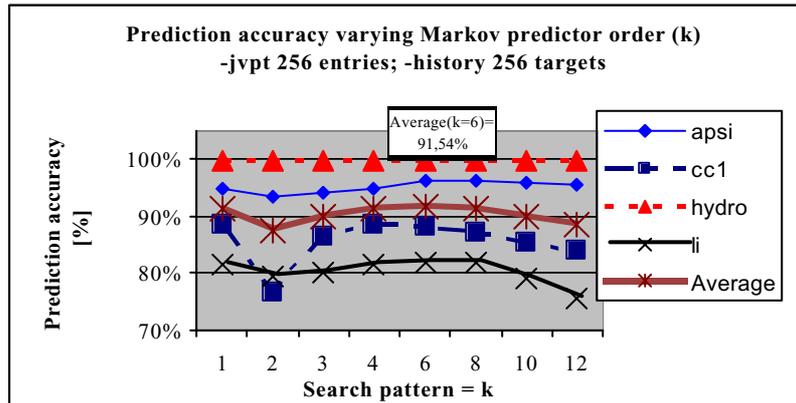


Figure 7. Obtaining the proper Order of Markov Predictor (*rich Context* – 256 targets)

More or less obviously the obtained results suggest that a longer history retained implies a higher prediction accuracy by increasing the search pattern (if the context would permit it could be seen a correlation between branches situated at a large distance in the dynamic instruction stream). The simulation results using a contextual (Markov) predictor showed that for a history of 32 targets the maximum prediction accuracy 89.10% it was obtained for a pattern of 3, while the history became longer the maximum value 91.54% was obtained with a pattern of 6 (Figure 7). This could

suggest implementing a hybrid or cascaded indirect branch predictor with component having different path length. The results emphasize the researchers trends (Thomas et al. 2003) to keep in prediction process a very long history. The authors argue that for a branch under prediction, some of the correlated branches may have appeared at a large distance in the dynamic instruction stream. This can happen if two correlated branches are separated by a call to a function containing many branches.

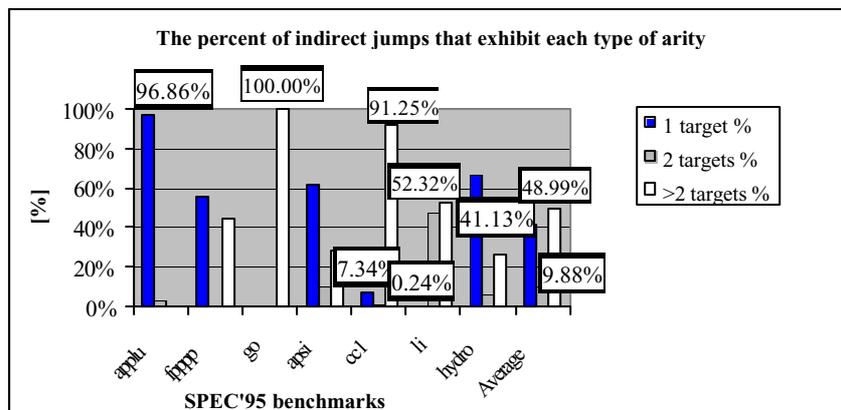


Figure 8. The Indirect Branches Arity –dynamically Point of View

Arity-based classification which classifies indirect branches according to the number of different targets permit us to implement a hybrid predictor having components a Last Value predictor (without history) for a monomorphic branches and the best previous determined contextual predictor for polymorphic jumps. Simulation results on SPEC'95 benchmarks with a large number of indirect branches exhibit 41.13% monomorphic branches, 9.88% duomorphics and 48.99% polymorphics. Our developed hybrid predictor with arity-based selection improves indirect branch

prediction accuracy with percentages between 2.44% and 5.42% reaching in average 93.77%, comparable with that reported in literature (Driesen and Hoelzle 1999). Simulation results on SPEC2000 suite show that three of the eight simulated benchmarks (*gcc*, *crafty*, *twolf*) generate polymorphic indirect branches in proportion of 96%. In our opinion, the significant percentage of polymorphic indirect branches and higher targets entropy specific for some indirect jumps (see the *ccl*, *li* benchmarks) fundamentally limits the indirect jumps prediction accuracy.

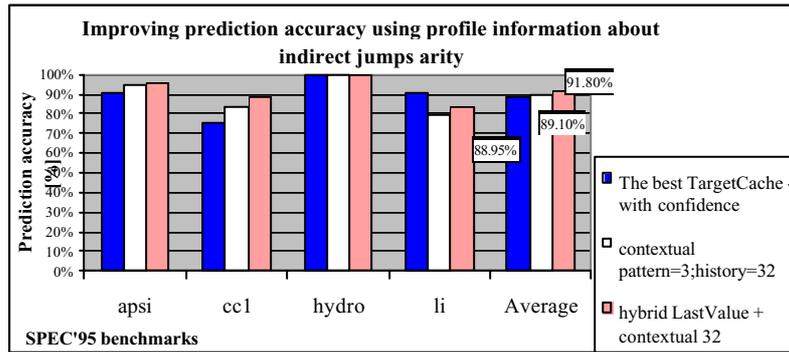


Figure 9. Indirect Branch Prediction Accuracy using a Hybrid Predictor with Arity-based Selection (*poor Context*)

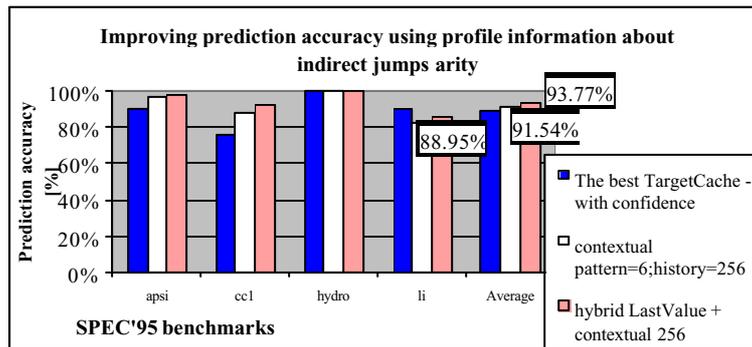


Figure 10. Indirect Branch Prediction Accuracy using a Hybrid Predictor with Arity-based Selection (*rich Context*)

## V. CONCLUSIONS AND FURTHER WORK

Due to higher degree of target localities associated with indirect jumps, we predicted these indirect jumps and calls using some contextual value predictors, derived from the complete PPM predictor respectively the Target Cache predictor. The obtained results were better than those reported by other researchers that used more simplified context predictors. PPM predictor seems to be an almost ultimate limit of context target prediction, and, thus, a good frame for further deriving new practical prediction schemes.

In this sense, we tried to extend the context prediction information adding new correlations. This information, available during the instruction fetch stage in the pipeline, consists of global history register together with its corresponding PCs (one PC for each previous encountered branch (Vintan and Egan 1999)). Using this new correlation information together with the global history register, the current indirect jump's context becomes more precisely and therefore its prediction accuracy is showed to be better. Our first simulation results are encouraging; we showed that a scheme based on this principle performs better than a classical Target Cache scheme (Chang et al. 1997), at the same hardware complexity level. Also, we extend and improve the native Target Cache structure (Chang et al. 1997) with a confidence mechanism for improving the indirect jumps' prediction accuracy. The prediction accuracy generated by the new scheme proposed is only with 0.4 % under the accuracy provided by a PPM predictor, but our scheme is much simpler and feasible

to be implemented in hardware. The best prediction accuracy was obtained using a hybrid predictor with arity-based selection that improves indirect branch prediction accuracy reaching in average to 93.77%, comparable with a more complex multi-stage cascaded predictor.

The excellent results obtained impose introducing and exploiting the hybrid and cascaded predictor in other computer architecture issues to increase the instruction and thread level parallelism: conditional branch prediction, instruction and register value prediction. As a further work we will try to replace the arity-based selection hybrid predictor with a simple neural network which will select dynamically between ordinary component predictors (Last Value, Target Cache, contextual, hybrid). Also we will study the feasibility of an indirect branch predictor correlated and decision trees based. Another solution could appear from development of some "semantic predictors", based on High Level Language applications' information that we prove being important related to indirect jumps generation (polymorphism, indirect function calls, etc.; see our investigations presented in (Florea et al. 2004)). This might be a completely new approach in branch prediction domain, where HLL semantics are often hidden. As far as the architecture designers are concerned, their proposed schemes could be more efficient if not only the object code from benchmarks ("wear off by any semantic information") is analyzed but they will also look "higher" towards high level sources of simulated programs.

## REFERENCES

- Burger D. and T. Austin. 1997. "The SimpleScalar Tool Set, Version 2.0", University of Wisconsin Madison, USA, Computer Science Department, *Technical Report #1342*, June, 1997.
- Calder B. and D. Grunwald. 1994. "Reducing Indirect Function Call Overhead in C++ Programs", In 1994 *ACM Symposium on Principles of Programming Languages*, pages 397-408, January 1994.
- Chang P.Y., E. Hao, Y.N. Patt. 1997. "Target Prediction for Indirect Jumps", *Proceedings of the Int'l Symposium on Computer Architecture*, 1997.
- Deswet V., B. Goeman, K. Bosschere. 2002. "Independent hashing as confidence mechanism for value predictors in microprocessors", *Int'l Conf. EuroPar*, Augsburg, Germany, 2002.
- Driesen K. and U. Holzle. 1998a. "Accurate Indirect Branch Prediction". In *Proceedings of the International Symposium on Computer Architecture*, pages 167-178, Barcelona, Spain, June 1998.
- Driesen K. and U. Holzle. 1998b. "Improving Indirect Branch Prediction With Source- and Aritybased Classification and Cascaded Prediction". *Technical Report TRCS98-07*, Computer Science Department, University of California, Santa Barbara, 1998.
- Driesen K. and U. Holzle. 1999. "Multi-stage Cascaded Prediction". *Euro-Par'99 Conference Proceedings*, Toulouse, France, September 1999.
- Egan, C., G. Steven, P. Quick, R. Anguera, F. Steven, and L. Vintan. 2003. "Two-level branch prediction using neural networks", *Journal of Systems Architecture* 49(12), Elsevier, December, 2003.
- Florea A., L. Vintan, D. Sima. 2002. "Understanding Value Prediction through Complex Simulations". *Proceedings of the 5th International Conference on Technical Informatics*, University "Politehnica" of Timisoara, Romania, October, 2002.
- Florea A., L. Vintan, I.Z. Mihiu. 2004. "Understanding and Predicting Indirect Branch Behavior". *Studies in Informatics and Control Journal: With Emphasis on Useful Applications of Advanced Technology*, March 2004, Vol.13, No. 1, Bucharest.
- Kaeli D. and P. Emma. 1991. "Branch History Table Prediction of moving Target Branches due to Subroutines Returns", *Proceedings of the Int'l Symposium on Computer Architecture*, May 1991.
- Kaeli D. and P. Emma. 1997. "Improving the Accuracy of History-Based Branch Prediction". *IEEE Transactions on Computer Architecture*, 46(4):469-472, April 1997.
- Kalamatianos J. and D. Kaeli. 1998. "Predicting Indirect Branches via Data Compression". In *Proceedings of 31st International Symposium on Microarchitecture*, pages 272-281, Dec. 1998.
- Lee J. and A. Smith. 1984. "Branch Prediction Strategies and Branch Target Buffer Design", *Computer* 17:1, January 1984.
- Lipasti M., C. Wilkerson, P. Shen. 1996. "Value Locality and Load Value Prediction". In *17th ASPLOS International Conference VII*, SUA, 1996.
- Mudge T., I.K. Chen, J.T. Coffey. 1996. "Analysis of Branch Prediction via Data Compression". *Proceedings of the 7th International Conference on ASPLOS VII*, Cambridge, MA, USA, October 1996.
- Sazeides Y. 1999. "An analysis of value predictability and its application to a superscalar processor". *PhD Thesis*, University of Wisconsin-Madison, 1999.
- Seznec A. 2004. "Revisiting the Perceptron Predictor". *IRISA research reports*, IRISA Editeur, May, 2004.
- Steven G.B., C. Egan, P. Quick, and L. Vintan. 1999. "Reducing Cold Start Misspredictions in Two-Level Adaptive Branch Predictors", *Proceedings of the Int'l CSCS-12 Conference*, Bucharest, May 1999.
- Tarlescu M., K. Theobald, and G. Gao. 1997. "Elastic history buffer: A low cost method to improve branch prediction accuracy". In *Proceedings of the IEEE International Conference on Computer Design*, 1997.
- Thomas R., M. Franklin, C. Wilkerson, J. Stark. 2003. "Improving Branch Prediction by Dynamic Dataflow-based Identification of Correlated Branches from a Large Global History". *The 30th Annual International Symposium on Computer Architecture*, San Diego, California, 2003.
- Vintan L. and C. Egan. 1999. "Extending Correlation in Branch Prediction Schemes". *International Euromicro '99 Conference*, Italy, September 1999.

## AUTHOR BIOGRAPHIES



**ADRIAN FLOREA** obtained an MSE (Computer Science) in 1997 and is a PhD student in (Computer Science) since 2000 from the University "Politehnica" of Bucharest, Romania. He is a lecturer of Computer Science and

Engineering at the University "Lucian Blaga" of Sibiu, Sibiu, Romania. Adrian is an active researcher in the fields of High Performance Processor Design and Dynamic Branch Prediction. His Web-page could be found at <http://webspaces.ulbsibiu.ro/adrian.florea>



**LUCIAN VINTAN** obtained an MSE (Computer Engineering) in 1987 and a PhD (Computer Science) in 1997, both from the University "Politehnica" of Timisoara, Romania. He is a Professor of Computer Science and Engineering at the University "Lucian Blaga"

of Sibiu, Romania. Professor Vintan is an active researcher in the fields of High Performance Processor Design and Dynamic Branch Prediction and has collaborated with the Computer Architecture Research Group at the University of Hertfordshire since 1996. His Web-page could be found at <http://webspaces.ulbsibiu.ro/lucian.vintan>

# Design and use of the CPAN Branch & Bound for the solution of the Travelling Salesman Problem (TSP)

Mario Rossainz López  
Facultad de Ciencias de la  
Computación  
Benemérita Universidad Autónoma  
de Puebla  
C/ San Claudio y 14 Sur, San  
Manuel – 72000 Puebla, México  
E-mail: mariorl@siu.buap.mx

Manuel I. Capel Tuñón  
Dpt. Lenguajes y Sistemas  
Informáticos.  
E.T.S. Ingeniería Informática  
Universidad de Granada  
C/ Periodista Daniel Saucedo Aranda  
s/n – 18071 Granada, Spain  
E-mail: mcapel@ugr.es

## KEYWORDS

Parallel Objects, Structured Parallel Programming, High Performance Computing, Object Oriented Programming, Concurrent Programming.

## ABSTRACT

This article presents the design of a High Level Parallel Composition or CPAN (according to its Spanish acronym) that implements a parallelization of the algorithmic design technique named Branch & Bound and uses it to solve the Travelling Salesman Problem (TSP), within a methodological infrastructure made up of an environment of Parallel Objects, an approach to Structured Parallel Programming and the Object-Orientation paradigm. A CPAN is defined as the composition of a set of parallel objects of three types: one object manager, the stages and the Collector objects. By following this idea, the Branch & Bound design technique implemented as an algorithmic parallel pattern of communication among processes and based on the model of the CPAN is shown. Thus, in this work, the CPAN Branch & Bound is added as a new pattern to the library of classes already proposed in [9], which was initially constituted by the CPANs Farm, Pipe and TreeDV that represent, respectively, the patterns of communication Farm, Pipeline and Binary Tree, the latter one implementing the design technique known as Divide and Conquer. As the programming environment used to derive the proposed CPANs, we use C++ and the POSIX standard for thread programming.

## INTRODUCTION

The present investigation centres its attention on the Methods of Structured Parallel Programming, proposing a new implementation, carried out with C++ and the POSIX Threads Library, as a CPAN (Corradi and Zambonelli 1995; Danelutto, and Orlando) of the algorithmic design technique known

as Branch & Bound (BB). CPANs are Structured Parallel constructs based on the Object-Orientation paradigm useful to solve problems of high computational complexity by parallelizing their algorithms using a class of concurrent active objects. In this work the library of classes that we propose in (Rossainz and Capel 2004) is complemented with the design and implementation of the CPAN Branch & Bound, which is intended to provide the programmer with an additional communication and interaction pattern among processes in parallel applications, which allows him to solve optimization problems, such as the Travelling Salesman Problem discussed here, which is an optimization problem with NP-Complete complexity.

## BRANCH AND BOUND METHOD

Branch-and-bound (BB) makes a partition of the solution space of a given optimization problem. The entire space is represented by the corresponding BB *expansion tree*, whose root is associated to the initially unsolved problem; the children at each node represent the subspaces obtained by *branching*, i.e. subdividing, the solution space represented by the parent node; and the leaves of the tree represent nodes that cannot be subdivided any further, thus providing a final value of the cost function associated to a possible solution of the problem. BB carries out a partial enumeration, i.e. a non-exhaustive search, over the nodes of the expansion tree until an optimal solution to the initial problem is found or the set of *live nodes*, i.e. those that still have the possibility of being branched, becomes exhausted.

There are different possibilities to generate nodes and follow a route to a solution during the algorithm execution, known as the branching strategies of the BB algorithm, such as the ones given by the following search methods: *First in depth* (strategy LIFO), *First in width* (strategy FIFO) and *First best node*. The latter uses cost functions calculation to select the node that in principle seems to be more promising to explore, i.e. to further expand in order

to find better solutions from it (strategy HEAP, using minimum cost or LC). In addition to these strategies, BB fixes bounds to the values of the suboptimal solution found at a certain point of the algorithm execution in order to prune those branches below a node that cannot lead to the optimal solution. A bound of the possible value of those reachable solutions is calculated in each node from the information contained in it. If the bound shows that any one of these solutions is necessarily worse than the best solution found up to that point, then there is no need for the algorithm to continue exploring on that branch and, therefore, prunes it off.

### The Algorithm

Three stages are carried out in BB algorithms:

1. **Selection:** A node belonging to the set of live nodes is extracted. The selection directly depends on the strategy search which was decided for use in the algorithm.
2. **Branch:** the node selected in the previous step is subdivided in its children nodes by following a ramification scheme to form the expansion tree. Each child node receives from its father node enough information to enable it to search a suboptimal solution.
3. **Bound:** Some of the nodes created in the previous stage are deleted, i.e. those whose partial cost, which is given by the cost function associated to this BB algorithm instance, is greater than the best minimum bound calculated up to that point.

The contribution of this algorithm is that it offers a way to perform the greatest possible reduction of the space search and, therefore, obtains a decrease in the exploration complexity of the expansion tree which contains the optimal solution being searched. The nodes that have not yet been pruned are included in the live node list, and thereby, the selection process begins again until the algorithm finalizes. The general structure of the algorithms that implement the BB technique is based on three main modules:

1. The module that contains the scheme of general operation of the technique.
2. The module that represents the structure of data where the generated nodes are stored.
3. The module that describes the structures of data that conform the nodes.

The first module is the only one that remains without modification, independently of the problem to solve with the BB algorithm and it is valid for all the algorithms that follow the technique. The pseudo-code implementing it is as follows:

```

CLASS CONCRETE EsquemaBB
{
    Estructura e;
    Nodo n;
    Nodo[] hijos;
    int numhijos,i,j;

    PUBLIC Nodo B&B()
    {
        e= Estructura CREATE();
        n= nodoInicial();
        e.inserta(n,n.h());
        WHILE (!E.esVacía())
        {
            n=e.extrae();
            numhijos=n.expandir(VAR hijos);
            eliminar(n);
            n.poner_cota_sup(numhijos,hijos);
            FOR i=(0,numhijos)
            {
                IF (n.aceptable(hijos[i]))
                {
                    IF (n.esSolucion(hijos[i]))
                    {
                        FOR j=(0,numhijos)
                            IF (i!=j)
                                DELETE hijos[j];
                        e.clear();
                        RETURN hijos[i];
                    }
                    ELSE
                        e.inserta(hijos[i],
                                n.h(hijos[i]));
                }
                ELSE
                    DELETE hijos[i];
            }
        }
    }
}

```

The definition of the abstract data type that represents the structure  $e$  where the nodes are stored corresponds to the ADT HEAP, because the strategy used to search a node containing a solution is that of selecting the minimum cost (LC) one among the contained in the HEAP. The definition of the class that represents the type *Node* used in the previous pseudo-code is composed of the following functions:

- **expandir( ):** It is the function that creates the children nodes out of a given node and returns the number of children to where the function is called. This function is the one that implements the process of node ramification of the algorithm
- **aceptable( ):** Function that carries out the pruning of unpromising nodes and, when it obtains a live node, decides whether to continue exploring or to reject it.
- **esSolucion( ):** It is a function that decides whether its parameter node is a leaf of the tree, that is to say, a possible solution to the original problem.
- **h( ):** This function in its two versions, i.e. it can be overloaded, is the one that implements the cost

function for the search strategy LC and its value is used as a priority position value when storing the nodes in the HEAP structure.

- **poner\_cota\_sup()**: It allows for the upper bound of the problem to be established. The function that carries out node pruning uses this datum to prune those nodes whose cost value is greater than the currently obtained bound of the optimal solution.

The CPANS can provide the parallel algorithms necessary to solve problems, such as the one of the TSP, using the BB technique. When solving this kind of problems, in addition to the solution, reasonable run times of the whole computation can be obtained with the CPAN model. Since the complexity in NP-complete problems as in the TSP one is intrinsic, parallelization is the only way to obtain a solution in the practice. Each object node of the expansion tree therefore needs to be independent that is to say, it must contain all the necessary information to be an active object, (i.e. to have the capability of execution in itself) which makes it possible for the processes of branch and bound to perform the reconstruction of the solution found up to that moment.

## THE TRAVELLING SALESMAN PROBLEM (TSP)

The Travelling Salesman Problem could be represented by a directed graph consisting of a set of vertices (cities) and labelled arcs (distances between cities). One optimized solution of the TSP is a path in which all the vertices have been visited exactly once with minimal cost (Capel and Palma 1992). Formally, the problem can be enunciated as follows: given a connected and weighted graph  $g$  and given one of its vertices  $v_0$  as the initial one, we must find the Hamiltonian cycle of minimum cost that begins and finishes in  $v_0$  (Guerrequeta and Vallecillo).

### Solution of the TSP with the Branch & Bound technique

The problem is solved by a BB algorithm, which dynamically builds a search tree, whose root is the initial problem and its answer nodes are complete tours (Capel and Palma 1992) around all the cities represented by the nodes of the graph. Numerous strategies of Branch & Bound exist that solve the problem of the TSP. The first of the three strategies described in (Horowitz 1978) is used in this work where the following important elements are defined, as well:

- **LC(P) - Cost of a node P**: Is the distance of a complete graph's tour after  $P$  gets included in it, if  $P$  is a solution node, or, otherwise, is the cost of a

solution of minimal cost in the sub-tree whose root is  $P$ .

- **cota - Upper bound**: The length of the shortest complete tour found up until that moment. This value must be a global variable in the program and is used to prune the search tree of nodes.
- **h(P) - lower bound** of the cost of a solution for a problem or sub-problem  $P$ . If  $P$  is a solution node  $LC(P) = h(P)$ .
- **s[] - vector solution**: It is a vector that indicates the order in which the vertices must be visited to reach the optimal solution. Each element of the vector contains a number between 1 and  $N$ , being  $N$  the number of vertices of the graph that defines the problem. The vector cannot contain repeated elements.
- **M[][] - Matrix of adjacency**: It is the representation of the graph where the vertices are the indices of the matrix and the contents of the matrix elements given are the arcs between two vertices. The matrix of adjacency is not necessarily symmetrical, although, it is so with respect to its nonnegative elements.

Computing the lower bound of a node of the search tree is carried out by obtaining the *reduced cost matrix* for each node (Capel and Palma 1992). A row (or column) of a matrix is reduced if it contains an element zero at least, and the rest of the elements are nonnegative. We say that a matrix is reduced if and only if all its rows and columns are reduced (Guerrequeta and Vallecillo). With respect to the interpretation of the cost of a node, this is obtained by adding the amounts  $t_i$  in which the rows or columns of the matrix of adjacency of the graph are reduced when the process of obtaining the reduced matrix is carried out. This amount which is removed when reducing a matrix is a lower bound of the total cost of any possible tour traversing the graph nodes. This is exactly what it is used as, the cost function  $LC$  to prune nodes  $p$  of the expansion tree. Therefore, a *reduced matrix* and an *accumulated cost* are associated to each node and if we suppose that  $M'[][]$  is the reduced matrix associated to the node  $p$  and  $p'$  is a children node of  $p$  which is obtained by including the arc  $\{i,j\}$  in the partially built tour  $s[]$ , then:

- If  $p'$  is a leaf node of the tree, i.e. a possible solution, its cost is the accumulated cost of  $p$  plus  $M[i,j] + M[j,1]$ . The latter term is the one that completes the tour. The amount obtained is the cost of such a tour.
- If  $p'$  is not a leaf, its reduced matrix  $M$  obtained from matrix  $M'$  assigned to this node  $p'$  will have as cost the cost of  $p$  added to the cost of the reduction of  $M'$  added to the value of  $M[i,j]$ .

## DEFINITION OF CPAN

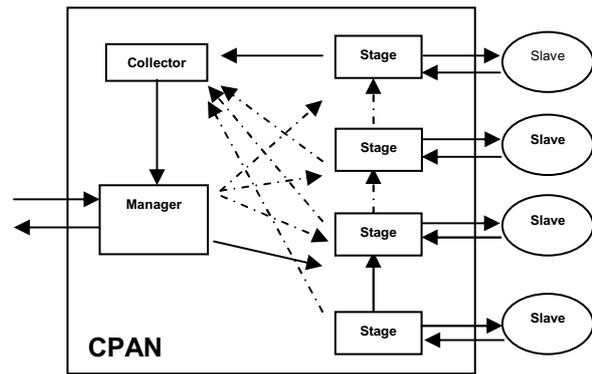
The basic idea is implementing different types of parallel patterns of communication between the processes of an application and implementing distributed/parallel algorithms as classes, by following the Object-Oriented paradigm. The execution of a method of the objects that constitute a CPAN can be carried out through a message sent to an instantiated class, which acknowledges it as a service petition.

A CPAN comes from the composition of a set of objects of three types (see Figure 1):

**An object manager** representing the CPAN itself and makes of it an encapsulated abstraction that hides its internal structure. The manager controls the references of a set of objects (a denominated object Collector and several denominated Stage objects) that represent the components of the CPAN and whose execution is carried out in parallel and should be coordinated by the manager itself.

**The Stage objects** are objects of specific purpose responsible for encapsulating a client-server type interface between the manager and the object slaves (objects that are not actively participative in the composition of the CPAN, but rather, are considered external entities that contain the sequential algorithm that constitutes the solution of a given problem) as well as providing the necessary connection among them to implement the communication semantic pattern that seeks to be defined. In other words, each stage should act in parallel as a node of the graph that represents the communication pattern and should be capable of executing its methods as an active object. A stage can be directly connected to the manager and/or to another component stage depending on the pattern peculiar to the CPAN being implemented.

**And an object Collector** which is an object in charge of storing in parallel the results that it receives from the stage objects that are connected to it. That is to say, during the service of a petition, the control flow within the stages of a CPAN depends on the implemented communication pattern. When the composition concludes its execution, the result does not return to the manager directly, but rather to an instance of the class Collector which takes charge of storing these results and of sending them to the manager, which will then send them to the exterior, as soon as they arrive to it, without the need of waiting for all the results to be obtained.

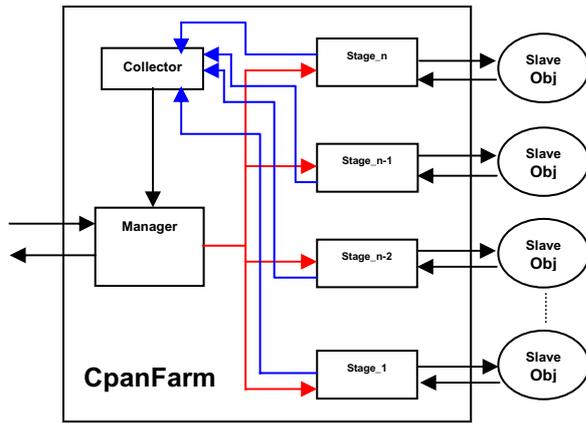


**Figure 1.** Internal Structure of a CPAN (Rossainz and Capel 2004).

The objects manager, collector and stages are included within the definition of Parallel Object (PO) (Corradi and Zambonelli 1995). The Parallel Objects are active objects, that is to say, objects that have execution and communication capabilities in themselves in three ways: the synchronous, the asynchronous and the asynchronous future way (Rossainz and Capel 2004).

In addition, for each one of these, synchronization mechanisms have been implemented when parallel petitions of service take place in a CPAN (*MaxPar*, *Mutex* and *Sync*) (Rossainz and Capel 2004).

The parallel patterns worked in (Rossainz and Capel 2004) have been the pipeline, the farm and the treeDV, to be a significant set of reusable patterns in multiple applications and algorithms. These patterns have been implemented on the basis of the model of the CPAN, constituting a library of High Level Parallel Compositions, formed by *the Cpan Farm*, *the Cpan Pipe* and *the Cpan TreeDV* Making use of the technique of the reusability of code and demonstrating the utility of the library proposed in (Rossainz and Capel 2004), the use of the *Cpan Farm* for the design of the technique of Branch & Bound like a CPAN becomes of interest. *The Cpan farm* is composed of a set of worker processes and a controller. The workers are executed in parallel until reaching a common objective. The controller is the one in charge of distributing work and of controlling the progress of the global calculation. See Figure 2.



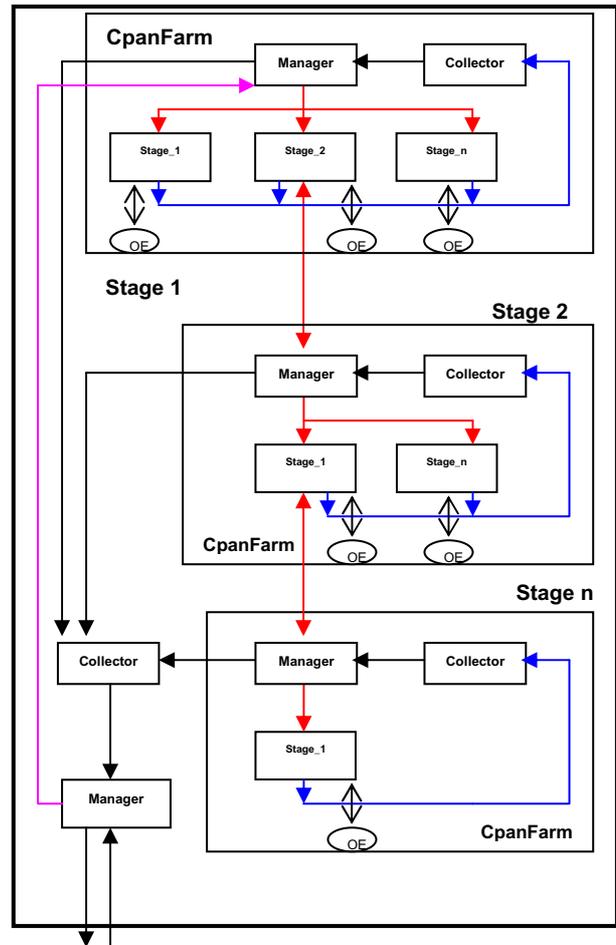
**Figure 2.** The Cpan of a Farm communication pattern (Rossainz and Capel 2004)

### PARALELIZATION OF BRANCH & BOUND TECHNIQUE

The ramification is separated from the bounding of nodes on the expansion tree when the BB algorithm executes. These two structures were implemented using the *Cpan Farm* of the library proposed in (Rossainz and Capel 2004), so that the ramification and the distribution of work to the processes were carried out by using *the scheme of the Farm*. As figure 3 shows, the expansion tree, for a given instance of the BB algorithm, is obtained by iteratively subdividing the stage objects according to the farm pattern until a stage representing a leaf-node of the expansion tree is found, i.e., one stage in charge of solving a sub-problem that cannot be additionally subdivided.

On the other hand, the pruning is carried out implicitly within a *farm* construction by using a *scheme totally connected* between all the processes. It can communicate a sub-optimal bound found in a process to all the processes that are branching to avoid ramifications of useless sub-problems, i.e., those that do not lead to improving the best upper bound obtained up to that moment.

*The Cpan Branch & Bound* is composed of a set of *Cpans Farm* that represent worker processes and a controller, therefore, forming a new type of Farm, the *Farm Branch & Bound* or *FarmBB* that will be included in the library of CPANS. The *Cpan Farm* workers are executed in parallel forming the expansion tree of nodes given by this technique. The process controller of the initial *Cpan Farm* represents the root of the expansion tree that is in charge of distributing the work and of controlling the progress of the global calculation given to the collector of the *FarmBB* which sends the result to the process controller of the *Cpan FarmBB*, which then shows it to the user. See Figure 3.



**Figure 3.** The Cpan Branch & Bound

### RESULTS

The search strategy which was used in the implementation and test of the distributed TSP was: The first best search strategy that uses the calculation of cost functions for each live node to select the node that, in principle, seems the most promising to analyze (strategy HEAP, using minimum cost or LC). The graph taken from reference (GoodMan and Hedetniemi 1977) is an example of execution of the application considered. The graph represents a network of 5 cities whose matrix of costs is as follows:

	0	1	2	3	4
0	$\infty$	25	40	31	27
1	5	$\infty$	17	30	25
2	19	15	$\infty$	6	1
3	9	50	24	$\infty$	6
4	22	8	7	10	$\infty$

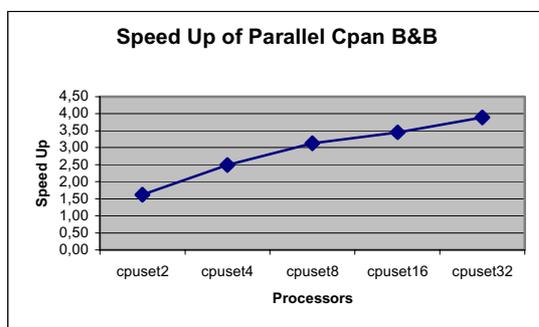
The solution found was the path 0, 3, 4, 2, 1, 0, with a cost of 64, which is the same result found in the example in (GoodMan and Hedetniemi 1977).

## ANALYSIS OF SPEEDUP

The analysis of Speedup of the CPANS B&B that appears in table 1 and Figure 4 was carried out in a Parallel System Origin 2000 Silicon Graphics (of 64 processors) available in the European Center for Parallelism of Barcelona CEPBA.

	CPU Seq.	CPU 2	CPU 4	CPU 8	CPU 16	Cpu 32
Run time	35.42 Seg.	21.88 Seg.	14.21 Seg.	11.34 Seg.	10.27 Seg.	9.10 Seg.
Time CPU	27.10 Seg.	23.25 Seg.	21.17 Seg.	19.19 Seg.	22.69 Seg.	22.18 Seg.
CPI	1.321	0.952	0.943	0.928	0.924	0.914
Speed Up	1.00	1.62	2.49	3.12	3.45	3.89
Amdal h	1.00	1.68	2.55	3.43	4.16	4.64

**Table 1.** Execution of Parallel CpanB&B in 2, 4, 8, 16 and 32 processors with N=50 cities



**Figure 4.** Speed Up of Parallel Cpan B&B

## CONCLUSIONS

1. The technique of Branch & Bound as a High Level Parallel Composition or CPAN has been implemented.
2. The utility of the library of CPANS proposed in (Rossainz and Capel 2004) which serves to make compositions of CPANS and to define new CPANS models as in the *Cpan Branch & Bound* has been demonstrated.
3. With the model of the *Cpan Branch & Bound* we have been able to offer an optimal solution of a TSP NP-Complete problem.
4. The CPANS Pipe, Farm, TreeDV and Farm-Branch-&Bound constitute the library of classes of the Cpan.

## REFERENCES

- Brassard G.; Bratley P. 1998. "Fundamentos de Algoritmia". Prentice may. España. ISBN: 84-89660-00-X
- Capel M.I.; Palma A. 1992. "A Programming tool for Distributed Implementation of Branch-and-Bound Algorithms". Parallel Computing and Transputer Applications. IOS Press/CIMNE. Barcelona.

- Ceballos F.J. 2003. "Programación Orientada a Objetos con C++". Editorial RA-MA. España. ISBN: 84-7897-570-5
- GoodMan S.E.; Hedetniemi S.T. 1997. "Introduction to the Design and Analysis of Algorithms. Mc Graw Hill Book Company. United States of America. ISBN:0-07-023753-0.
- Guerrequeta G.R.; Vallecillo M.A. "Técnicas de Diseño de Algoritmos". Manuales. Universidad de Malaga.
- Troya J.M.; Capel T.M. "An Object Based Tool for Distributed Programming on Transputer Systems"
- Rossainz, Mario; Capel, Manuel. 2004. "A Parallel Programming Methodology based on High Level Parallel Compositions (CPANS)". Proceedings of XIV International Conference on Electronics, Communications, and Computers, CONIELECOMP. ISBN 0-7695-2074-X.
- Corradi A.; Zambonelli F. 1995. "Experiences toward an Object-Oriented Approach to Structured Parallel Programming". DEIS technical report no. DEIS-LIA-95-007.
- Danelutto, M.; Orlando, S; et al. "Parallel Programming Models Based on Restricted Computation Structure Approach". Technical Report-Dpt. Informatica. Università de Pisa.
- Horowitz, Sahni. 1978. "Fundamentals of Computer Algorithms". Ed. Computer Sc. Press.

## AUTHOR BIOGRAPHIES



**MARIO ROSSAINZ LOPEZ** was born in Puebla, México and went to the University of Puebla, where he studied Sciences of Computation and obtained his degree in 1994. He works in the Faculty of Sciences of Computation of the University of Puebla from the year of 1995 where he is now in the research group of Software Engineering in the field of distributed systems. His e-mail address is: [mariorl@siu.buap.mx](mailto:mariorl@siu.buap.mx) and his Web-page can be found at <http://www.cs.buap.mx>.



**MANUEL I. CAPEL TUÑON** was born in Spain and went to the University of Granada, where he studied Physics and obtained the MSC degree in 1982. He worked in the University of Murcia before moving in 1989 to the University of Granada where he is now leading a research group in the field of Concurrent Systems. His e-mail address is: [mcapel@ugr.es](mailto:mcapel@ugr.es) and his Web-page can be found at <http://lsi.ugr.es/~mcapel>.

# Numerical Solution to the Performability of a Multiprocessor System with Reconfiguration and Rebooting Delays

Orhan Gemikonakli  
Middlesex University  
White Hart Lane  
London N17 8HR  
United Kingdom  
E-mail: [o.gemikonakli@mdx.ac.uk](mailto:o.gemikonakli@mdx.ac.uk)

Tien Van Do  
Dept. of Telecommunications  
Budapest University of Technology  
and Economics  
Hungary  
E-mail: [do@hit.bme.hu](mailto:do@hit.bme.hu)

Ram Chakka  
RGM CET  
Nandyal  
India 518501  
E-mail: [ramchakka@yahoo.com](mailto:ramchakka@yahoo.com)

Enver Ever  
Middlesex University  
White Hart Lane  
London N17 8HR, UK  
E-mail: [e.ever@mdx.ac.uk](mailto:e.ever@mdx.ac.uk)

## KEYWORDS

Queuing, multi-processor systems, rebooting and reconfiguration delays.

## ABSTRACT

Multiprocessor system models are extensively used in modelling transaction processing systems, nodes in communication networks, and flexible machine shops with groups of machines. Such systems clearly, are prone to break-downs. Even if cover is provided with some probability  $c$ , there will be rebooting and/or reconfiguration delays to resume operation following the break-down of a processor. In this paper, the performance modelling of a multiprocessor system, with identical processors, serving a stream of arriving jobs is considered. To account for delays due to reconfiguration and rebooting, such systems are modelled and solved for *exact* performability measures for both bounded and unbounded queuing systems.

## INTRODUCTION

Multiserver system models are useful to model multiprocessor systems (Trivedi 2002; Harrison and Patel 1993), nodes in communication networks, and flexible machine shops (Stecke and Kim 1989; Stecke 1992; Righter 1996; Buzacott and Shantikumar 1993; Fiems et al. 2004) in a manufacturing environment. In this paper we develop approaches to model homogeneous multiprocessor systems with reconfiguration and rebooting delays by suitably extending the resulting quasi birth death (QBD) process in the performance models of multiprocessor systems with breakdowns and repair strategies (Chakka and Mitrani 1992; Chakka et al. 2002). This was considered in (Trivedi and Sathaye 1990) and an approximate performance model based on Markov reward models was presented. In this paper, we derive an exact solution for the steady state probabilities of the same problem using the spectral expansion method. The effects of reconfiguration and rebooting delays are analysed.

The paper is organised as follows. The next section presents the homogeneous multiprocessor system with breakdowns and repairs considered in this work, and models the system as a QBD process. The section on modelling reconfiguration and rebooting delays in multiprocessor systems deals with a homogeneous multiprocessor system with breakdowns, repairs, and with reconfiguration and rebooting delays (Trivedi and Sathaye 1990). Exact solution for steady state performability for is derived using the spectral expansion method in the section on steady state solution. The model considered is very useful in the computer industry. Exact solution to this model and numerical results are also presented for both unbounded and bounded systems.

## MULTIPROCESSOR SYSTEM WITH IDENTICAL PROCESSORS

The homogeneous multiprocessor system, shown in Figure 1, consists of  $K$  identical parallel processors, numbered  $1, 2, \dots, K$ , with a common queue. The queue is of capacity  $L$  (finite or infinite  $L \geq K$ ), including the jobs in service. Jobs arrive into the system in a Poisson stream at rate  $\sigma$ , and join the queue. Jobs are homogeneous and the service rates of the processors assumed identical. Thus, the service times of jobs serviced by processor  $k$  ( $k=1, 2, \dots, K$ ) are distributed exponentially with mean  $1/\mu$ . However, processor  $k$  executes jobs only during its operative periods (during an operative period the processor is capable of its intended operation, whether working or idle), which are distributed exponentially with mean  $1/\xi$  (equivalent to a constant failure rate of  $\xi$  when operative). At the end of an operative period, processor  $k$  breaks down and requires an exponentially distributed repair time with mean  $1/\eta$ . The number of repairs that may proceed in parallel could be restricted. This is expressed by saying that there are  $R$  repairmen ( $R \leq K$ ), each of whom can work on at most one repair at a time. Thus, an inoperative period of a processor would also include the

possible waiting for a repairman. No operative processor can be idle if there are jobs awaiting service, and no repairman can be idle if there are broken-down processors waiting for repair. All inter-arrival, service, reconfiguration, rebooting, operative and repair time random variables are independent of each other. The reconfiguration delay  $1/\delta$  and the rebooting delay  $1/\varphi$  relate to the system and not to individual processors.

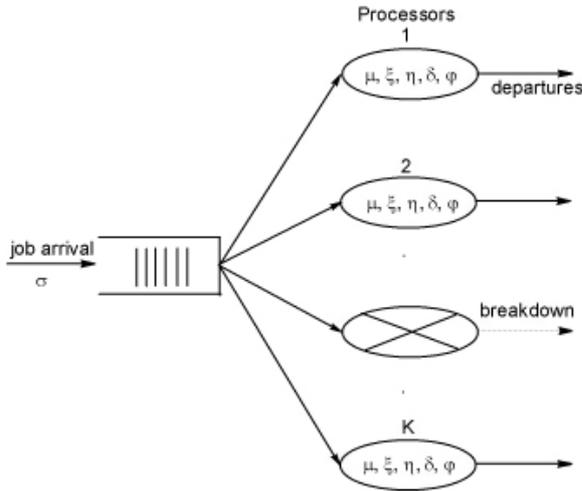


Figure 1: A Homogeneous Multiprocessor System with Breakdowns, Repairs, Reconfiguration and Rebooting Delays

If there are more operative processors than jobs in the system, then the busy processors are selected randomly. Services that are interrupted by breakdowns are eventually resumed (perhaps on a different processor but at a similar service rate). Similarly, if  $R < K$  and the repair strategy allows preemptions of repairs, then interrupted repairs are eventually resumed from the point of interruption and there are no switching delays.

The state of the system at time  $t$  can be described by a pair of integer valued random variables,  $I(t)$  and  $J(t)$  specifying the *processor configuration* (can also be termed, *operative state* of the multiprocessor system) and *the number of jobs present*, respectively. Here, the precise meaning of processor configuration, and hence the range of values of  $I(t)$ , mean the number of operational processors and associated reconfiguration/rebooting delay when appropriate.

In general, let's assume that there are  $N+1$  processor configurations, (operative states of the multiprocessor) represented by the values  $I(t) = 0, 1, \dots, N$ . These  $N+1$  configurations are the *operative states* of the model. The model assumptions are assumed to ensure that  $I(t), t \geq 0$ , is an irreducible Markov process.  $J(t)$  is the total number of jobs in the system at time  $t$ , including the ones in service. Then,  $Z = \{[I(t), J(t)]; t \geq 0\}$  is an irreducible Markov process on a lattice strip (a QBD process), that models the system. Its state space is,  $\{0, 1, \dots, N\} \times \{0, 1, \dots, L\}$ .

This system was analysed for exact performability (Chakka and Mitrani 1994; Chakka 1995), for single repairman ( $R=1$ ) and  $L \rightarrow \infty$  and for some repair strategies but reconfiguration and rebooting delays were not considered.

## MODELLING RECONFIGURATION AND REBOOTING DELAYS IN MULTIPROCESSOR SYSTEMS

In multiprocessor systems, in practice however, some delay is encountered when a failed processor is being mapped out of the system (reconfiguration/rebooting delay), and when a repaired processor is being admitted into the system. It is possible to model the system affected by such reconfiguration and rebooting delays effectively using the spectral expansion method.

Consider the homogeneous multiprocessor system with  $K$  processors, given in Figure 1.  $\mu$  and  $\xi$  are the service and failure rates of each of the processors. There is a single repair facility (i.e.  $R=1$ ) with repair rate  $\eta$ . When a processor fails the fault is covered with probability  $c$  and is not covered with probability  $1-c$ . Subsequent to a covered fault, the system comes up in a degraded mode after a brief reconfiguration delay, while after an uncovered fault a longer reboot action is required to bring the system up at a degraded mode. Here, degraded mode indicates a state with one less operative processor than the previous state. For reconfiguration/rebooting period, the system is assumed to be down.

The reconfiguration and rebooting times are exponentially distributed with mean  $1/\delta$  and  $1/\varphi$  respectively. The queuing capacity is  $L$ , where  $L$  can be finite or infinite.  $\sigma$  is the arrival rate of jobs.

An approximate performance modelling of this system was carried out in (Trivedi and Sathaye 1990). We intend to carry out an exact performance evaluation of this system. Figure 2 is the Markov chain that represents the operative states of the multiprocessor system.

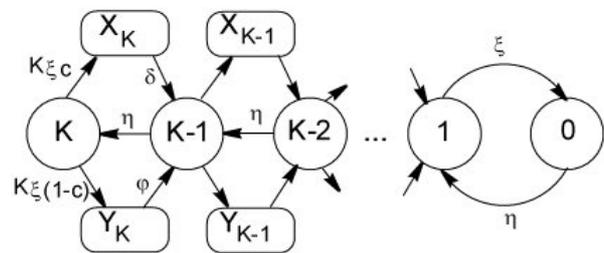


Figure 2: A Homogeneous Multiprocessor System with Breakdowns, Repairs, Rebooting and Reconfiguration Delays

The states labelled 1, 2, ...,  $K$  are the  $K$  working states of the multiprocessor, with that many number of processors in each state. State 0 means no processor is operational. The  $K-1$  states, labelled as  $X_2, X_3, \dots, X_K$ , are the states representing the reconfiguration delay.

The  $K-1$  states labelled as  $Y_2, Y_3, \dots, Y_K$ , are the rebooting delay states. Hence, the total number of operative states is  $3K-1$ . Let these be renumbered as, states  $0, 1, \dots, K$  unchanged, states  $X_2, X_3, \dots, X_K$  as  $K+1, K+2, \dots, 2K-1$ , and the states  $Y_2, Y_3, \dots, Y_K$  as  $2K, 2K+1, \dots, 3K-2$ .

The system now can be represented by a QBD process with finite or infinite state space. The state of the system can be defined by  $(I(t), J(t))$  where  $I(t)$  is the operative state and  $J(t)$  is the number of jobs in the system. Let the operative states be represented in the horizontal direction and the number of jobs in the vertical direction of a two-dimensional lattice strip. Here  $A$  is the matrix of instantaneous transition rates from operative state  $i$  to operative state  $k$  with zeros on the main diagonal. These are the purely lateral transitions of the model  $Z$ . Matrices  $B$  and  $C$  are transition matrices for one-step upward and one-step downward transitions respectively. When the transition rate matrices depend on  $j$  for  $j \geq M$ , where  $M$  is a threshold having an integer value, the process  $Z$  evolves with the following instantaneous transitions:

$A_j$ : Purely lateral transition rate, from state  $(i, j)$  to state  $(k, j)$ , ( $0 \leq (i \& k) \leq N$ ;  $i \neq k$ ;  $j=0, 1, \dots, L$ ), caused by a change in the operative state (i.e. a break-down followed by reconfiguration or rebooting, and a repair).

$B_j$ : One-step upward transition rate, from state  $(i, j)$  to state  $(k, j+1)$ , ( $0 \leq (i \& k) \leq N$ ;  $j=0, 1, \dots, L$ ), caused by a job arrival into the queue.

$C_j$ : One-step downward transition rate, from state  $(i, j)$  to state  $(k, j-1)$ , ( $0 \leq (i \& k) \leq N$ ;  $j=1, 2, \dots, L$ ), caused by the departure of a serviced job.

Clearly, the elements of  $A$  depend on the parameters  $K, \xi, \eta, c, \delta$  and  $\varphi$ . The state transition matrices  $A, A_j, B, B_j, C, C_j$ , can be given as follows;

$$A = \begin{bmatrix} 0 & \eta & 0 & 0 & 0 & 0 & 0 & 0 \\ \xi & 0 & \eta & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \eta & 2\xi c & 0 & 2\xi(1-c) & 0 \\ 0 & 0 & 0 & 0 & 0 & 3\xi c & 0 & 3\xi(1-c) \\ 0 & \delta & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \delta & 0 & 0 & 0 & 0 & 0 \\ 0 & \varphi & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \varphi & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (1)$$

In Equation (1)  $K=3$  is assumed. Since time-dependent failures are considered, the matrices  $A_j$  do not depend on  $j$ , and hence,  $A_j=A$  for all values of  $j$ . Similarly,

$$B_j = B, j=0, 1, \dots, L;$$

$$B = \text{Diag}[\sigma, \sigma, \dots, \sigma] \text{ of size } (3K-1) \times (3K-1).$$

$$C_j = C \text{ for } j \geq K;$$

$$C = \text{Diag}[w(0)\mu, w(1)\mu, \dots, w(3K-2)\mu];$$

$$C_0 = [\mathbf{0}];$$

$$C_j = \text{Diag}[\text{Min}\{w(0), j\}\mu, \text{Min}\{w(1), j\}\mu, \dots, \text{Min}\{w(3K-2), j\}\mu] \text{ for } 1 \leq j < K$$

where  $w(i)$  is the number of working processors in the operative state  $i$ .

We define the matrices  $Q(\lambda)$  and  $\bar{Q}(\beta)$  as before (Chakka 1998). Then, the steady state probabilities,  $p_{i,j}$ , can again be expressed in a similar manner as shown in (Chakka 1998). From this, the required performability measures such as the steady state probabilities, average number of jobs in the system, utilization of the processors, and mean response time can be obtained exactly following the computational procedure found in (Chakka 1998; Chakka 1995). Using the steady state probabilities, the response time distribution can also be derived.

### THE STEADY STATE SOLUTION

The solution is given for an unbounded queue (i.e.  $K \leq L < \infty$ ) as well as a bounded queue (i.e. finite  $L \geq K$ ).

Following the spectral expansion solution, the steady-state probabilities of the system considered can be expressed as:

$$p_{i,j} = \lim_{t \rightarrow \infty} P(I(t) = i, J(t) = j);$$

$$0 \leq i \leq N, \quad 0 \leq j \leq L$$

where  $L$  can be finite or infinite. Let's define diagonal matrices of size  $(N+1) \times (N+1)$  as:

$$D_j^A(i, i) = \sum_{k=0}^N A_j(i, k);$$

$$D_j^B(i, i) = \sum_{k=0}^N B_j(i, k);$$

$$D_j^C(i, i) = \sum_{k=0}^N C_j(i, k);$$

$$D^A(i, i) = \sum_{k=0}^N A(i, k);$$

$$D^B(i, i) = \sum_{k=0}^N B(i, k);$$

$$D^C(i, i) = \sum_{k=0}^N C(i, k);$$

and

$$Q_0 = B, \quad Q_1 = A - D^A - D^B - D^C, \quad Q_2 = C.$$

These matrices are used in the spectral expansion solution for both bounded and unbounded queuing systems.

## Unbounded Queuing System

For an unbounded queuing system, all state probabilities in a row (the row vectors,  $\mathbf{v}_j$ ) can be defined as:

$$\mathbf{v}_j = (p_{0,j}, p_{1,j}, \dots, p_{N,j}); \quad j = 0, 1, 2, \dots$$

The steady-state balance equations can now be written as:

$$\mathbf{v}_0 [D_0^A + D_0^B] = \mathbf{v}_0 A_0 + \mathbf{v}_1 C_1 \quad (2)$$

$$\mathbf{v}_j [D_j^A + D_j^B + D_j^C] = \mathbf{v}_{j-1} B_{j-1} + \mathbf{v}_j A_j + \mathbf{v}_{j+1} C_{j+1}; \quad 1 \leq j \leq M-1 \quad (3)$$

$$\mathbf{v}_j [D^A + D^B + D^C] = \mathbf{v}_{j-1} B + \mathbf{v}_j A + \mathbf{v}_{j+1} C \quad (4)$$

$j \geq M$

and the normalizing equation:

$$\sum_{j=0}^{\infty} \mathbf{v}_j \mathbf{e} = \sum_{j=0}^{\infty} \sum_{i=0}^N p_{i,j} = 1.0$$

from Equation (4) one can write

$$\mathbf{v}_j Q_0 + \mathbf{v}_{j+1} Q_1 + \mathbf{v}_{j+2} Q_2 = 0; \quad j \geq M-1$$

Furthermore, the *characteristic matrix polynomial*  $Q(\lambda)$  can be defined as:

$$Q(\lambda) = Q_0 + Q_1 \lambda + Q_2 \lambda^2$$

where

$$\psi Q(\lambda) = 0; \quad |Q(\lambda)| = 0.$$

$\lambda$  and  $\psi$  are eigenvalues and left-eigenvectors of  $Q(\lambda)$  respectively. Note that,  $\psi$  is a vector and

$$\begin{aligned} \psi &= \psi_0, \psi_1, \dots, \psi_N \\ \lambda &= \lambda_0, \lambda_1, \dots, \lambda_N \end{aligned}$$

Finally, for an unbounded system, and avoiding large numbers resulting from the positive powers of eigenvalues greater than 1.0, one can obtain the general solution as:

$$\mathbf{v}_j = \sum_{k=0}^N a_k \psi_k \lambda_k^{j-M+1}, \quad j \geq M-1$$

and in the state probability form,

$$p_{i,j} = \sum_{k=0}^N a_k \psi_k(i) \lambda_k^{j-M+1}, \quad j \geq M-1$$

where  $a_k$  ( $k=0, 1, \dots, N$ ) are arbitrary constants which can be scalar or complex. The remaining  $\mathbf{v}_j$  and  $a_k$

values can be obtained using an iterative process (Chakka 1995).

## Bounded Queuing System

Analyses presented for the unbounded queue apply to the bounded queue with  $0 \leq j \leq L$  with the balance equations given as follows:

$$\mathbf{v}_0 [D_0^A + D_0^B] = \mathbf{v}_0 A_0 + \mathbf{v}_1 C_1 \quad (5)$$

$$\mathbf{v}_j [D_j^A + D_j^B + D_j^C] = \mathbf{v}_{j-1} B_{j-1} + \mathbf{v}_j A_j + \mathbf{v}_{j+1} C_{j+1}; \quad 1 \leq j \leq M-1 \quad (6)$$

$$\mathbf{v}_j [D^A + D^B + D^C] = \mathbf{v}_{j-1} B + \mathbf{v}_j A + \mathbf{v}_{j+1} C \quad (7)$$

$M \leq j < L$

$$\mathbf{v}_L [D^A + D^C] = \mathbf{v}_{L-1} B + \mathbf{v}_L A \quad (8)$$

The normalizing equation is given as:

$$\sum_{j=0}^L \mathbf{v}_j \mathbf{e} = \sum_{j=0}^L \sum_{i=0}^N p_{i,j} = 1.0$$

From Equation (7)

$$\mathbf{v}_j Q_0 + \mathbf{v}_{j+1} Q_1 + \mathbf{v}_{j+2} Q_2 = 0; \quad (M-1) \leq j \leq (L-2)$$

and the *characteristic matrix polynomials* can be expressed as:

$$Q(\lambda) = Q_0 + Q_1 \lambda + Q_2 \lambda^2$$

$$\bar{Q}(\beta) = Q_2 + Q_1 \beta + Q_0 \beta^2$$

where

$$\psi Q(\lambda) = 0; \quad |Q(\lambda)| = 0$$

$$\phi \bar{Q}(\beta) = 0; \quad \left| \bar{Q}(\beta) \right| = 0$$

$\beta$  and  $\phi$  are eigenvalues and left-eigenvectors of  $\bar{Q}(\beta)$  respectively. Note that,  $\phi$  is a vector and

$$\begin{aligned} \phi &= \phi_0, \phi_1, \dots, \phi_N \\ \beta &= \beta_0, \beta_1, \dots, \beta_N \end{aligned}$$

Furthermore,

$$\mathbf{v}_j = \sum_{k=0}^N (a_k \psi_k \lambda_k^{j-M+1} + b_k \phi_k \beta_k^{L-j}), \quad M-1 \leq j \leq L$$

and in the state probability form,

$$p_{i,j} = \sum_{k=0}^N (a_k \Psi_k(i) \lambda_k^{j-M+1} + b_k \phi_k(i) \beta_k^{L-j}),$$

$$M-1 \leq j \leq L$$

where  $b_k$  ( $k=0,1, \dots, N$ ) are arbitrary constants which can be scalar or complex.

We have solved the balance equations for both cases (Equations (2) – (8)), using the spectral expansion method, computed the state probabilities and obtained the mean queue length as:

$$MQL = \sum_{j=0}^L j \sum_{i=0}^N p_{i,j}$$

where  $L$  can be finite or infinite depending on whether the case concerned is bounded or unbounded.

### NUMERICAL RESULTS

To show the effectiveness of the method presented and evaluate the performance of the proposed system, we first considered 1, 2, 3, and 4-processor systems with break-downs and an infinite queue. Other parameters are given as  $\sigma$  jobs/sec,  $\xi=0.01$ ,  $\eta=0.5$ ,  $\mu=4000$  jobs/hr,  $\varphi = 2$  /hr, and  $\delta = 60$  /hr unless stated otherwise.

Figure 3 shows the relationship between the mean queue length and the mean arrival rate  $\sigma$ , for different number of servers and  $c=0$  (a single-server system is independent of  $c$ ).

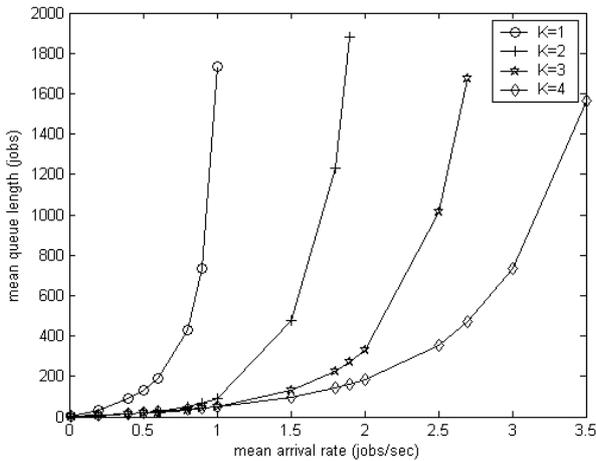


Figure 3: Mean Queue Length versus Mean Arrival Rate

Figure 4 shows the mean queue length as a function of  $c$ . Other parameters are  $\sigma = 20$  jobs/sec,  $\sigma/(K\mu)=0.7$ ,  $\xi=0.01$ ,  $\eta=0.5$ ,  $\varphi = 10$  /hr, and  $\delta = 50$  /hr. It is clearly evident that an increase in  $c$  results in a decrease in the mean queue length because reconfiguration delays are shorter than rebooting delays. Here, it is important to note that when  $\sigma/(K\mu)$  is kept constant, the 3-processor system considered performs better than the 4-processor system specified, especially as  $c$  increases. This is because failure rate is proportional to the number of

operational processors and in case of a single processor failure, the whole system goes down for a period of  $1/\delta$  or  $1/\varphi$  as appropriate.

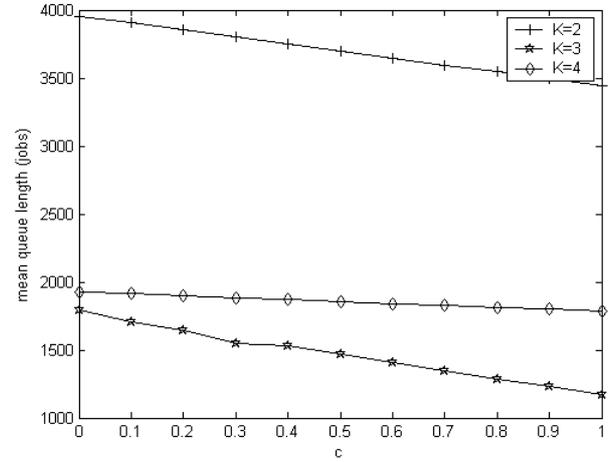


Figure 4: Mean Queue Length as a Function of  $c$  for 2, 3, and 4-Processor Systems

The parameters  $1/\delta$  or  $1/\varphi$  are certainly important in determining the system degradation due to reconfiguration/rebooting delays and identifying the optimum number of processors. We computed the mean queue length as a function of reconfiguration rate  $\delta$  for  $K=3$ .  $\sigma = 1$  job/sec,  $\sigma/(K\mu)=0.7$ ,  $\xi=0.01$ ,  $\eta=0.5$ , and  $\varphi = 2$  /hr. This is illustrated in Figure 5. The results indicate that as  $c$  increases the change in the mean queue length decreases considerably. This shows that  $\varphi$  plays an important role in degrading the system's performance.

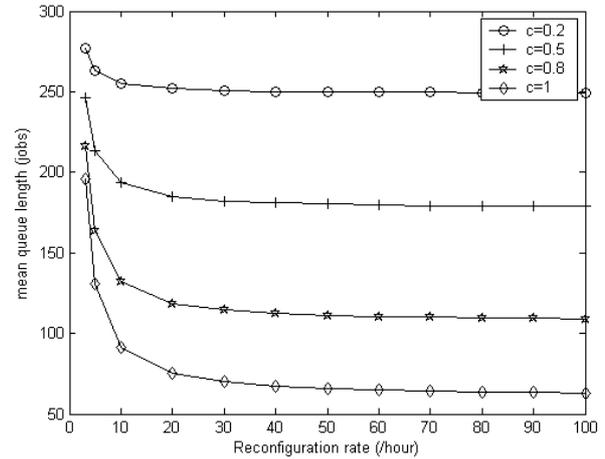


Figure 5: Mean Queue Length as a Function of  $\delta$  and  $c$

Figure 6 shows mean queue length as a function of  $c$ , with  $K = 1, 2$ , and 3,  $\sigma = 20$  jobs/sec, and  $\sigma/(K\mu) = 0.7$ . This shows that depending on the values of reconfiguration and rebooting delays, the mean queue length performance of a 2-processor system may become better than that of a 4-processor system for some  $c$  values and approaches to the performance of a 3-processor system. The performance of a single processor system does not depend on  $c$  and the

performance of such a system is shown for comparison purpose only.

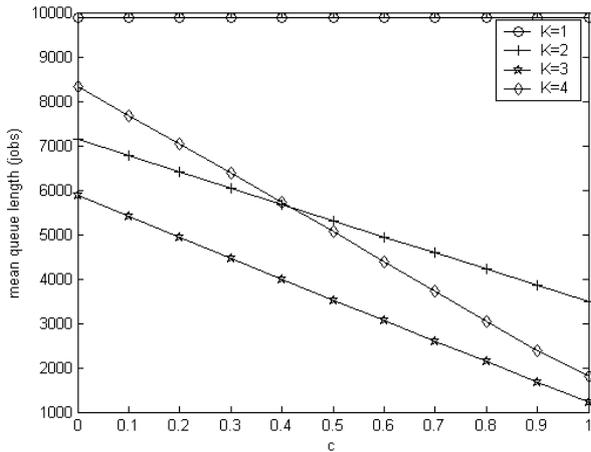


Figure 6: Mean Queue Length as a Function of  $c$

Figure 7 shows how mean queue length decreases as  $c$  increases for  $K=3$ ,  $\sigma = 1$  job/sec, and  $\sigma/(K\mu) = 0.7$ . The corresponding mean queue length values are presented for various values of  $\delta$ . Again, for larger reconfiguration delays (i.e.  $\delta$  small) performance degradation is evident even at higher  $c$  values. As the reconfiguration delay decreases (e.g.  $\delta > 20$ ) the degradation is mainly due to  $c$  values.

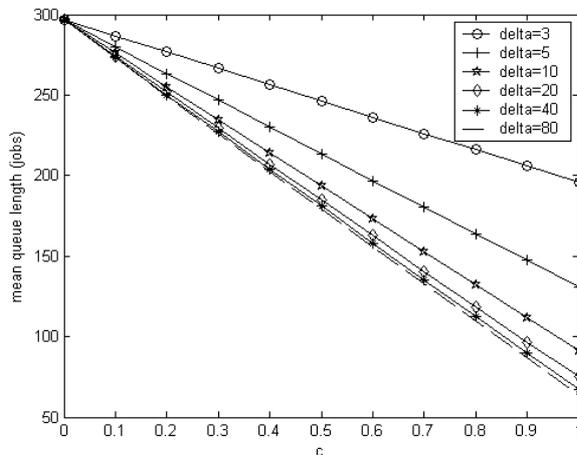


Figure 7: Mean Queue Length as a Function of  $c$  and  $\delta$

Figure 8 shows the probability that the system is idle, for various  $K$  and  $c$  values with  $\sigma = 20$  jobs/sec, and  $\sigma/(K\mu) = 0.7$ . Here, different  $c$  values did not seem to matter.

To demonstrate the effects of finite queuing capacity, mean queue length of a bounded system is calculated. First, Figure 6 has been reproduced for  $L = 100$ , and  $\sigma = 1$  job/sec. Figure 9 shows MQL as a function of  $c$  for finite  $L$ . All other parameters are same as the ones used for Figure 6. Clearly, when  $\sigma/(K\mu)=0.7$ ,  $L$  is the limiting factor, and  $K$  has a negligibly small effect. Similarly,

Figure 7 was reproduced for  $L = 300$ . Again, the limiting factor here is  $L$ . This is illustrated in Figure 10.

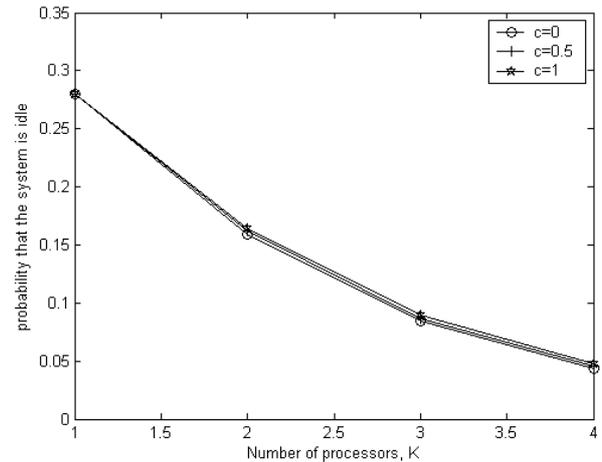


Figure 8:  $p_0$  as a Function of the Number of Servers with  $\sigma/(K\mu)=0.7$  and Various  $c$  Values

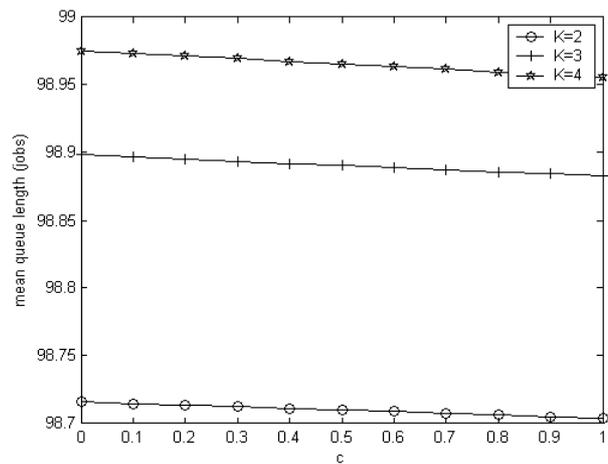


Figure 9: Mean Queue Length as a Function of  $c$  for 1, 2, 3, and 4-Processor Systems and  $L = 100$

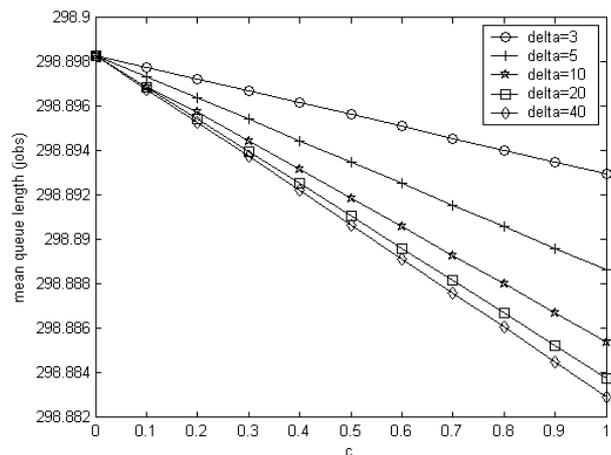


Figure 10: Mean Queue Length as a Function of  $c$  and  $\delta$  for  $L = 300$

Finally, the mean queue length and job loss rate are calculated for various  $\sigma$  values with  $K=3$  and 4,  $c=0$ , 0.5, and 1,  $\mu=2$  jobs/sec,  $\xi=0.01$ ,  $\eta=0.5$ ,  $1/\varphi = 500$  sec,

$1/\delta = 50$  sec, and  $L = 200$ . Figure 11 and Figure 12 illustrate the results obtained.

As it can be seen on Figure 11, the effect of  $c$  on mean queue length is significant for smaller  $\sigma$  values for  $K=3$  as well as  $K=4$ . This is due to the relatively low demand on resources. However, as  $\sigma$  increases,  $L$  becomes the main limiting factor in the performance of the system and the mean queue length approaches  $L$ .  $K=4$  will only be preferable over  $K=3$  if good cover is provided (i.e.  $c=1$ ) and  $\sigma$  is relatively large.

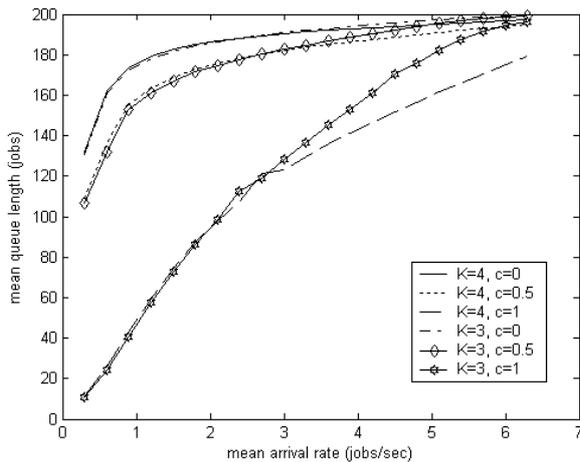


Figure 11: Mean Queue Length as a Function of  $K$ ,  $c$ , and  $\sigma$  for  $L = 200$

Figure 12 shows that, when the percentage of jobs lost is considered,  $c$  becomes an important parameter for all  $\sigma$  values. For larger  $\sigma$  values,  $L$  becomes an important limiting factor, especially for low  $c$  values.

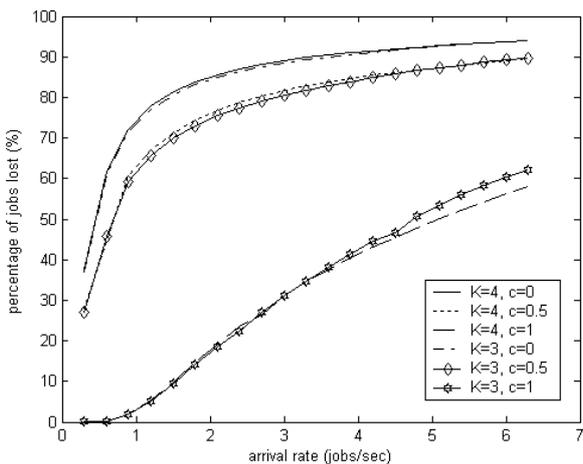


Figure 12: Percentage of Jobs Lost as a Function of  $K$ ,  $c$ , and  $\sigma$  for  $L = 200$

## CONCLUSIONS AND RECOMMENDATIONS

In this paper multiprocessor systems with break-downs, and, reconfiguration and rebooting delays have been modelled for exact solution. The state probabilities in

the case of a homogeneous multiprocessor system with breakdowns, repairs, reconfiguration and rebooting delays are derived using the spectral expansion method. Numerical results have been obtained and presented for various performability parameters, for both bounded and unbounded systems. Results show that, when queue limit is not an important factor on mean queue length performance, the choice of the optimum number of processors depends on the values of  $1/\delta$ , and  $1/\varphi$  as well as  $c$  and demonstrate the effect of these parameters on system performance. However, for bounded queuing systems,  $L$  is the main factor affecting the mean queue length performance of the system at relatively large  $\sigma$  values.

The method can be extended to the case of heterogeneous multiprocessor systems with non-identical servers and also to many of the high performance/highly available/highly reliable computer architectures. The performability of flexible manufacturing cells can also be modelled using the method presented and hence the model becomes highly relevant to manufacturing or production research.

## REFERENCES

- Buzacott, J. A. and J.G. Shanthikumar, 1993. *Stochastic Models of Manufacturing Systems*, Prentice-Hall.
- Chakka, R. and I. Mitrani, 1992. "A numerical solution method for multiprocessor systems with general breakdowns and repairs". In *Proceedings of the 6th International Conference Modelling Techniques and Tools*, (Edinburgh, UK), 289-304.
- Chakka, R. and I. Mitrani, 1994. "Heterogeneous multiprocessor systems with breakdowns: performance and optimal repair strategies", *Theoretical Computer Science*, 125, 91-109.
- Chakka, R., 1995. *Performance and Reliability Modelling of Computing Systems Using Spectral Expansion*, PhD Thesis, University of Newcastle Upon Tyne, UK.
- Chakka, R. 1998. "Spectral expansion solution for some finite capacity queues", *Annals of Operations Research*, 79, 27-44.
- Chakka, R., O. Gemikonakli and P. Basappa, 2002. "Modelling Multiprocessor Systems With Time or Operation Dependent Breakdowns, Alternate Repair Strategies, Reconfiguration and Rebooting Delays". In *Proceedings of SPECTS 2002*, (San Diego, CA, July).
- Fiems, D., B. Steyaert, and H. Bruneel, 2004. "Discrete-time queues with generally distributed service times and renewal type server interruptions", *Performance Evaluation*, 55, 277-298.
- Harrison, P. G. and N.M. Patel, 1993. *Performance Modelling of Communication Networks and Computer Architecture*, Addison-Wesley.
- Stecke, K. E. and I. Kim, 1989. "Performance evaluation for systems of pooled machines of unequal sizes: unbalancing vs. blocking". *European Journal of Operations Research* 42, 22-38.
- Stecke, K.E. 1992. "Machine interference: Assignment of machines to operators". *Handbook of Industrial Engineering, Second Edition*, Chapter 57, Gavriel Salvendy (Ed.).

Righter, R. 1996. "Optimal Policies for Scheduling Repairs and Allocating Heterogeneous Processors". *Journal of Applied Probability*, 33, 536-547.

Trivedi, K. S., 2002. *Probability and Statistics with Reliability, Queuing, and Computer Science Applications*, Wiley, NY, USA.

Trivedi, K. S. A.S. Sathaye, O.C. Ibe and R.C. Howe, 1990. "Should I add a processor?", In *Proceedings of 23rd Annual Hawaii International Conference on System Sciences*, IEEE Computer Society Press, (Los Alamos, CA), 214-221.

## AUTHOR BIOGRAPHIES



**ORHAN GEMIKONAKLI** was born in Yesilyurt, Cyprus, and went to Eastern Mediterranean University in Cyprus, where he studied electrical engineering and obtained his degree in 1984. He then continued his studies at King's College, University of London where he obtained his MSc in Digital Electronics, Computers and Communications and PhD in Telecommunications in 1985 and 1990 respectively. He worked there as a post-doctoral Research Associate for a couple of years before moving in 1990 to Middlesex University, London where he is now a Principal Lecturer leading postgraduate studies in Computer Communications. He is a member of the IEE, a member of the IEEE and a Chartered Engineer. His research interest is in network security and performance modelling. His e-mail address is: o.gemikonakli@mdx.ac.uk and his Web-page can be found at <http://www.cs.mdx.ac.uk/staffpages/orhan/>.



**TIEN VAN DO** obtained a PhD degree from the Budapest University of Technology and Economics in 1996. He is an associate professor at Department of Telecommunications, Budapest University of Technology and Economics and a leader of Networking Technology Laboratory. His research interests include network technology, performance evaluation and network design.



**RAM CHAKKA** received his B.Engg. (Electrical and Electronics, 1980), M.S.(Engg) (Computer Science, 1986) from the Indian Institute of Science, Bangalore, India and Ph.D. (Computer Science, 1995) from the University of Newcastle upon Tyne, UK. Presently he is a Professor and Head of the RDQ (Research, Development, Quality of education) unit at the RGM CET, Nandyal, India. Earlier, he worked at Indian Institute of Science, University of Newcastle upon Tyne, Imperial College (London), Middlesex University (UK), Norfolk State University (USA) and Sri Sathya Sai Institute of Higher Learning

(India). He published a number of papers in Performability Modelling.



**ENVER EVER** was born in Finsbury, London. He obtained his BSc. degree from the Department of Computer Engineering, Eastern Mediterranean University, Cyprus in 2002 and his MSc in Computer Networks from Middlesex University, London in 2003. Currently he is a PhD student at Middlesex University. His research is on mathematical modelling for performance evaluation of various multiserver systems. His research interest is in networks and performance modelling. His email address is: [e.ever@mdx.ac.uk](mailto:e.ever@mdx.ac.uk).

# RSIM x86: A COST-EFFECTIVE PERFORMANCE SIMULATOR

Ricardo Fernández and José M. García  
Departamento de Ingeniería y Tecnología de Computadores  
Universidad de Murcia, 30071-Murcia (Spain)  
E-mail: {r.fernandez,jmgarcia}@ditec.um.es

## KEYWORDS

High-performance simulators, multiprocessor architectures, computer architecture, large scale computing platforms.

## ABSTRACT

In this paper we present RSIM x86, a port of the widely used RSIM performance simulator for cc-NUMA multiprocessors to GNU/Linux and x86 hardware. Then, we evaluate the simulation throughput obtained by RSIM in several platforms with respect to the hardware cost of each platform. We show that this port of RSIM obtains much better execution times using cheaper and more easily available hardware than the original RSIM, allowing a more efficient usage of our research resources.

## INTRODUCTION

Doing research or system design in computer architecture involves deciding among many interrelated tradeoffs. Computer architecture is increasingly driven by quantitative data. Usually, developers can devise analytical models to bound the design space in the very early development stages but the interactions between many design decisions in today increasingly complex systems make impossible to use these analytical models to accurately predict the performance of a finished system. Hence, we need experimental models in order to guess the performance impact of a possible design decision before building a finished system.

Doing direct performance measurements requires a finished model, hence it is not possible to do it during the design phase. Also, building prototypes is too expensive for most research projects. As an alternative, system architects and researches use performance simulators to predict the effect of the ideas and techniques that they need to evaluate.

Performance simulators are complex software systems which accurately model the behavior of a hardware system. Doing a simulation of a hardware model is several orders of magnitude slower than running the simulated system. Developers need fast and accurate simulators to

be able to perform as many useful experiments as possible.

There are two main types of performance simulators for processors: trace driven and execution driven. Trace driven simulators use traces obtained from the real execution of programs to drive a performance model while execution driven simulators simulate the actual execution of a program recording detailed performance statistics. The current trend in performance simulation is to use execution driven simulation because it allows much more precise results specially for current processors which exploit instruction level parallelism using out of order execution and speculation.

There are several popular execution driven performance simulators and simulation frameworks like SimOS (Rosenblum et al. 1997), MASE (Larson et al. 2001), Winsconsin Wind Tunnel II (Mukherjee et al. 2000), SimpleScalar (Austin et al. 2002), Simics (Magnusson et al. 2002), Asim (Emer et al. 2002) or RSIM (Hughes et al. 2002).

RSIM (Hughes et al. 2002; Pai et al. 1997a) is a simulator primarily targeted to study shared-memory cache coherent (cc-NUMA) multiprocessor architectures built from processors that aggressively exploit instruction-level parallelism (ILP).

RSIM key advantage is that it models a system comprised by several out-of-order processors which aggressively exploit instruction level parallelism (ILP). The model includes an aggressive memory system and a scalable interconnection network. Using detailed ILP models for the simulated processors provides a realistic approximation to modern and future multiprocessor systems. RSIM provides a great flexibility which allows using it to simulate a range of systems from monoproductors to different cc-NUMA configurations.

The accurate and flexible model provided by RSIM implies a slower execution speed than other less detailed simulators. Furthermore, although RSIM is supposed to be portable it was not available on common and cheap architectures like Linux/x86, requiring instead Solaris/SPARC, IRIX/MIPS or other big-endian machines. This has proved to be a serious problem to our research group due to the limited access to these kind of machines.

In this work we show how we ported RSIM to Linux/x86 and how that allows us to obtain an increased performance for our simulations at a fraction of the original cost.

In the next section we examine some other performance simulators available, specially those derived from RSIM. Later, we explain some key characteristics of RSIM and the approach we have followed to porting RSIM to Linux/i386. After that section, we evaluate the performance of the ported simulator with respect to the cost of the hardware used to run the simulations. And, finally, in the last section, we present our conclusions from this work.

## RELATED WORK

Some performance simulators already mentioned, like Simics or SimpleScalar, already have functional Linux/x86 ports. However, none of them is suitable for our purposes due to the lack of detail of the simulation and the different system architectures that they model.

Simics is a full system functional simulator which attempts to strike a balance between accuracy and performance. The goal of Simics is to allow the simulation of realistic workloads running unmodified operating systems and applications. Simics supports several simulated architectures including x86, UltraSPARC-II, PowerPC and others. It includes accurate device models to simulate I/O intensive applications common in commercial workloads. However, the big performance that it needs and its genericity prevents a detailed enough simulation necessary for many of our tasks.

SimpleScalar is a simulation toolset which provides an infrastructure for simulation and architectural modeling. The toolset can model a variety of platforms ranging from simple unpipelined processors to detailed dynamically scheduled microarchitectures with multiple-level memory systems. However, it simulates only uniprocessor systems.

RSIM has been used by many research groups since its publication and continues to be used nowadays. However, up to our knowledge, no one has published a functional x86 port until now. The original RSIM authors reported initial efforts in this direction, but their x86 port was never published, if it was ever completed.

Schaelicke and Parker ML-RSIM (Schaelicke and Parker 2002) is a derivative of URSIM (Zhang 2001) which is based on the original RSIM. It models the entire Input/Output subsystem and includes a functional operative system kernel called *Lamix* which is System V compatible. ML-RSIM has been ported to Linux/x86 architectures.

Unfortunately, ML-RSIM differs significantly from RSIM and does not model a cc-NUMA architecture, which makes it unfit for our research.

## PROBLEMS PORTING RSIM

RSIM is an interpreter for Solaris/SPARC v9 application executables. Internally, RSIM is a discrete event-driven simulator based on the YACSIM (Yet Another C Simulator) library from the Rice Parallel Processing Testbed (RPPT) (Convington et al. 1991; Pai et al. 1997b).

RSIM interprets application executables rather than uses traces, enabling more accurate modeling of the effects of contention and synchronization in multiprocessor simulations as well as speculation in multiprocessor and uniprocessor simulations. For speed, it converts the SPARC v9 instructions into an expanded, loosely encoded instruction set format and internally caches them.

RSIM subsystems include the processor engine, the memory module, the cache module, the directory module and the interconnection network. Each subsystem is mostly independent from each other and they interact through a small number of predefined interfaces.

RSIM is written in a modular fashion using C++ and C for extensibility and portability. Initially, it was developed using Sun systems (Solaris 2.5) on SPARC. It has successfully ported to HP-UX 10 running on a Convex Exemplar and to IRIX running on MIPS. However, porting it to 64-bit or little-endian architectures requires significant additional effort.

We have successfully ported RSIM to GNU/Linux running on x86 architectures. The main problems that we have had to solve were:

- Build issues due to differences in libraries and headers between Solaris and Linux.
- Byte Ordering Issues.
- System call interface differences.
- Floating point incompatibilities.

RSIM was developed using big-endian machines and simulates a big-endian architecture. This configuration is straightforward, but on the other hand our port needs to simulate a big-endian architecture on top of a little-endian machine. This implies that at some places we have to change the order of bytes in words.

We keep the simulated memory always unmodified in the target endianness, so that unaligned accesses or packed arrays are easy to handle. Basically, we swap bytes when performing each simulated memory operation.

The predecoded executable data is generated at benchmark build time. Therefore, it is in the same endianness than the original executable. We swap its bytes after loading it for simulation and cache the byteswapped version.

The endianness differences also subtly affect other parts of the simulator, like partially overlapped forwarding of memory operations and some initialization routines. Some of these cases have been detected only after careful debugging.

Linux and Solaris system call interfaces are not exactly the same, although they are very similar since they are both based on System V. Hence, some simulator traps that rely on host system calls require a translation of its parameters before they can be executed.

The floating point incompatibilities between SPARC and i386 are caused by the fact that SPARC implements the IEEE 754 floating point standard while i386 processors use the Intel x87 80-bit format for representing floating point numbers in the processor registers, even if they are stored in IEEE 754 format in memory.

In most cases, the extra precision is beneficial, but not for our purposes because it causes different results due to rounding differences. We wish to obtain the exact same results independently of the underlying architecture to be able to compare with our previous results, and the rounding differences would make this impossible.

To solve this, we instruct the compiler to produce floating point code using SSE2 instructions and registers present in newer x86 processors (Intel Corporation 2004). These instructions use eight 128-bit registers which can hold two double precision (64-bit) or four single precision (32-bit) numbers using the IEEE 754 format and perform short vectorial operations with them or scalar operations.

Another problem related with floating point differences is the different bit representation of the “Not a Number” (NaN) value in SPARC and i386 using SSE. Both representations are correct according to the IEEE 754 standard, but a benchmark could behave slightly differently if it tried to interpret the in memory representation of a NaN value.

Our port can optionally normalize the NaN representation ensuring that the SPARC representation is always used (this normalization process may affect simulation performance for our port. This option was enabled in the simulations used to measure times for this work). Most benchmark results are unaffected by this difference, but it makes debugging the port harder due to the different values stored in the simulated registers and memory.

Finally, the functions used to implement the simulated SPARC instructions to change the rounding modes are different between Solaris and Linux.

We have ensured that our port obtains exactly the same simulation results in both Solaris/SPARC and Linux/i386 architectures. This allows us to use all our machines to perform simulations and, more importantly, meaningfully compare the results of benchmarks and compare new results with old results from experiments performed prior to the port.

During the development of the port, we extended RSIM to produce extensive trace information detailing the content of every register and the instructions being executed at every moment in an easy to parse and compare format. This tracing support has proved useful beyond its initial purpose to diagnose problems during the development of new experiments.

## EVALUATION

The purpose of our evaluation is to check if using the ported version of RSIM on off-the-shelf x86 machines is a cost-effective solution to perform the great number of long running simulations needed for our research.

Firstly, we compare the execution speed of RSIM running in several different architectures. Secondly, since the execution time of a single benchmark is not the most valuable metric for our purposes, we define a better indicator of the usefulness of each simulation platform and version of RSIM.

Usually, simulations are run in batches using a queue management system like Condor (Thain et al. 2004). We will measure which version allows us to utilize our computing resources more efficiently in terms of hardware cost and execution time. We will use a metric based in the normalized number of simulations per hour per thousand euros.

We have measured the impact that the actual benchmark being simulated has in the speedup obtained by our port and have found that it is small, but not inexistent. Hence, we have chosen a small set of representative benchmarks from the SPLASH suite to perform our experiments. The chosen benchmarks are:

*fft*: Complex unidimensional version of the radix- $\sqrt{n}$  six-step FFT algorithm optimized to minimize interprocessor communication. It is one of the fastest benchmarks with less memory requirements.

*em3d*: Models the propagation of electromagnetic waves through objects in three dimensions.

*ocean*: Studies large-scale ocean movements based on eddy and boundary currents.

We have also measured the impact that varying the problem size of the simulated benchmarks has in the achieved

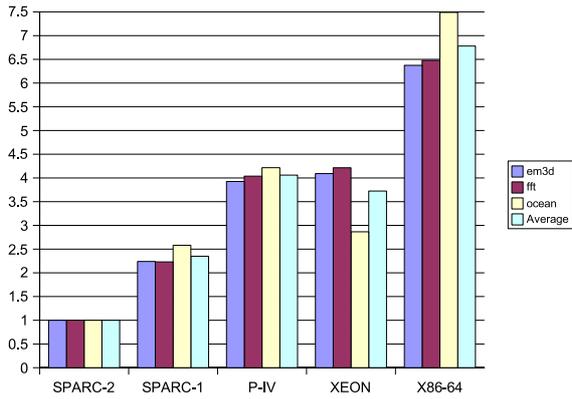


Figure 1: Normalized Throughput Per Processor for Each Architecture

speedup and have found that it is very small once a certain threshold is reached. Other simulator parameters, like the number of processors, have a very small influence too.

We have chosen medium problem sizes and we use two processors and default values for the rest of the parameters for our experiments.

We have evaluated the speed of running our port of RSIM in the following machines:

- A high-end Solaris/SPARC Sunblade-2000 system: *SPARC-1*.
- A low-end Solaris/SPARC Sunblade-100 system: *SPARC-2*.
- A high-end Linux/Athlon64 SMP system (running in legacy IA-32 mode): *X86-64*.
- A high-end Linux/Xeon SMP system: *XEON*.
- A low-end Linux/Pentium-IV system: *P-IV*.

The relevant characteristics and price of each machine is shown in table 1. When testing both SPARC machines, we have used a version of RSIM targeted for that platform. The prices indicated for the machines are necessarily approximate. These are the approximate prices that those systems would cost as of January 2005 in Spain.

In figure 1 we show the normalized throughput time of our set of benchmarks for each architecture. In other words, we show the throughput speedup of each machine compared with the slowest one (*SPARC-2*). For now, we only use one processor per machine even on those machines which have two processors, to allow a fair comparison between them.

We see that Linux/x86 based machines outperform the more expensive Solaris/SPARC based ones in raw simulation speed per processor. The fastest platform is

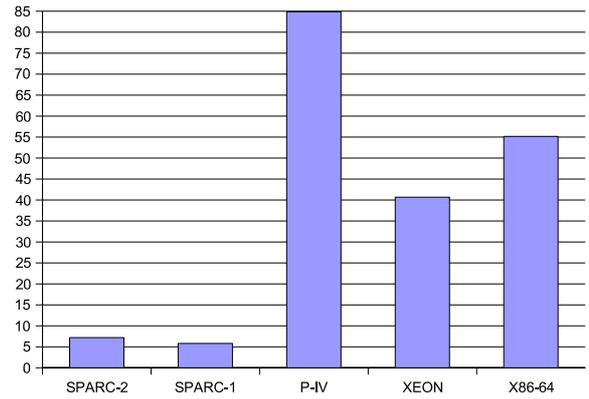


Figure 2: Average Number of Simulations Per Hour Per Thousand Euro

the Linux/Athlon64 based machine, which is 81% faster than the Xeon based machine and 67% faster than the Pentium-IV based machine (despite the much higher clock frequency of the Pentium-IV).

Some of our machines are SMP systems with two processors. In those cases, we can run two instances of RSIM simultaneously effectively doubling the throughput. Since the simulation work is CPU limited with very little IO and modest memory requirements there is no contention between the two independent processes.

In figure 2 we show the average number of simulations per hour per thousand euros achieved for each platform. When we account for the price of each machine, we see that the cheapest platform, the *Pentium-IV*, is the best option for efficiently take advantage of any given budget. Also, the easy availability of these kind of machines make them an even more attractive alternative to the expensive Solaris/SPARC machines used until now to run simulations based on RSIM. The two Solaris/SPARC based machines are much less cost-effective than the other platforms despite their higher prices.

## CONCLUSIONS

The purpose of our port of RSIM is to allow us to use our research resources more efficiently. Prior to the port, the small number of available machines to develop and run our simulations created long waiting queues and serious organizational problems.

Using a RSIM version which runs on cheap and readily available x86 hardware allows us to provide each researcher with its own workstation to comfortably develop and test his experiments and use an inexpensive cluster of Linux/x86 machines to execute the longest simulations. The x86 version not only executes each benchmark faster, but more importantly, it is easier to provide more resources to increase the throughput of the whole team.

Table 1: Characteristics of Evaluated Configurations

	SPARC-1	SPARC-2	X86-64	XEON	P-IV
Processor	UltraSPARC-III	UltraSPARC-III	AMD Opteron	Intel Xeon	Intel Pentium-IV
No. of processors	1	1	2	2	1
Frequency	1015 MHz	650 MHz	1791 MHz	2 GHz	3 GHz
RAM Memory	2GB	256 MB	1GB	1GB	1GB
L2 Cache	8MB	512 KB	1024 MB	512 KB	1024 KB
Price	5000 €	1800 €	3000 €	2600 €	600 €

The work required to port RSIM to x86 resulted greater than initially expected. RSIM code makes too many assumptions about low level details that are not portable and debugging problems caused by the different endianness proved to be difficult and time-consuming.

The code resulting from the port still has many portability problems that prevent its use in 64-bit architectures, like AMD x86-64 in native mode.

We are looking forward into making RSIM code more portable. In particular, we will make it 64-bit clean so it can be compiled natively for x86-64 architectures.

## AVAILABILITY

The source of our port is publicly available at:  
<http://www.ditec.um.es/gacop/tools/RSIM-x86>

## ACKNOWLEDGEMENTS

This work has been supported by the Spanish Ministry of Ciencia y Tecnología and the European Union (Feder Funds) under grant TIC2003-08154-C06-03, and by fellowship 01090/FPI/04 from the Comunidad Autónoma de la Región de Murcia (Fundación Séneca, Agencia Regional de Ciencia y Tecnología).

## REFERENCES

- Austin, T.; E. Larson and D. Ernst. February 2002. "SimpleScalar: An infrastructure for computer system modeling". *IEEE Computer* 35(2), 59–67.
- Convington, R. G.; S. Dwarkadas; J. R. Jump; J. B. Sinclair and S. Madala. 1991. "Efficient simulation of parallel computer systems". *International Journal in Computer Simulation* 1(1).
- Emer, J.; P. Ahuja; E. Borch; A. Klauser; C.-K. Luk; S. Manne; S. S. Mukherjee; H. Patil; S. Wallace; N. Binkert; R. Espasa and T. Juan. February 2002. "Asim: A performance model framework". *IEEE Computer* 35(2), 68–76.
- Hughes, C.; V. Pai; P. Ranganathan and S. Adve. February 2002. "RSIM: Simulating shared-memory multiprocessors with ILP processors". *IEEE Computer* 35(2), 40–49.
- Intel Corporation. 2004. *IA-32 Intel architecture software developer's manual*. Intel Corporation. Available at <http://developer.intel.com/>.
- Larson, E.; S. Chatterjee and T. Austin. 2001. "MASE: A novel infrastructure for detailed microarchitectural modeling". In *Proceedings of the IEEE International Symposium on Performance Analysis of Systems and Software*, pp. 1–9.
- Magnusson, P. S.; M. Christensson; J. Eskilson; D. Forsgren; G. Hållberg; J. Högberg; F. Larsson; A. Moestedt and B. Werner. February 2002. "Simics: A full system simulation platform". *IEEE Computer* 35(2), 50–58.
- Mukherjee, S. S.; S. K. Reinhardt; B. Falsafi; M. Litzkow; M. D. Hill; D. A. Wood; S. Huss-Lederman and J. R. Larus. October 2000. "Wisconsin wind tunnel II: A fast, portable parallel architecture simulator". *IEEE Concurrency* 8(4), 12–20.
- Pai, V. S.; P. Ranganathan and S. V. Adve. 1997a. "RSIM: An Execution-Driven Simulator for ILP-Based Shared-Memory Multiprocessors and Uniprocessors". In *Proceedings of the Third Workshop on Computer Architecture Education*. Also appears in IEEE TCCA Newsletter, October 1997.
- Pai, V. S.; P. Ranganathan and S. V. Adve. 1997b. "RSIM reference manual. version 1.0". Technical Report 9705, Electrical and Computer Engineering Department, Rice University.
- Rosenblum, M.; E. Bugnion; S. Devine and S. Herrod. January 1997. "Using the SimOS machine simulator to study complex computer systems". *ACM Transactions on Modeling and Computer Simulation* 7(1), 78–103.
- Schaelicke, L. and M. Parker. 2002. "ML-RSIM reference manual". Technical Report 02-10, Department of Computer Science and Engineering, University of Notre Dame.

Thain, D.; T. Tannenbaum and M. Livny. 2004. "Distributed computing in practice: The condor experience". *Concurrency and Computation: Practice and Experience* 17, 323–356.

Zhang, L.. 2001. "URSIM reference manual". Technical Report UUCS-00-015, University of Utah.

## AUTHOR BIOGRAPHIES



**RICARDO FERNÁNDEZ PASCUAL** was born in Madrid, Spain and went to the University of Murcia (Spain), where he studied Ingeniería Informática and obtained his master degree in 2004. He is currently researcher in the Computer Engineering Department at the Universidad de Murcia. His main research interests are cc-NUMA architectures, fault tolerance and performance simulation. His e-mail address is [r.fernandez@ditec.um.es](mailto:r.fernandez@ditec.um.es).



**JOSÉ M. GARCÍA** received the MS and the PhD degrees in electrical engineering from the Technical University of Valencia (Spain), in 1987 and 1991, respectively. Currently, Dr. García is a professor in the Computer Engineering Department at the Universidad de Murcia (Spain), and also the head of the research group on parallel computing architecture. He specializes in computer architecture, parallel processing, and interconnection networks. Dr. García served as vice-dean of the School of Computer Science from 1995 to 1997, and also as director of the Computer Engineering Department from 1998 to 2004. His current research interests lie in high-performance coherence protocols for shared-memory multiprocessor systems, and high-speed interconnection networks. Dr. García is member of several international associations such as the IEEE and ACM, and also member of some European associations (Euromicro and ATI). His e-mail address is [jmgarcia@ditec.um.es](mailto:jmgarcia@ditec.um.es).

# A VERSATILE SIMULATOR FOR CACHE MEMORIES ON DSM SYSTEMS

Miguel A. Vega-Rodríguez, R. Jorge Gil-Ramos, Juan A. Gómez-Pulido, Juan M. Sánchez-Pérez  
*Univ. Extremadura. Dept. Informatica*  
*Escuela Politecnica. Campus Universitario s/n. 10071 Caceres. SPAIN*  
*E-mail: mavega@unex.es*

## ABSTRACT

*In this work we present a simulator for the analysis of cache memories on scalable systems with distributed shared memory (DSM). In this kind of systems, the cache coherence is usually provided by means of directory-based protocols. The simulator, called DSMCache, has a full graphic and user-friendly interface, and it operates on PC systems with Windows. We think this tool is useful for the analysis of programs and design strategies of memory systems on DSM architectures. Thus, the simulator could be used in order to understand the design of organizations that run optimally a determinate type of parallel programs or to improve the operating mode of a concrete parallel architecture. Furthermore, due to its interface, it is an interesting tool for the teaching of cache memories on DSM systems.*

## 1. INTRODUCTION

Caches are a critical component in the performance of any computer system (part of the existent bibliography about caches can be found in (Smith 1986)).

A very important and promising part of the evolution of parallel architectures is the inclusion of cache coherence in a scalable machine with distributed shared memory (Culler et al. 1999). Most of these systems use directory-based coherence protocols and a scalable interconnection network. In this way, it is possible to build massively parallel processing systems (MPPs). In fact, DSM architectures are suitable for both commercial and scientific applications. Especially, cache coherent non-uniform memory access architectures (CC-NUMA), which can support both easy programming (due to their shared address space) and scalability (with a great number of processors) (Chung et al. 2001).

Trace-driven simulation is often a cost-effective method to estimate the performance of computer system designs. Above all when designing caches, translation-lookaside buffers (TLBs), or paging systems, trace-driven simulation is a very popular way to study and evaluate computer architectures, obtaining an acceptable estimation of performance before a system is built.

In this paper we present a trace-driven simulator for cache memories on scalable systems with distributed shared memory. There are well-known cache memory simulators (see (Uhlig and Mudge 1997) for a detailed study about trace-driven simulators): TYCHO (Gee et al. 1993; Hill and Smith 1989), DINERO (Edler and Hill 2005; Hill 1985), ACS (Acme Cache Simulator) (Hunt 1997), SISMEC (Gómez et al. 1996; Gómez 2005), CVT (Deijl et al. 1997), bigDIRN (Agarwal 2004), SMPCache (Vega et al. 2001; Vega 2005),... Using these simulators as a point of departure we have developed a new simulator, where new considerations have been taken into account regarding its capacity for working on multiprocessor environments, its analysis potentiality (variability in the simulation process, obtained data and their format, etc.), its interface and portability. The design considerations have been oriented to satisfy a set of requirements with didactic and research goals.

In the following section we mention the different hardware architectures supported by the simulator. In section 3 we explain both its interface and its functionality. Section 4 details the memory trace formats that can be used with the simulator. Then, in section 5, we display the practical results that the simulator has obtained, and finally, the conclusions are presented in the last section. The theoretical considerations concerning cache memory systems, and particularly regarding their use in multiprocessor environments, are widely discussed in many computer architecture texts (Culler et al 1999; Sima et al. 1998; Hennessy and Patterson 2003; Hwang 1993; Stallings 2003), and we will not mention them here. All operations and algorithms we use are similar to those found in these computer architecture texts. As a consequence, the results obtained with the simulator are very close to the real world.

## 2. SUPPORTED HARDWARE ARCHITECTURES

As for the hardware architecture, the simulator offers the possibilities summarized in table 1. Furthermore, it has been designed so that it can be easily extended (for example, adding a new coherence protocol).

Table 1: Architectural characteristics supported by the simulator

Number of processors	1, 2, 4, 8, 16, 32, 64 or 128
Directory protocols	SGI Origin (Sequent NUMA-Q in project)
Snoopy protocols	MSI or MESI (for used directory protocols)
Word wide (bits)	8, 16, 32 or 64
Words by block	1, 2, 4, 8, 16, 32, 64, 128, 256, 512 or 1024
Blocks in main memory	1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048, 4096, 8192, 16384, 32768, 65536, 131072, 262144, 524288, 1048576, 2097152 or 4194304
Blocks in cache	1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024 or 2048
Mapping	Direct, Set-Associative or Fully-Associative
Cache sets (for set-associative caches)	1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024 or 2048
Replacement policies	Random, LRU, FIFO or LFU
Writing strategies	Write-Back (for coherence protocols)
Cache levels in the memory hierarchy	1
References	To memory words
Maximum block size	8 KB
Maximum main memory size	32 GB
Maximum cache size (excluded labels, block state bits, counts, etc.)	16 MB

All these configuration parameters are related among themselves according to the theoretical models. If the user makes a choice which contradicts other parameters, the simulator warns him/her, and blocks the choice.

The user selects different choices in the simulator (by interactive windows) to configure a given architecture that may be stored on an ASCII data file for future loading, so the need to make many selections for configuring the same memory model is avoided. What is more, it is possible to set a default initial configuration for the simulator. These characteristics allow us to build a database with different memory structures, emulating architectures like Silicon Graphics, Sequent Computer Systems, and others.

### 3. SIMULATOR INTERFACE AND FUNCTIONALITY

DSMCache is a software tool for the evaluation of hierarchical memory systems on scalable architectures with distributed shared memory. This simulator operates on PC systems with Windows, and it has been written in a visual language. The simulator offers a Windows typical graphic interface, having a very complete contextual help system. Figure 1 shows an overall view of its graphic interface. This interface can be modified by the user: with or without XP style, with or without menu shadows, with or without certain bars and panels, etc.

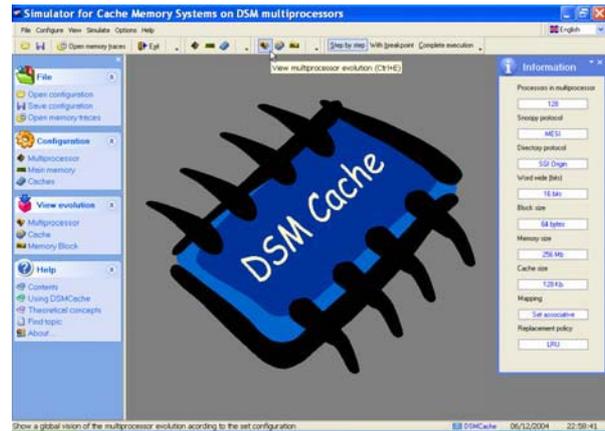


Figure 1: Graphic interface of the simulator

The simulator allows us to select the different choices for configuring a given architecture, and it shows us how the system responds to the memory accesses that the programs generate (memory traces used for the different processors during the simulation). Therefore, it is an application that could be used in order to evaluate memory systems on DSM architectures with research goals.

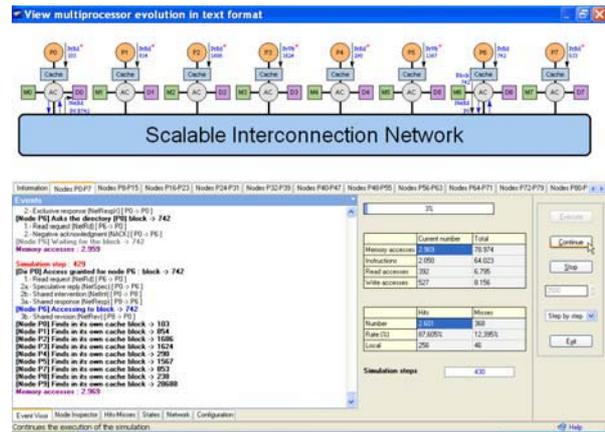


Figure 2: Multiprocessor vision during a simulation. Note that, in this case, at the top we can see the tab with the nodes P0-P7

Because of its easy and user-friendly interface, the simulator can also be used for teaching purposes; since it allows us to observe, in a clear and graphic way, how the complete system evolves as the execution of the programs goes forward (the memory traces are read). With the simulator, it is possible to obtain a global vision of the multiprocessor evolution, a vision of the evolution of a particular cache, or even, a vision of the evolution of a specific memory block. It always displays the memory accesses that every processor demands, the state of the interconnection network (network messages or transactions), of every communication assist, of every cache, of every directory, of every memory block within every cache, etc. Figure 2 shows the appearance of the screen in a simulation. At the top we can see a graphic representation of the multiprocessor: After a tab with generic information, the rest of tabs present the multiprocessor, 8 by 8 processors (P0-P7, P8-P15,...).

At the bottom there are also several tabs. The first one includes the event visor (see figure 2), in which the most important events are detailed after each simulation step. Another tab of great interest is the node inspector (figure 3), in which we can inspect (at any simulation step) the state of each node, including its processor, cache, directory, communication assist, and local main memory.

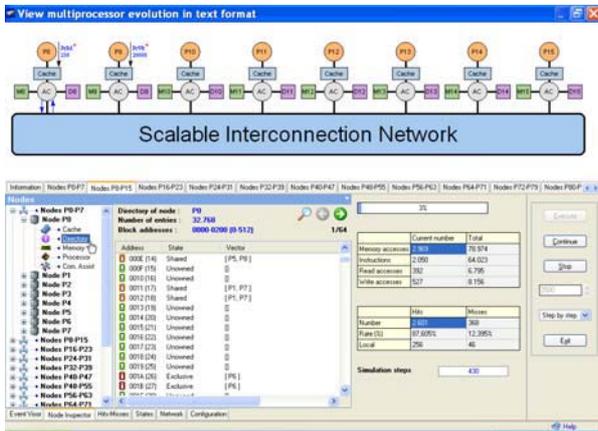


Figure 3: The node inspector during a simulation. Note that, now, at the top we can see the tab with the nodes P8-P15

The simulator also allows us to study the most suitable memory system for our needs before actually implementing it, or it simply helps us to simulate real systems in order to see their efficiency and compare results in an easy way. Some of the parameters we can study with the simulator are: program locality; influence of the number of processors, directory protocols, snoopy protocols, mapping, replacement policies, cache size (blocks in cache), number of cache sets (for set-associative caches), number of words by block (block size), word wide,...

Furthermore, DSMCache presents, using statistical data (in figures 2 and 3, tables in the bottom right hand corner, besides Hits-Misses, States and Network tabs at the bottom) and several kinds of graphics (see figure 4), interesting measurements like:

- Global number of network transactions/messages, and for types: read requests, read-exclusive requests, upgrade requests, invalidation requests, shared responses, exclusive responses, speculative replies, shared interventions, exclusive interventions, revisions, acknowledgments, NACKs, etc.
- Number of remote network messages, and number of local messages (inside the node).
- Total number of block transfers, and also distinguishing between block transfers through the interconnection network (remote block transfers) and inside the node (local block transfers).

- Network traffic taking into account the previous measurements.
- Total number of replacements, and number of replacements of local (memory blocks that are allocated in the local main memory) and remote blocks.
- Total number of write-backs, and number of write-backs by each node.
- Number of state transitions in cache (each block in a cache has a state associated with it) and directory (each memory block also has a directory state).
- Number of state transitions (in cache or directory) from a particular state to other.
- Global number of memory accesses, and for types: instruction captures, data readings and data writings.
- Number of remote accesses (to a remote block) and local accesses (to a local block) using the network.
- Number of global and local cache hits and misses, as well as the hit and miss rate.
- Distribution of the hits and misses (and their rates) for readings and writings: distinguishing between remote and local data.
- Number and rate of misses satisfied by the local main memory, by a remote cache or by a remote main memory.
- Number of simulation steps.

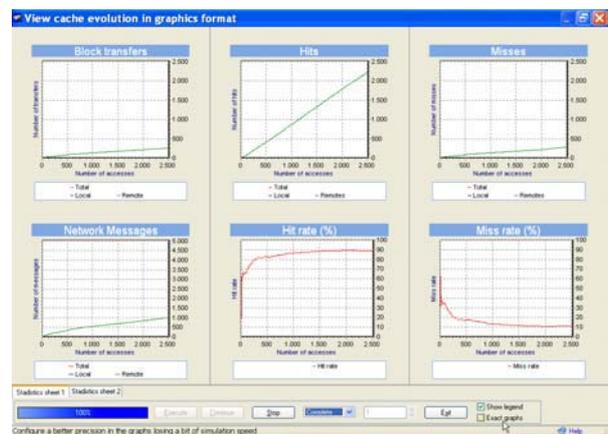


Figure 4: Data for a specific cache during a simulation in graphic format

All these data are shown at very different vision levels, although the relationship among all the system elements is taken into account in all cases. We can consequently carry out a simulation observing the

complete multiprocessor, and all the memory blocks (figure 2) or only one particular block. We can also observe a specific cache, and all the memory blocks or only a concrete block (figure 5).

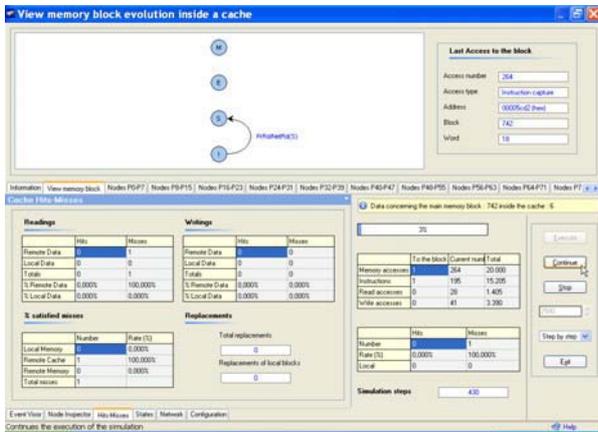


Figure 5: State transition diagram for a cache block

The simulation consists in the programmed reproduction of operations that would actually be performed by the components of cache memory system on a real multiprocessor. Suitable computations are performed to achieve that goal, and the present and accumulated results are shown at the same time. The simulation can be carried out as a whole (complete execution) or, as is usually much more interesting, step by step, in order to observe the internal operation of the system. For very long traces, breakpoints can also be inserted. There are therefore three kinds of simulation (step by step, with breakpoint and complete execution), and it is possible to change from one to another without waiting for the end of the simulation. It is also possible to abort the simulation at any time, in order to correct any architectural detail.

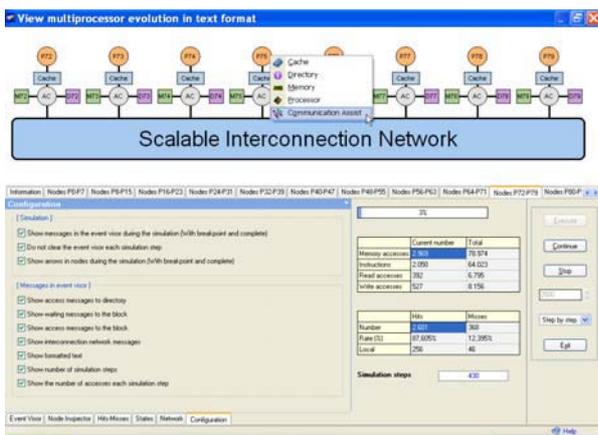


Figure 6: Contextual menu to obtain data of the different elements in the system, and Configuration tab for configuring the information and messages shown in a simulation. Note that, at the top, we can see the tab with the nodes P72-P79

Another aspect to highlight is the possibility of changing the information and messages shown in a simulation using the Configuration tab (see figure 6).

Figure 6 also includes an example of use of the contextual menus to obtain information about a particular system element (cache, directory, communication assist,... of certain node).

Finally, it is important to point out that, at present, the user can select between two languages for the simulator: English or Spanish.

#### 4. ADMITTED MEMORY TRACE FORMATS

In order to check and use the simulator we have a set of memory traces. Many of these traces come from tests performed with benchmarks and with application programs on different real (multiprocessor and uniprocessor) architectures. Some have been obtained by anonymous ftp from different trace databases (for example, from the PARL (PARL 2004)). Others have been created and edited using a common text editor. In this way, we can create, for example, memory trace files for teaching purposes (with some special feature). Finally, we have also generated some trace files for basic operations with arrays by means of C++ programs.

These traces have different formats. In particular, the DSMCache simulator, until now, admits the following memory trace formats: the trace format of SMPCache (Vega 2005), the trace format of LIMES (Ikodinovic et al. 1999), the canonical format for multiprocessor traces developed by Anant Agarwal (Agarwal 2004), besides the new trace format created for DSMCache.

Figure 7 presents an example in which a memory trace with DSMCache format (files with “\*.spr” extension) is being loaded. In this window, we can also observe several trace files with SMPCache format (“\*.prg” extension), which could also be loaded in the simulator.

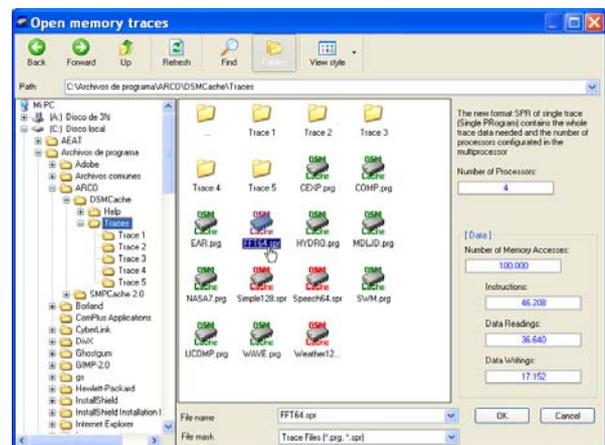


Figure 7: Window for loading memory traces in the simulator. It is possible to load memory traces in both SMPCache and DSMCache format

Figure 8 shows an example of the new trace format for DSMCache. The first comments indicate the number

of processors for which the trace was created, as well as the quantity of accesses that contains, and the distribution for types (instruction captures, data readings or data writings) and for processors. These data are not obligatory, although they speed up the initial management of the trace by the simulator.

```
# Number of Processors
4
# Number of Memory Accesses
11
# Number of Instructions
4
# Number of Data Readings
5
# Number of Data Writings
2
# Accesses by Processor
P0 Acce=3 Inst=1 Read=1 Writ=1
P1 Acce=3 Inst=1 Read=2 Writ=0
P2 Acce=3 Inst=1 Read=1 Writ=1
P3 Acce=2 Inst=1 Read=1 Writ=0
# Trace Data
T=0
P0=0 d80d0200
P1=0 d80d0208
P2=0 d80d0210
P3=0 d80d0218
T=4
P0=2 980e0200
P1=2 e80d0200
P2=2 242a0400
P3=2 b6150200
T=10
P0=3 980e0210
P1=2 e80d0220
P2=3 242a0420
```

Figure 8: DSMCache trace format

After that, the trace lists the accesses grouped by time/simulation steps ( $T=nnn$ ). Each time step usually includes one access by processor, though it is not required that all the processors have an associate access. Each line means one memory access, using the following format:

$$P_i = \text{Type Address}$$

Where:

- $P_i$  indicates the processor that does the access, where  $i$  ranges from 0 to the configured number of processors (minus 1, because  $i$  begins from 0).
- $Type$  is a decimal number that identifies the memory access operation type demanded by that processor in a given time: to capture an instruction (0), to read a memory data (2), or to write a data in memory (3).
- $Address$  is a hexadecimal number that indicates the effective address of the memory word to be accessed. This address will be translated by the simulator for locating the word in the memory system block structure.

As it can be observed, we have looked for an easily intelligible format, which allows us to modify the trace using a common text editor. In conclusion, in the example of figure 8, this memory trace presents 4 instruction captures of a certain parallel program. Five accesses imply data reading, and two require writing in memory. In total, those 4 instructions imply 11 memory accesses.

As for the LIMES trace format and the canonical format for multiprocessor traces proposed by Anant Agarwal, we have also developed a converter from these formats to both the DSMCache and SMPCache format. Figure 9 shows the interface of this trace format converter.



Figure 9: Graphic interface of the trace format converter

## 5. PRACTICAL RESULTS

In this section we show some experiments and the associated conclusions for a set of specific architectures with a set of concrete traces. We will thus illustrate the use of the simulator and some of the possible practical experiments to be carried out. A wide set of experiments has been carried out, studying parameters of interest like the locality of different programs, the influence of the cache size, the mapping, the replacement policies, the block size,... on the miss rate, network traffic, etc. For reasons of space, we will only show some of these experiments.

Table 2: Multiprocessor traces used

Name	References	Language	Comments
FFT	7,451,717	Fortran	Parallel application that simulates the fluid dynamics with FFT
Speech	11,771,664	---	Kirk Johnson and David Kranz (both at MIT) are responsible for this trace
Simple	27,030,092	Fortran	Parallel version of the SIMPLE application
Weather	31,764,036	Fortran	Parallel version of the WEATHER application, which is used for weather forecasting. The serial version is from NASA Space Flight Center, Greenbelt, Md.

In these experiments we have studied traces with tens of millions of memory accesses (references) for four benchmarks (*FFT*, *Simple*, *Speech* and *Weather*). These traces were provided by David Chaiken (then of MIT) for PARL (PARL 2004). The traces represent several real parallel applications. A summary of the traces is shown in table 2. *FFT*, *Simple* and *Weather* traces were generated using the post-mortem scheme implemented by Mathews Cherian with Kimming So at IBM.

We are first going to analyse an architecture with eight processors, SGI-Origin directory protocol, MESI (or Illinois, for a detailed description see (Culler et al. 1999)) snoopy protocol, 16-bit words, 64-byte blocks, four-way set associative caches and LRU replacement. On figure 10 the miss rate versus cache size is represented. From this figure we can obtain the conclusion that the global miss rate for the system decreases as the caches size increases, because capacity and conflict misses are reduced. For large cache sizes the miss rate is stabilized, this shows us the compulsory and coherence misses, which are independent of the cache size. Current measurements demonstrate that the shared data has less spatial and temporal locality than other data types. In other words, in general, parallel programs exhibit less spatial and temporal locality than serial programs. It is thus usual for the miss rates to be higher for multiprocessor traces than for uniprocessor traces. Figure 11 shows the network traffic on the system per memory access for this same experiment. The conclusions we obtain are similar to the previous ones for the miss rate. The network traffic is reduced as the miss rate decreases because of two fundamental reasons.

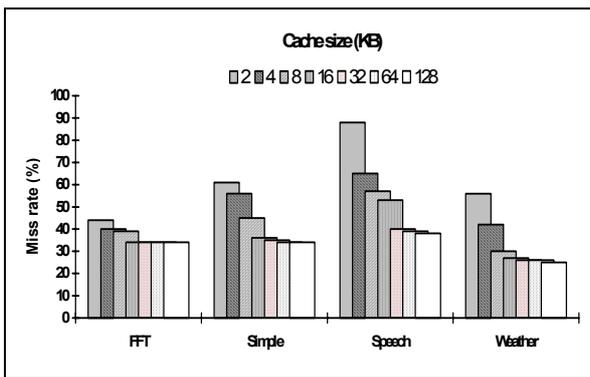


Figure 10: Miss rate versus cache size

On the one hand, there are less data transfers from the distributed shared main memory to the caches. On the other hand, due to there being less misses, less network transactions are necessary in order to manage the cache coherence protocols.

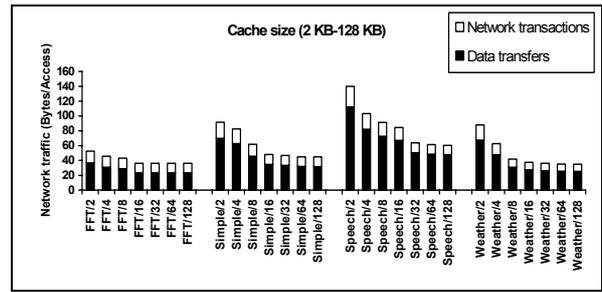


Figure 11: Network traffic versus cache size. The traffic is split into data transfers and network transactions

We are now going to discuss the three traces that were generated using the post-mortem scheme (*FFT*, *Simple* and *Weather* traces). We will study the influence of the number of processors on the miss rate, the network traffic, the execution time, and the speedup, bearing in mind the previously mentioned architecture (SGI-Origin directory protocol, MESI snoopy protocol, 16-bit words, 64-byte blocks, four-way set associative caches and LRU replacement). Figures 12 and 13 present the results for the miss rate and the network traffic.

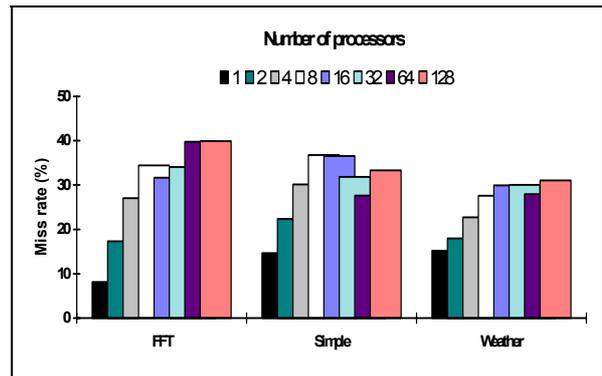


Figure 12: Miss rate versus number of processors

We can conclude that, in general, the greater the number of processors for a parallel application, the higher the miss rate and network traffic. This is possible because with a invalidation-based protocol, like the MESI protocol, the more processors there are, the more possible it is that several caches will share the same block, and hence that in a writing operation, a cache forces the other caches to invalidate that block, producing new misses (coherence misses) and increasing the number of block transfers.

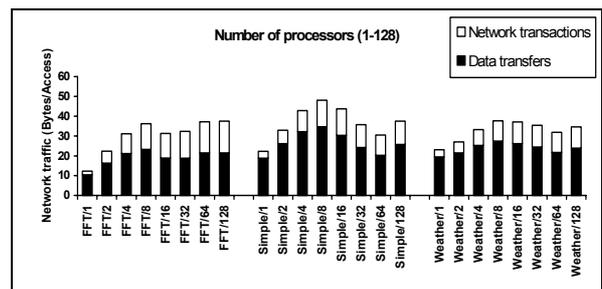


Figure 13: Network traffic versus number of processors

On the other hand, the greater the number of processors, the greater the number of network transactions that are needed to hold the cache coherence. In short, as the number of processors increases for a given problem size, the *working set* (Denning 1968) starts to fit in the cache, and local misses (mainly, capacity misses) are replaced by coherence misses.

In this line, figure 14 presents the network traffic, splitting into local and remote traffic. We can see clearly how the local traffic (traffic inside the node) is replaced by remote traffic (traffic using the interconnection network) when the number of processors grows.

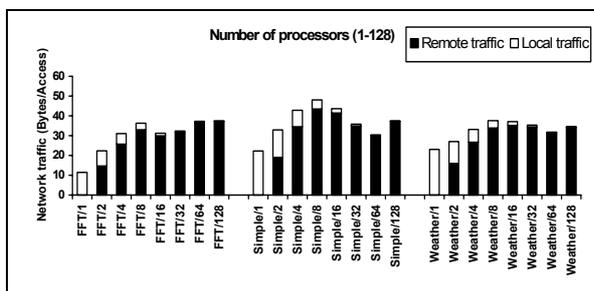


Figure 14: Network traffic versus number of processors. The traffic is split into remote and local traffic

Figures 15 and 16 display the results for the execution time and the speedup.

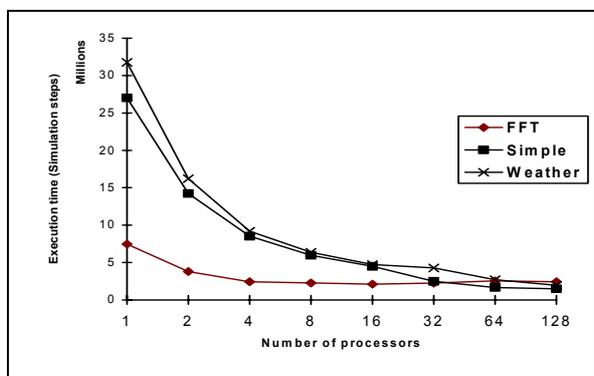


Figure 15: Execution time versus number of processors

In this case, the theoretical predictions are also fulfilled, which indicate a decrease of the execution time, or an increase of the speedup, as the number of processors is scaled. Anyway, this performance increment in the execution of the parallel programs also depends on the programs, and not only of the number of used processors (double processors does not indicate double speedup). Furthermore, in order to obtain a better performance we should have improved the balance of work done by each processor, as the number of processors was increased.

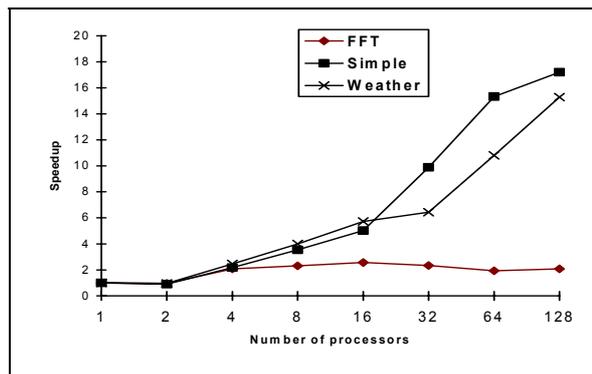


Figure 16: Speedup versus number of processors

## 6. CONCLUSIONS

In this paper we have shown the main features of a cache memory system simulator on scalable architectures with distributed shared memory. We think the simulator has many advantages from an educational point of view. Students could use it as a tool for experimenting the different theoretical aspects about cache memories and DSM systems in the regular courses of Computer Architecture. In this way, students would acquire better and larger knowledge about these subjects.

With research goals, we think it is an attractive and easy tool in order to study memory models on DSM systems that have a better performance for certain parallel programs.

At present, we are extending the simulator with all the SMPCache functionalities (Vega 2005). In this way, we will have a simulator for cache memory systems on any kind of multiprocessor: scalable systems (to many processing nodes) with distributed shared memory (usually based on directory coherence protocols and a scalable interconnection network), and symmetric multiprocessors (small-to-moderate scale multiprocessors, usually based on a centralized shared memory, a shared bus and snoopy coherence protocols).

## 7. REFERENCES

Agarwal, A. 2004. *bigDIRN simulator*. Available at <ftp://tracebase.nmsu.edu/pub/tracebase4/MP/mit/schedule/src/dirsim/>.

Chung, Y.; H. Kim; J.-W. Park and K. Lee. 2001. "Performance Evaluation for CC-NUMA Multiprocessors Using and OLTP Workload". *Microprocessors and Microsystems*, Elsevier, vol. 25, no. 4 (Jun), pp. 221-229.

Culler, D.E.; J.P. Singh and A. Gupta. 1999. *Parallel Computer Architecture. A Hardware/Software Approach*. Morgan Kaufmann.

Deijl, E.; G. Kanbier; O. Temam and E.D. Granston. 1997. "A Cache Visualization Tool". *Computer*, IEEE, vol. 30, no. 7 (Jul), pp. 71-78.

Denning, P.J. 1968. "The Working Set Model for Program Behavior", *Communications of the ACM*. ACM, vol. 11, no. 5 (May), pp. 323-333.

- Edler, J. and M.D. Hill. 2005. *Dinero IV Trace-Driven Uniprocessor Cache Simulator*. Available at <http://www.cs.wisc.edu/~markhill/DineroIV/>.
- Gee, J.D.; M.D. Hill; D.N. Pnevmatikatos and A.J. Smith. 1993. "Cache Performance of the SPEC Benchmark Suite". *IEEE Micro*, IEEE, vol. 3, no. 2 (Aug), pp. 17-27.
- Gómez, J.A.; J.M. Sánchez and J.A. Moreno. 1996. "An Educational Tool for Testing Hierarchical Multilevel Caches". *ACM Computer Architecture News*, ACM, vol. 24, no. 4 (Sep), pp. 11-15.
- Gómez, J.A. 2005. *SISMEC Simulator*. Available at <http://arco.unex.es/jangomez/investigacion/proyectos/sismec/sismec.html>.
- Hennessy, J.L. and D.A. Patterson. 2003. *Computer Architecture. A Quantitative Approach*. 3<sup>rd</sup> edition, Morgan Kaufmann.
- Hill, M.D. 1985. *DineroIII Documentation*. Unpublished Unix-style man page, Univ. of California, Berkeley (Oct).
- Hill, M.D. and A.J. Smith. 1989. "Evaluating Associativity in CPU Caches". *IEEE Transactions on Computers*, IEEE, vol. 38, no. 12 (Dec), pp. 1612-1630.
- Hunt, B.R. 1997. *Acme Cache Simulator*. Available at <ftp://tracebase.nmsu.edu/pub/tracebase4/src/acs/> (Jun).
- Hwang, K. 1993. *Advanced Computer Architecture: Parallelism, Scalability, Programmability*. McGraw-Hill.
- Ikodinovic, I.; D. Magdic; A. Milenkovic and V. Milutinovic. 1999. "Limes: A Multiprocessor Simulation Environment for PC Platforms". 3<sup>rd</sup> *International Conference on Parallel Processing and Applied Mathematics (PPAM)*, Poland (Sep), pp. 398-412.
- PARL (Performance and Architecture Research Lab), New Mexico State University (NMSU). 2004. *The NMSU TraceBase*. Available by anonymous ftp at <ftp://tracebase.nmsu.edu>.
- Sima, D.; T. Fountain and P. Kacsuk. 1998. *Advanced Computer Architectures. A Design Space Approach*. Addison-Wesley.
- Smith, A.J. 1986. "Bibliography and Readings on CPU Cache Memories and Related Topics". *ACM Computer Architecture News*, ACM, vol. 14, no. 1 (Jan), pp. 22-42.
- Stallings, W. 2003. *Computer Organization & Architecture: Designing for Performance*. 6<sup>th</sup> edition, Prentice-Hall.
- Uhlrig, R.A. and T.N. Mudge. 1997. "Trace-Driven Memory Simulation: A Survey". *ACM Computing Surveys*, ACM, vol. 29, no. 2 (Jun), pp. 128-170.
- Vega, M.A.; J.M. Sánchez and J.A. Gómez. 2001. "An Educational Tool for Testing Caches on Symmetric Multiprocessors". *Microprocessors and Microsystems*, Elsevier, vol. 25, no. 4 (Jun), pp. 187-194.
- Vega, M.A. 2005. *SMPCache simulator*. Available at <http://arco.unex.es/smpcache>.

## AUTHOR BIOGRAPHIES

**MIGUEL A. VEGA-RODRIGUEZ** is professor of Computer Architecture in the Department of Computer Science, University of Extremadura, Spain. He received a PhD degree in Computer Science from the University of Extremadura. Dr. Vega-Rodriguez's main research interests are cache memory systems on multiprocessors (including their simulation), and applications of reconfigurable computing (using FPGAs) to image processing and cryptography.

**R. JORGE GIL-RAMOS** received a MS degree in Computer Science from the University of Extremadura, Spain, in 2004. His main research interest is simulation of cache memory systems on multiprocessors.

**JUAN A. GOMEZ-PULIDO** is professor of Computer Architecture in the Department of Computer Science, University of Extremadura, Spain. He received a PhD degree in Computer Science from the Complutense University of Madrid in 1993. His main research interests are performance evaluation of the multilevel cache memory systems and applications of reconfigurable hardware to different fields of signal processing.

**JUAN M. SANCHEZ-PEREZ** is Professor of Computer Architecture in the Department of Computer Science, University of Extremadura, Spain. He received a PhD degree in Physics from the Complutense University of Madrid in 1976. His research interests are modern computer architectures, applications of reconfigurable hardware and logic design.

# THUMPSim: One Simulation Framework for Processor Architecture Evaluation

Youhui Zhang

Yu Gu

Dongsheng Wang

Weimin Zheng

Department of Computer Science

Tsinghua University

100084, Beijing, P.R.China

E-mail: zyh02@tsinghua.edu.cn

## KEYWORDS

Cycle-accurate simulation, Co-verification, embedded processor, MIPS.

## ABSTRACT

Evaluation has become the mainstay of computer architecture design and relies heavily on simulators and simulator infrastructure. To improve the evaluation performance and design efficiency, one process simulation framework, THUMPSim, is implemented. THUMPSim can provide the designer with different evaluation accuracies on several simulation levels. Its core contains two main parts, the architecture-independent control module and the definition of the processor, to provide an infrastructure for pluggable components to improve its flexibility. In addition, an event-driven signal update algorithm is employed to simulate concurrent activities in real systems. Some other techniques, including object-oriented component definition, and code auto-generation, are also employed to predigest the simulation work. THUMPSim is employed by our team to develop THUMP107 embedded CPU and speed up the software/ hardware co-development remarkably.

## INTRODUCTION

As ASIC designs explode in size and complexity, evaluation has become the mainstay of processor architecture design before the traditional RTL coding, layout design and verification flow. The tremendous complexity of systems is making it both difficult to reason about and expensive to develop. Detailed software simulations have therefore become essential for evaluating ideas in the architecture field. Industry uses simulation extensively during processor and system design because it is the easiest and least expensive way to explore design options. Simulation is even more important in research to evaluate radical new ideas and characterize the nature of the design space (Skadron et al. 2003). However, current simulation infrastructure is written in ways that limit code sharing and lacks a satisfying component-based architecture. Current tools tend to dictate the majority of research that gets done, because it is easier to study questions that can be answered with

existing tools. This means that there are parts of the design space that end up only lightly explored, because of the difficulty of doing so.

The workshop summary (Adve et al. 2001) suggests that simulation frameworks and simulator construction frameworks were superior to monolithic simulators or simulator. But in fact, most existing simulation tools belong to the latter. As we know, examples of some popular simulators include the SPIM simulator (Breach et al. 1999), the simplescalar simulator (Burger and Austin 1998) and the SuperDLX simulators (Moura 1993). Once developed, such simulators are difficult to retarget to a modified micro architecture without a significant amount of effort.

Therefore, we develop a simulation framework and its construction tool, named THUMPSim, to provide users with a method to construct the fast prototype of the target design. In contrast with the most existing simulators, THUMPSim has the following features.

1. A component-based infrastructure is completed. That is, it contains two main parts, the architecture-independent driven engine and the definition of the target design, to provide an infrastructure for pluggable components to improve its flexibility.
2. An event-driven signal update algorithm is employed to simulate all hardware activities. Based on this algorithm, it provides a set of definition rules for the design target to resemble concurrent communication in real systems---not the function-call interface found in sequential programming languages such as C.
3. Based on the previous definitions, designers can describe the interface and function of any component, signal and instruction in the target design, then a pre-compilation-program will convert the description to proper C++ codes automatically to generate the target simulator, which completes the construction framework and ease the development.

Owing to THUMPSim, we implemented one cycle-accurate processor simulator fast when our research team began to design a MIPS 4KC (MIPS Technologies, Inc. 2002) like embedded processor. This simulator is an accuracy C-model of our CPU, which implements the micro-architecture of the processor and some basic peripherals. Based on this simulator, software engineers

began to start work on OS and applications in parallel to the development and manufacturing of the silicon components. In addition, the simulator plays a very important role in our validation and verification work. THUMPSim helps us to speed the validation of the functionality of our processor accurately, and most of the hardware and software development can be executed simultaneously.

The rest of this paper is organized as follows: The next section presents the related projects and shows what aspects are different in THUMPSim. The detailed design and implementation of the framework are described in section 3. Section 4 presents the brief overview of its usage for our CPU design and the relative workflow. Section 5 summarizes this paper.

## RELATED WORK

Examples of some popular simulators include the SPIM simulators, the simplescalar simulator and the SuperDLX simulators. Once developed, such simulators are difficult to retarget to a modified micro architecture without a significant amount of effort. And these simulators lack the cycle accurate feature so they are unfit for our development.

In fact, a number of researchers met under the aegis of the US National Science Foundation's Computer Systems Architecture program to discuss the experimental and evaluation problems that processor architecture research faces in December 2001. The attendants agreed that simulation frameworks and simulator construction frameworks were superior to monolithic simulators or simulator code libraries written in sequential languages. In addition, modularity, portability and accuracy should be considered seriously (Adve et al. 2001). Some examples frameworks include Asim (Emer et al. 2002) and the Liberty Simulation Environment (Vachharajani et al.

2002).

Therefore, these issues were also evaluated firstly when we began to design THUMPSim. In our sense, it is should be a simulation framework rather than a single simulator. And it should own a component-based infrastructure and provide some construction tools for designers to ease the development. According to this policy, we design and implement the framework, which will be introduced in the following sections.

## ARCHITECTURE OF THUMPSIM

### Overview

We define accuracy as a measure of the fidelity of the simulated machine to the actual, while flexibility means the ability of the simulator to continue to explore a broad design space. As more and more features are modeled accurately, it becomes increasingly difficult to support design space exploration that strays far from the chosen direction. In contrast, flexibility in turn decreases as a consequence of increasing accuracy. To pursue both flexibility and accuracy, THUMPSim accomplishes a component-based infrastructure, which has the following features.

1. The framework core contains two parts, the architecture independent driven engine and the definition of the processor. So it is suitable for construction of different processors with high flexibility.
2. It gives object oriented component definition syntax. Based on the method, designers can describe some definitions on different levels for the target design. That is, the target can be simulated with different accuracies, including cycle-accuracy and functional accuracy.

As showed in Figure 1, THUMPSim contains two main parts, the simulation framework (gray modules) and the definition of the target design.

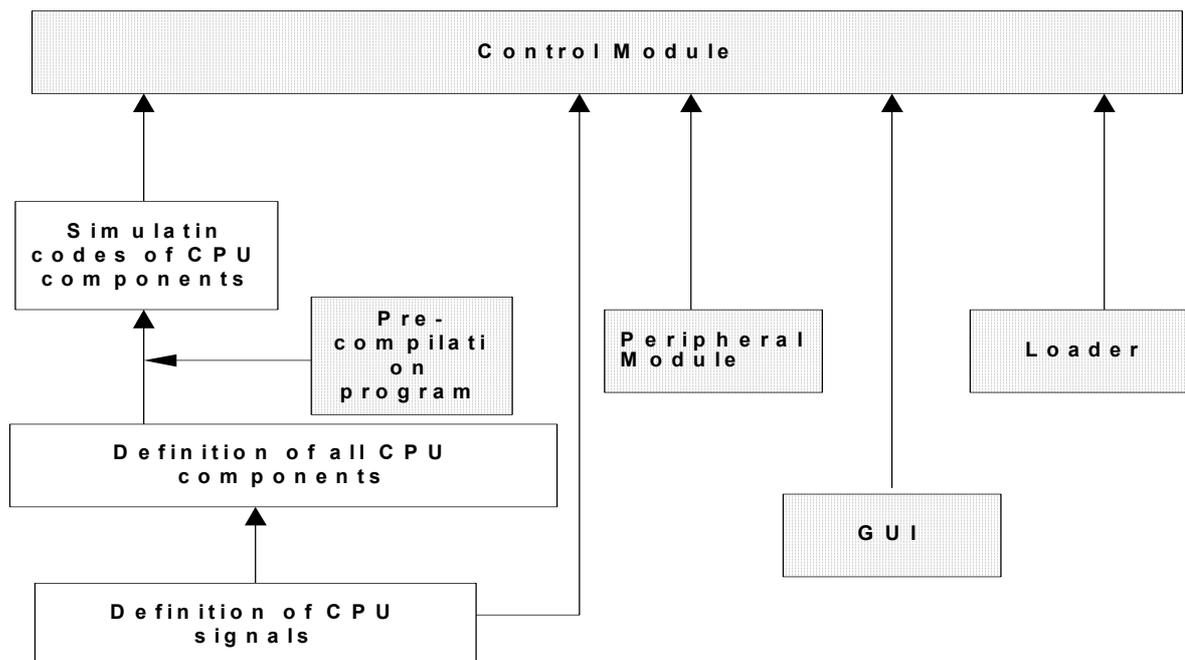


Figure 1: The Internal Modules of THUMPSim

Control module manages the whole system and schedules other modules to implement the driven algorithm. Loader can load OS or verification vectors into the simulator. And the key board, UART, terminal console and memory controller are simulated by peripheral module. It means different simulated designs can share the common modules, which increases the reusability of code. At the same time, to separate definitions of the target design from the framework improves its flexibility. That is, modification of some components, even of the whole architecture, can be implemented just through altering the definition of necessary signals and components. When the elementary architecture design is finished, designers can implement the simulation as the steps below.

1. Signals connecting processor components, inter-pipeline-stage registers and all instructions are declared according to the definition rule.
2. Processor components, including Decoder, ALU, MMU, DCache / ICache, BIU, Shifter, MDU, controller, CP0 and so on, are defined. Based on the predefined virtual interfaces in THUMPSim, the function and signals of one component can be described in a C++ like language.

Based on the description of interconnection signals and components, THUMPSim provides the tool to construct the target architecture, which will be introduced in 3.3. In fact, the original version of our CPU design was 5-stage and we implemented its simulator firstly. And then, we modified it into 7-stage, which only took us less than 3 weeks owing to the high flexibility of THUMPSim.

### Definition Rules

One instruction definition sample looks like the following line, which describes the format of "ADD" instruction.

```
INST (I_ADD, "add r%d, r%d, r%d", RD(inst),
      RS(inst), RT(inst))
```

Instruction description is used to disassemble a decoded instruction to show it on the GUI. RD, RS and RT are macros to draw different subfields of one instruction. For signals, the following format is employed:

```
SIGNAL_DEFINE(Name, Type, Width, Stage)
```

Name: the identification.

Type: wire signal or register signal.

Width: the data width.

Stage: the pipeline stage that the signal belongs to.

Two different types of signal are supported, wire signal and register signal. One type is just the connection between two or more components while the other is used to describe the inter-pipeline-stage registers or the general register file. And the stage info is only meaningful to register signals. Two typical signals are presented as follows.

```
SIGNAL_DEFINE(Alu_Out_EX_MEM_v32, ST_REG,
              32, STAGE_MEM)
SIGNAL_DEFINE(BranchPC, ST_WIRE, 32,
              STAGE_NONE)
```

The first is an inter-pipeline-stage-register, Alu\_Out\_EX\_MEM\_v32, that stores the 32-bit computing result of ALU component and the content of this register is used in MEM stage.

The second describes a 32-bit wire signal, BranchPC, which is the target of a branch instruction.

Definition of the logic of every component is stored in a distinct .def file, which looks like a C++ object declaration file.

The .def file is made up of five fields.

1. *Component* contains three sub fields:  
Name: the identification.  
Input :the set of its input signals  
Output: the set of its output signals  
All input / output signals should be declared before.
2. *Variable* describes the internal local variables.
3. *Function* defines the local functions used by *Execute* field.
4. *Initialize* gives initialization codes executed at the start.
5. *Execute* presents the function of this component, which is called by the driven engine when any input signal is modified.

Based on the previous definition, the whole net list of the target design can be described and one event-driven signal update algorithm is employed to drive all signals and components.

We should note that different levels of simulation can be implemented on the definition rules. For example, any stage of the target processor can be regarded as one single component to complete its function after the inter-pipeline signals have been defined. Of course, to divide the stage component into detailed components, including decoder, ALU, MMU, Cache and so on, can define more accurate descriptions with higher simulate accuracy. The choice lies on the users' needs. General speaking, a rough simulation is required in the early stage while the more accurate will be needed later.

### Code Auto-generation and Debugging Functions

THUMPSim provides a simulator construction framework to enable rapid exploration of design alternatives by automatically weaving architectural component models together as implied by the definitions of the design. When the definitions are completed, one pre-compilation-program converts them to proper C++ code automatically. And then, the code can be linked to the other modules to create the target simulation executable. For example, the definition of Adder is presented in Figure 2 and the converted code is listed in Figure 3. We can see that Adder has three input and three output signals. After conversion, the definition becomes a C++ class and all input/output signals are maintained in respective arrays, which will be used by the event-driven signal update algorithm in the next section.

Some useful debugging functions are provided in the executable. First, a round signal trace mechanism is supported. Behaviors of the target design in every cycle are logged, including memory & I/O accessing, register

read & write, cache miss, TLB miss and so on, which can be used to evaluate the performance of the design. Moreover, these traces can be compared with those from RTL-level simulation directly to locate bugs if existing. Second, instruction & data breakpoints are implemented. In addition, most signals and registers can be watched and modified on-line in THUMPSim.

```
[Component]
NAME = Adder
INPUT = Add_OpA_v32
INPUT = Add_OpB_v32
INPUT = Add_Op_Ctrl_DE_EX_v3
OUTPUT = Add_Out_v32
OUTPUT = Add_Flag
OUTPUT = Overflow_Exception
[Execute]
.....
```

Figure 2: The Definition of Adder

```
Adder.h:
class CAdder_COM : public CComponent{
public:
    void Execute();
    CAdder_COM();
    ~CAdder_COM();
    bool Initialize();
protected:
};

Adder.cpp:
void CAdder_COM::Execute()
{
.....
}
CAdder_COM::CAdder_COM(){
    m_strComponentName =
    "Adder_COM";
    m_nInputNum = 3;
    m_nOutputNum = 3;
    m_psgnInput=new
    SIGNAL_ENUM[3];
    m_psgnInput[0]=SG_Add_OpA_v32;
    m_psgnInput[1]=SG_Add_OpB_v32;
    m_psgnInput[2]=SG_Add_Op_Ctrl_
    DE_EX_v3;
    m_psgnOutput=new
    SIGNAL_ENUM[3];
    m_psgnOutput[0]=SG_Add_Out_v32;
    m_psgnOutput[1]=SG_Add_Flag;
    m_psgnOutput[2]=SG_Overflow_Exc
    eption;
}
Bool CAdder_COM::Initialize()
{ return TRUE; }
```

Figure 3: The Converted Code

Moreover, the status of the simulated target can be saved to a file. Then the status file will be loaded into THUMPSim to restore the previous state if necessary, which is very useful to debug a long term program or the OS.

## Event-driven Signal Update Algorithm

As we know, the definitions of the target design will be converted into proper C++ code. It is a sequential programming language provides the function-call interface and is not suitable to describe the concurrent activities in real systems. To simulate the transformation of all signals updated by components concurrently, one signal update algorithm is implemented to simulate the hardware execution with high performance.

Event-driven mechanism is employed in the algorithm, that is, one component will be executed if and only if any of its input signals is modified in the current cycle. And our algorithm guarantees that no component will be called more than once in a cycle. To perform this algorithm correctly and rapidly, THUMPSim must generate the execution sequence of all components automatically. The following steps are performed just once at the start to make the sequence.

1. Attach all components to Group 0.
2. Attach all register signals to terminated signal list.
3. one variable, CurrentGroup, is set to zero.
4. Look up components that belong to CurrentGroup. If none, go to step 7. Otherwise for every matched component, increase its Group NO by one if no input signal belongs to terminated signal list.
5. Look up components that belong to CurrentGroup. If none, go to step 7. Otherwise attach all output signals of all matched components to terminated signal list.
6. Increase CurrentGroup by one and go to step 4.
7. End.

When the execution sequence is fixed, our event-driven signal update algorithm will be called once in every simulated cycle.

1. Empty the current execution list.
2. Browser all signals. If one register signal is modified in the previous cycle, attach all components that employ this signal as the input to the current execution list.
3. one variable, CurrentGroup, is set to zero.
4. Browser the current execution list. If no component belongs to CurrentGroup, go to step 6.
5. If any, execute the component and check its output signals. If any has been modified, move all components that employ this signal as the input to the current execution list. Remove the executed component from the current execution list and go to step 4.
6. Increase CurrentGroup by one. If CurrentGroup is not greater than the number of groups, go to step 4.
7. Update all register signals and end!

Two values are maintained for every register signal.

One saves the original value used by components in the current cycle and the other stores the new value generated. So to update one register signal in step 7 means that the new value will overlay the original at the end of this cycle.

## The Workflow

For one instruction, its execution flow in THUMPSim can be described as the following basic steps.

1. It is fetched by the simulator, which is triggered by the modification of the content of PC register. During the fetch stage, instruction cache is accessed first. If cache misses, the memory main will be visited through BIU.
2. Then, decoder is triggered to parse the instruction and stores the result into the pipeline stages between DE and RF stages.
3. Register files are accessed if necessary to get the values of source registers, which will modify the relative signals to trigger execution components. So this instruction will be performed cycle by cycle till WB stage.

Of course, there are often several instructions running currently in the simulator and the potential interlock / hazard among them should be coped with by the pipeline control component.

We implement THUMPSim in MFC 7.0 and its GUI is showed in Figure 4. Window 1 displays the current instructions running in the processor and the pipeline stages they belong to. All internal signals and inter-pipeline-stage registers can be watched in windows 2. Window 3 is the register file view and window 4 is the memory view.

Now THUMPSim can load and run some types of verification vectors directly:

1. Executable files in ECOFF format, including the Linux OS kernel.
2. Other vectors written manually or generated by RTPG.

## THE USAGE

THUMPSim is used in the design flow of an embedded

processor developed by our team. The processor, THUMP107, is a MIPS 4KC like embedded processor. It was produced employing the TSMC 0.18um technology in October, 2003. And tests show that its highest frequency is 500MHz. The simulator played an important role in the design flow that is divided into four stages as showed in Figure 5.

During the design stage, the elementary micro-architecture specification was finished and designers implemented the simulator based on the specification. Then software engineers started to port OS for THUMP107 and develop applications in THUMPSim in parallel to the development and manufacturing of hardware components.

Many jobs, listed as follows, had been done in the simulation environment without any final hardware and prototype in hand.

1. Migrate OS
2. Test Spec2000 Integer benchmarks
3. Develop boot code for the system
4. Write and test most verification vectors
5. Implement the test bench of the whole system

In the co-design and co-verification stage, HDL codes were finished with the guidance of THUMPSim code. Verification vectors were also performed. It is necessary to note that all verification vectors, testing applications and OS suitable for the simulator can run on the RTL-level and the gate-level directly. That is, the c++ core of the simulator is replaced by the RTL core through PLI interface and the remaining parts, including basic peripherals, DRAM module, control module and the GUI are unmodified. Then we can operate the mixed test bench in the way of THUMPSim and the RTL simulator employed is Cadence NC-Verilog.

So testing traces generated by THUMPSim are able to be compared with those from the lower level simulation directly to locate bugs if existing. Its flow is presented

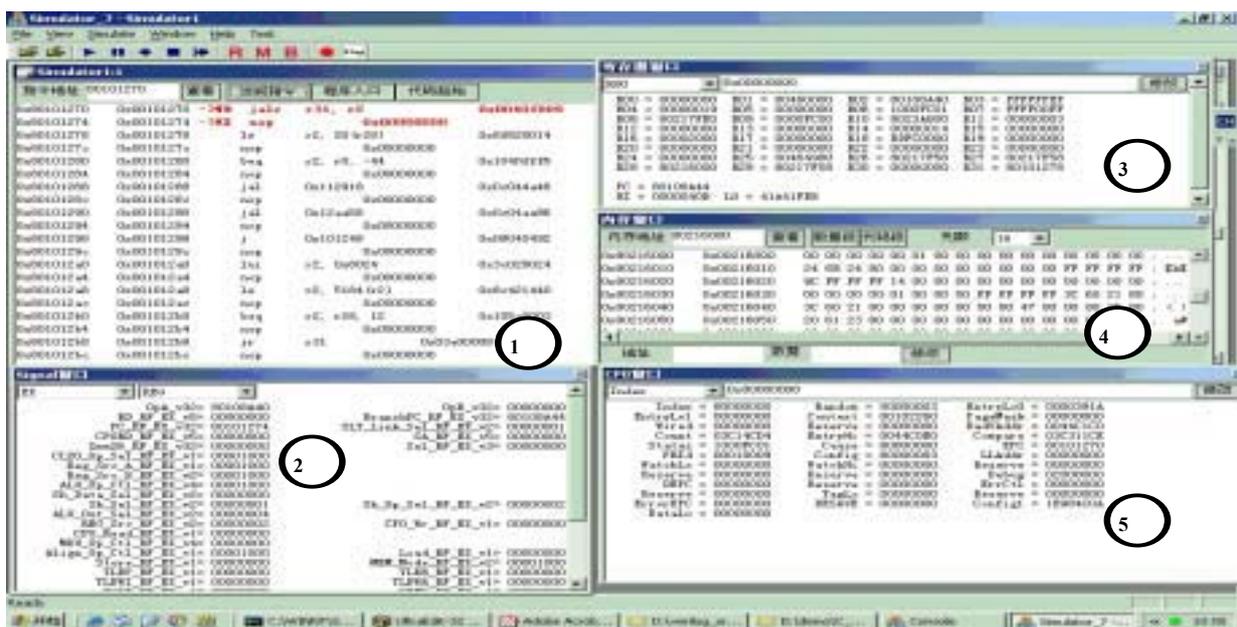


Figure 4. GUI of THUMPSim

briefly in Figure 6.

At last the post-synthesis prototype was tested under the same test bench. In addition to co-design and co-verification, software engineers are also optimizing the compiler for our CPU based on THUMPSim.

## CONCLUSION

This paper presents one process simulation framework written in C++. It can provide different evaluation accuracies for the designer based on several simulation levels. This framework contains two main parts, the architecture-independent driven engine and the definition of the processor, to provide an infrastructure for pluggable components to improve its flexibility. In addition, an event-driven signal update algorithm is employed to simulate all hardware activities, which resembles concurrent communication in real systems.

It is employed by our team to develop THUMP107 embedded CPU and speed up the software/ hardware co-development remarkably. That is, before writing HDL codes, it has been developed for OS migration, writing verification vectors and validating the architecture design. In addition, we implemented a co-verification environment that contains the simulator. When testing vectors are produced, both of the c-simulator and the design simulator of the target CPU in different levels can execute them. So testing traces generated can be compared automatically to locate bugs if existing, which simplify the verification process.

## REFERENCES

- Adve, K et al. 2001. "NSF Computer Performance Evaluation Workshop: Summary and Action Items." (December), [http://www.princeton.edu/~mrm/nsf\\_sim\\_final.pdf](http://www.princeton.edu/~mrm/nsf_sim_final.pdf).
- Breach, S. E. 1999. "Design and Evaluation of a Multiscalar Processor." PhD thesis, Computer Sciences Department, University of Wisconsin-Madison, (Feb).
- Burger, Doug and Austin, Todd M. 1998. "The SimpleScalar Architectural Research Tool Set, Version 2.0." Technical report, UW-Madison Computer Architecture Group, University of Wisconsin-Madison. <http://www.cs.wisc.edu/~mscalar/simplescalar.html>.
- Emer, J. et al. 2002. "Asim: A Performance Model Framework." *IEEE Computer*, (Feb), 68-76.
- MIPS Technologies, Inc. 2002. "MIPS32 4kc Processor Core Software User's Manual." <http://www.mips.com>.
- Moura, C. 1993. "SuperDLX a generic superscalar simulator." Technical Report 64, School of Computer Science, McGill University.
- Skadron, K et al. 2003. "Challenges in Computer Architecture Evaluation." *IEEE Computer*, Volume 36, No. 8 (Aug), 30-36.
- Vachharajani, M. et al. 2002. "Microarchitectural Exploration with Liberty." In the *35th International Symposium on Microarchitecture* (Istanbul, Turkey. Nov. 18-22). IEEE, 271-282.

## AUTHOR BIOGRAPHY

**YOUHUI ZHANG** was born in Zhejiang, China and went to Tsinghua University in 1993. Now he is an Assistant Professor in the Department of Computer Science at the University of Tsinghua where he also received his PhD degree. His research interests are cluster fault tolerance, network storage/ computing, and microprocessor design. He is currently working on chip-multiprocessor project supported by the National High-Tech Program in China. Zhang can be reached at [zyh02@tsinghua.edu.cn](mailto:zyh02@tsinghua.edu.cn).

# Performance Measures of Swarm based Active Network for Multiclass Packet Routing-A Simulation study

Constandinos X. Mavromoustakis and Helen D. Karatza

Department of Informatics

Aristotle University of Thessaloniki

54124 Thessaloniki, Greece

E-mail: {cmavrom,karatza}@csd.auth.gr

## KEYWORDS

Bandwidth Clustering, Embedded Packet based Swarms, Tunable Scheduling Scheme.

## ABSTRACT

Pervasive computing environments demand reliability and multiclass QoS for end to end real time applications. The concept of active network has been recently adopted in many distributed environments. This paper describes and examines the combination of the active network concept with swarm based control method called Swarm-based Active Network scheme for optimized and tunable QoS priority routing. This hybrid scenario is using the bandwidth clustering approach in which packets are following the predetermined path allocated by swarm based packets (embedded swarms). This mechanism enables the adaptation of the system to new conditions, and does not require message bro-kers to fulfill multiclass QoS demands. Simulation results reveal the response in terms of the end-to-end delay, available bandwidth and packet loss of this scheme under different traffic measures.

## 1. INTRODUCTION

Active routing has become one of the most attractive methods in wired and wireless communication networks. By combining various hybrid schemes of different active agent-based methods it enhances the overall performance of the system offering end-to-end user reliability and integrity. The objective of the generic concept of active agent-based techniques is to achieve high resource utilization and to reduce user contention for network resources. As load is not uniformly distributed, network resources decrease in the form of efficiency, bandwidth, processing power and memory, leading network to an unpredictable behavior. It becomes evident that the mechanism used for load balanc-ing and resource allocation has to avoid overloaded nodes such that transmission delays (latencies) are minimized, whereas alternative routes should be activated whenever load conditions are drastically changing.

While there are often models and efficient algorithms facing allocation problems in deterministic systems

(typically formulated as matching or flow optimization problems) solutions for the networks with stochastic components are not straightforward. Decentralized schemes [3, 6, 10] involve a number of controllers each of which is supervising a part of a network. This allows the central control to be applied locally to a part of a network enabling all “local” supervisors to cooperate deploying a decentralized exchange of information. Indeed many distributed systems are unreliable and subject to random failures of their components. Examples of such systems are power grids where the distribution ability of the network can be affected by demand overloads and other random events or various transportation/information networks subjected to congestions and intermittent failures. Thus optimally designed systems have to offer end user reliability and integrity by enabling equal share of network resources.

This work examines an active network technique, which combines a reactive and proactive behavior of message passing using the Split Agent Routing Technique (SART) [14, 15] based on swarms. Embedded swarm based packets [8] that are delay sensitive are marked as prioritized. In turn, agents recognize these packets as being a part of a packet. This hybrid method is called swarm-based message passing [15]. It provides a model for active and distributed network data flow organization and also continuous resource reservation on demand. SART technique is applied to an active network using smart/active swarm based packets and nodes. In turn the bandwidth clustering mechanism is activated for priority routing-as will be discussed later-in order to assign a certain bandwidth. A thor-ough study and discrimination is made for the class of service offered at any time in the network as well as for the QoS issue (particularly for delay sensitive packets where a slotted window tangles). We have considered a number of metrics that are associated with network performance and evaluation of the degree of distribution ability. Further-more measures for handling multimedia streaming (MMs) are presented with the relative trade-offs, based on the reliability and QoS offered by the proposed scheme.

The organization of the paper is as follows: In section 2 a description of the basic principles of the swarm-based scheme and adaptive resource allocation is presented

with extensive use of active multiclass bandwidth clustering method. Section 3 draws the simulation results and conclusions and suggestions for future research are summarized in section 4.

## 2. SWARM BASED ADAPTIVE RESOURCE ALLOCATION

Agent-based approach was first introduced and standardized by Appleby and Steward's mobile agent's algorithm [3]. Further studies [4, 6, 10] have shown that an ant-like mobile agent algorithm could be applied to a network with significant optimization of the QoS metrics of the network. In [10], Dorigo and Gambardella used the metaphor of trail laying by ants to certain combinatorial optimization problems [3, 4]. Several agent-oriented approaches [4, 6, 10, 14] have recently been proposed that appeal to principles extracted from Swarm Intelligence (SI) and aspire to solve routing problems to wired and wireless communication networks. On the other hand, within these methods there are some trade-offs that have to be taken into account. These trade-offs deal with network overall performance, such as generated overhead in message passing for agents communication, network utilization, simplicity for the implementation, etc.

Agent based network routing could be biologically inspired and based on insect colonies which exhibit a simple behavior for their communication and living. Real ants have similar behavior with agents. They are represented in the network as artificial agents that bias the network collecting useful information for the whole environment through their hormones called pheromones. In previous researches [8, 14, 15] a hybrid (proactive and reactive) agent behavior is developed in which ants are adapting their communicational behavior to network circumstances, simultaneously splitting themselves for passing information to neighbor nodes. In [15] the SART technique was used for path marking on demand and capacity reservation, and was shown that this method efficiently marks the path and reserves the capacity required for offering optimized QoS service metrics to end users.

One of the major challenges of the service/resource allocation problem is to find algorithms that are reactive and deliver reliable high quality solutions. The nature of the environment and the underlying infrastructure with the consolidated use of end-to-end paradigm have focused the research on transport level as the proper place to address congestion issues. Sources usually react by reducing their flow. Indeed, this is a response to congestion, but it has the unavoidable side effect of reducing the throughput. Figure 1 shows that the flow control method/scheme is closely related to routing schemes used in the network. From a source S to a destination D, the used routing scheme affects the underlying discrete flow control-for any given time t-

which results fluctuation in throughput response of the system.

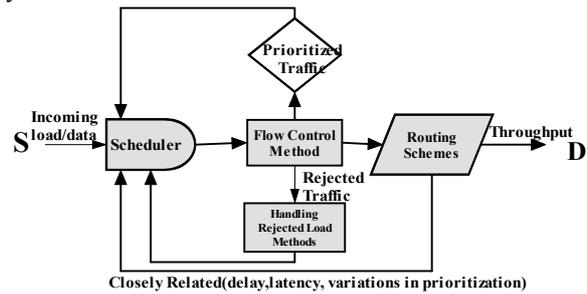


Figure 1: Relation between traffic-incoming load, routing schemes, flow control and throughput.

### 2.1 Embedded swarm based active packets for resource allocation

#### 2.1.1 Active network approach

In an active network, packets are more than just passive data. They are programs executed by the nodes that are passing the packets along or embedded in packets that influence control decisions. This facility allows packets to make on-the-fly decisions about how to route themselves. Tennenhouse et al [2] provides a survey of this work. Active network's approach is motivated by both lead user applications, which perform user-driven computation at nodes within the network.

Current resource discovery approaches for offering optimized QoS have been designed primarily for small networks or for networks where dynamic updates are not common or frequent. Hence, they do not scale well when the number of resources grows, and updates are common. Some of the most popular service discovery systems are Jini from Sun Microsystems, SLP [5], INS (Intentional Naming System) [11], and SDS [9]. In those systems, services advertise their descriptions to resolvers-sometimes to intermediate nodes- which act as resource directories. In turn resolvers allocate the dedicated streaming packets to a link where available resources are adequate to efficiently forward streams from a node to another.

This paper proposes an active network strategy for optimized QoS provision and adaptive resource allocation based on embedded swarm agents. This research is using the SART technique for interaction and updating information between packets and nodes. Resource capacities are provided locally to where demands occur, avoiding cross-network traffic. This active technique is proved to be very adaptive and can rapidly respond to changes in the environment while significantly reduces the generated network overhead.

#### 2.1.2 Swarm based active network

Active packets and nodes can perform different computations reactively. The encapsulation abilities of

such a net-work enable advanced mechanisms for end to end communication to scarce or sensitive resources. Traditional packet headers are replaced with control programs in order to make on the fly decisions.

The swarm based system approach associates pheromone trails to features of the solutions of a combinatorial prob-lem, which can be seen as a kind of adaptive memory of the previous solutions. In our implementation each packet-agent launches in the network and influences the pheromone table [10, 14] by increasing or reducing the entry for the proper destination using antipheromone [14]. The pheromone quantity represents the available end-to-end resources. The pheromone table at each node with neighbors can be measured as:

$$R_i = [r_{1,m}^i]_{n-1,k(i)}, n-1 \text{ destinations and } k \text{ next nodes(1)}$$

Artificial ants are biasing the network by generating at every simulation time step ant-packets destined to every node randomly. In the network ants are walking according to probabilities assigned in pheromone tables and they are visiting one node at every time step. In this way ants increase the entry in the pheromone table corresponding to the node from which they came from:

$$P = \frac{P_{old} + \Delta P}{1 + \Delta P} \quad (2)$$

where  $\Delta P$  is the quantity of pheromone increased and  $P_{old}$  is the previous entry. The other entries in the table of this node are decreased accordingly following the formula:

$$P = \frac{P_{old}}{1 + \Delta P} \quad (3)$$

Routing tables contain a two-way pheromone table parameters (bi-directional links with different capacity) which are maintained in each node, and are expressed as:

$$P_{k(i)}^{i_t \rightarrow n_t} \text{ and } P_{k(i)}^{n_t \rightarrow i_t} \quad (4)$$

where  $k(i)$  are the next nodes<sup>1</sup> for  $N_i$ ,  $n$  is one of the  $n-1$  possible destinations and  $n_t$  is the possible next node at a certain time step.

All probabilities are thresholded [14] between  $\frac{1}{(\text{number\_of\_neighbors})^2}$  and 0.75 in order to prevent the pheromone saturation state.

The route, where intermediate nodes have large pheromone quantities, is selected as the best-chosen path. If the destination can be reached on a hop-by-hop look-up table method, the route is valid otherwise the packet is blocked [14] and lost. Routing table entries updates are measured according to the following:

$$r_{i-1,s}^i(t+1) = \frac{r_{i-1,s}^i(t) + \delta r}{1 + \delta r} \quad (5)$$

where  $\delta r$  is the step size parameter and  $s$  is the source node. Similarly for all neighbors to  $i$ ,  $r_n^i(t)$  is found that:

$$r_{n,s}^i(t+1) = \frac{r_{n,s}^i(t)}{1 + \delta r}, n \neq i-1 \quad (6)$$

In this way smart data packets and ants have an interaction in the means that ants affect the routing tables while data packets influence the service rate of the traffic on nodes, which affects the ants with the delay mechanism. This swarm/agent-oriented approach encompasses the generic agent based concept which enables agents to move around the network, gathering information about the topology of the system and the traffic at a discrete time.

*Active swarm based packets* are used in a network for internal communication and auto-configuration between *packets* and *nodes*. Roughly speaking active packets are self-contained piece of software that has the properties of autonomy and interaction. Packets make on the fly decisions about how to route themselves and learn (gather information) from their passing-by environment. Figure 2 shows the structure of swarm based packet's and node's structure.

A packet consists of a "flow spec" field and a "filter spec" field as in the well known Resource reSerVation Protocol (RSVP) [13]. Packets 'carry' traffic spec and path information from a source to destination and reserve information during their journey to destination. The RS info field of the encapsulated active code is the main swarm based active field in the packet's header. RS info field, in cooperation with other fields in both node and packet, can expose the determined quantity of pheromone for the selected source and destination. This pheromone quantity is based on the available end-to-end resources [8, 15]. The flow spec specifies the desired QoS for the packet. The filter spec defines the set of data packets to receive the QoS defined by the flow spec. The entire service class in packet header (both A1 and active code A2) defines the desired QoS and beyond others, it describes the type of the data flow (priority degree of MM or don't care packets).

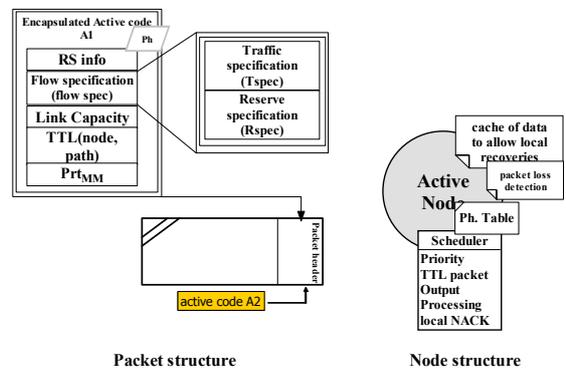


Figure 2: Active swarm based packet's and node's structure.

<sup>1</sup> Nodes can be both endpoints (can be source and destinations) and switches (can perform routing functions).

Active packet's communication, on a continuous basis, enables adaptation of the system to new conditions. This method is not error prone and enables additional information to be passed to neighboring nodes embodied in transmitted packets. This communication between node and packet is opaque to other network layers.

### 2.1.3 Active resource reservation mechanism using bandwidth clustering method

The bandwidth clustering approach is based on the available bandwidth on each data link in the network. Due to the unpredictable incoming traffic measurements that occur at any time in the network, the capacity of each channel (bandwidth) is reduced progressively with an increased flow of packets. Bandwidth clustering method is based on the idea of clustering nodes using different levels of bandwidth [8]. The clustering idea is based on whether the remaining capacity could efficiently be reserved by different streams to better utilize the path. This method takes place after the SART algorithm is applied to the network to bias the paths and overcome the transient state [14]. Overloaded paths are also clustered but the remaining capacity will mark the cluster as a lower bandwidth cluster [8]. The bandwidth clustering method enables the manipulation of different paths that offer different levels of bandwidth based entirely on the information collected from swarm based packets. Furthermore the bandwidth clustering method is associated with a cooperative learning and active environment, producing a robust and decentralized way for adapting link's changing capacities quickly.

Figure 3(i) illustrates the typical decision selections of a proper node  $i$  to destination  $D$ . In figure 3(i) node  $i$  has four different options for the  $D$  destination. Path options  $a$  and  $b$  on one hand are using different intermediate nodes ( $j$  and  $k$  respectively) and different clusters leading to  $D$  destination ( $C1$  and  $C2$  respectively). On the other hand  $c$  and  $d$  are using a combination of clusters  $C1$ ,  $C2$ ,  $C3$  using intermediate nodes  $l$  and  $m$  respectively. Figure 3(ii) shows a topology and the regions arising when clustering with respect to several levels of bandwidth. Having as source node  $A$  and a destination node  $D$ , agents have already marked the path [15] and allocated the remaining capacity to paths. The same time a cluster is being constructed at each path having the available bandwidth of the channel with the lowest free remaining capacity (1).

$$\text{Max}(C_{A \rightarrow D}) = BW \quad (7)$$

In (7)  $BW$  is the available bandwidth, which is determined by the lowest free remaining capacity in the path from  $A$  to  $D$  (bottleneck concept).

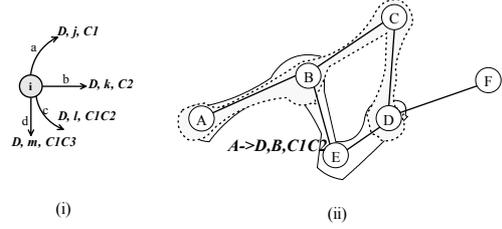


Figure 3(i) and (ii): A topology and the regions arising when clustering with respect to several levels of bandwidth. Color regions represent nodes connected by links with higher free bandwidth (solid lines). Dashed lines indicate more congested links.

After split agents measure the path from a source to destination, the clustering bandwidth mechanism is activated for each node estimating each time step the differences in their links. In figure 3 ( $i$ ,  $ii$ ) all nodes are informed and the clusters are constructed according to capacity reservation information spread by agents onto nodes. Thus a packet has to decide whether Cluster 1 or Cluster 2 should be chosen for reaching destination. These decisions are made according to information of remaining bandwidth which was obtained in the previous time step. In that step packets passed and updated the information as expressed in (5) and (6). Although routing is performed through the swarm based method, the bandwidth reservation method occurs simultaneously with routing decision. Bandwidth reservation method is based on the priority degree of each packet. Swarm based routing decisions affect the bandwidth clustering method, which in turn affects bandwidth reservation.

To define the accurate resource allocation in terms of bandwidth the updated entries of the pheromone table of node  $n$  should satisfy the following:

$$p_b = \sum_i r_{n,s}^i = 1 = \text{full\_BW} \quad (8)$$

Then decision of which path should be followed can be measured as follows:

$$\text{Max}(C_{A_{C1} \rightarrow D_{C1}}) > \text{Max}(C_{A_{C2} \rightarrow D_{C2}}) \quad (9)$$

for which the ideal path ensuring packet transmission with adequate capacity is  $(C_{A_{C1} \rightarrow D_{C1}})$ . This clustering scheme will enable the reduction of lookup tables and generated overhead, thereupon additional queries to neighbors are avoided.

### 2.1.4 Constructed clusters' recreation frequency and triggering

A critical issue arises as to when or whether the clusters should remain unchanged, as well as the duration that each cluster handles (remains in active period). After consecutive network flows it is shown that network is biased overcoming the transient state. Thus paths are marked [15] and ready for the creation of each cluster to host a bounded capacity. The creation criteria are entirely based on path marking of split agents where the

successfully marked path is chosen to be set as a cluster. Beyond the path marking shown in [15] we used cellular automata (CA) to overcome the pathlock [14]. For instance node  $N_A$ , uses CA simply to exchange information with the neighboring nodes examining<sup>2</sup> whether its neighbors have at least a single link leading to the proper destination (that has higher remaining capacity). If these links of  $N_A$ 's neighbor are not empty  $N_A$  remains 'alive' and the cluster is not destroyed. Otherwise the cluster is obliterated.

The issue of when the path should change is critical and has to be taken into account. Triggering with real traffic (swarm based active packets) is an action in order to set different paths in combination with the co-operative agents and CA comparison in each node. Each time a packet is transmitted from a source to a destination, information regarding the cluster of which the packet itself is a part of, is exchanged between the packet and the nodes. This mechanism occurs in a dynamic way (pheromones extraction) exchanging information about the available bandwidth at each time step. Each cluster has a threshold  $S_{cap}$ , and provides an active environment with a proper QoS. Some thresholds for bandwidth should be chosen so that nodes could be dynamically clustered in areas called "blocking areas". Every node in the path is grouped and belongs to the blocking area labeled with  $XMb/s$ , if at least one route with as much free bandwidth between nodes exists. Adaptivity to the different traffic flows which results a significant variation in bandwidth is performed using SART. SART simply enables the wider/distributed view of path availability in order to host traffic based on bandwidth availability. It is undoubtedly true that if the threshold  $S_{cap}$ , where  $S_{cap} < link\_available\_BW$ , is overtaken, meaning that the cluster has reached or passed the overloaded value, then once again the cluster is obliterated and recreation procedure takes place, with agents' contribution and their path marking.

## 2.2 Adaptive resource allocation with reliable traffic flows in constructed clusters

When transmitting data traffic, high reliability is a parameter of main importance; thus extensive retransmission and rate control schemes may be used. On the other hand, when transmitting continuous media (on demand Multimedia streaming) the requirement of on time delivery must be balanced against that of transmission reliability. In our approach we tried to balance both concepts. A freshness degree evaluation and the reliability of links are substantial metrics to ensure reliability for data traffic flow.

Each node measures the number of links at the beginning of each time step  $\tau$  and the number of its

broken links  $\tau_{Tot}$  (if any). It becomes evident that the rate of broken links is equal to:  $\frac{\tau_{Tot}}{\tau_b}$ , where  $\tau_b$  is the

number of node's broken links. One way to evaluate reliability is by using the following notation:

$$\left(1 - \frac{\tau_{Tot}}{\tau_b}\right)^{T \cdot N} \quad (10)$$

where T is the time steps that have passed since the creation of the cluster(i), and N is the number of nodes in the cluster.

According to (11) we have measured the total path reliability notation and the link survivability factor as follows:

$$Reliability = \left(1 - \frac{\tau}{\tau_b}\right)^{T \cdot N + \sum_{i=1}^{hops} \frac{i-1}{Rf}}, \text{ number of hops} > 1 \quad (11)$$

where  $Rf$  is the link capacity refreshment factor and  $i$  is the number of links in the path. In this way we know the reliability degree of each link where we can evaluate the total reliability by using equation (11) above.

## 2.3 Network generated overhead and Quality of Service (QoS)

Many applications are sensitive to the effects of delay, delay variation (jitter) and packet loss. As known, resource and bandwidth reservation generated overhead is the number of control packets that are sent relative to the data packets. In the described hybrid scenario using bandwidth clustering, packets are delivered from one node to another following the predetermined path allocated by agents. The swarm-based scenario does not use specific control packets like other routing schemes since ants pursue the control by being a part of the packets (smart-active packets). Therefore embedded swarm agents map and control the traffic at any time during transmission in the network, and consequently overhead is potentially reduced.

Roughly speaking the large number of control packets introduces more chances of packet collision, longer delay, and more packets dropped resulting in insufficient QoS. The above scheme with embedded swarms allows scalability, and efficient usage of network bandwidth. Additionally no message brokers are needed to fulfill QoS demands issued by applications. As known accurate network state information is very expensive to maintain and monitor.

Another issue that has to be taken into account for resource overhead reduction is that swarm based scheme enables each node to view its absolute neighbor(s), thereupon additional queries to neighbors are avoided. It must be stated out that the inexistence of generated overhead is a result of the non-transmitted routing tables values or other information blocks to neighbors or to all nodes of the network. In other words this scheme significantly reduces the cost of QoS

<sup>2</sup> Examination is performed within routing tables.

prioritized routing. But in essence it increases the efficiency with the no need for packet tunneling and the ‘useless’ need to process packets at layers other than the network layer.

### 3. SIMULATION RESULTS AND DISCUSSION

To demonstrate the design methodologies discussed in this paper, exhaustive simulations were performed to a partially meshed 100-node network. Nodes capacity has been chosen to be relatively high on each node’s buffer (680kb). Network’s performance is examined through a number of various metrics that characterize the efficiency of the proposed scheme. In the implementation-simulation of this work we used our own libraries implemented in C programming language (based in C/Objective C). In the implementation of the bandwidth clustering scheme different traffic input streams have been tested<sup>3,4</sup>. The evaluation took place for different levels of prioritization (marked 1-5(highest)) and ‘don’t care’ packets (no priority at all). The network traffic is modeled by generating constant bit rate (CBR) flows. Each source node transmits one 512-bytes (~4Kbits-light traffic) packet. In the described scenario link capacity is 6Mbps (bi-directional) each. Packet’s requests are routed only once using the saved values on nodes.

In figure 4 the average packet delay with the number of injected packets in the network is illustrated. As the number of packets increases the average packet delay increases slightly. After consecutive simulations for this metric it has been shown that when the number of injected packets reaches 550-600, the average packet delay increases dramatically. This occurs due to capacity limitations that bind each node. Swarm based active scheme shows that in order to reject the incoming packet, data flow scheduler and flow specification fields are responsible to cache locally the packet to node (figure 2). This results in significant delay but negligible compared with the case of packet loss shown in figure 5.

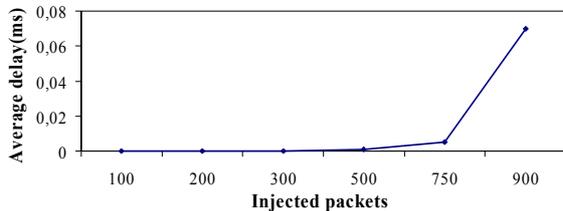


Figure 4: Average packet delay versus the number of injected packets in the network.

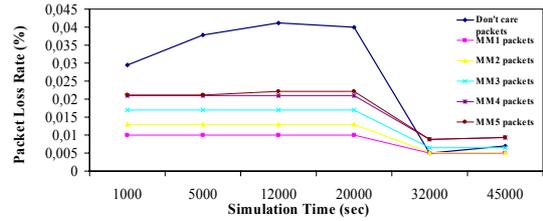


Figure 5: Packet Loss rate for different grade of prioritization (1-5) of MM packets and non prioritized packets (don't care).

Figure 5 illustrates packet loss rate for different grade of prioritization of MM packets as well as for “don’t care” packets. “Don’t care” packets have the highest packet loss rate because no prioritization takes place using the swarm based active scheme. Prioritized packets in real time could be video streams where packet loss and delay could be disastrous for the offered QoS to end users. From figure 5 we can discriminate that the packet loss rate for any of the MM 1-5 prioritized packets is relatively low compared with that of “don’t care” packets. Additionally the prioritized packets MM-4 and MM-5 are behaving almost the same for packets loss rate.

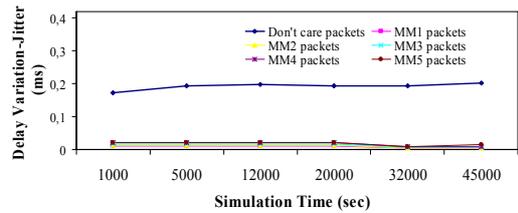


Figure 6: Delay variations (jitter) for different grade of prioritization of MM packets and non prioritized packets (don't care).

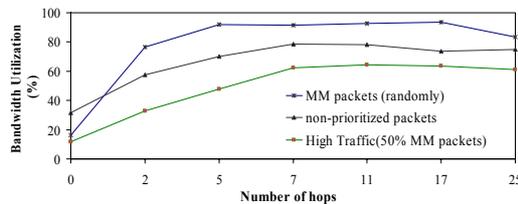


Figure 7: Bandwidth utilization (%) for different traffic flows versus the number of hops.

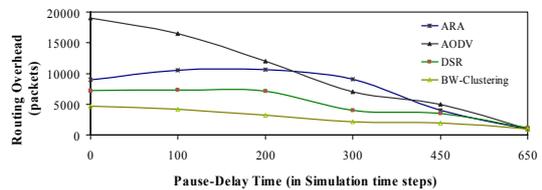


Figure 8: Comparison of different protocols for the generated routing overhead packets.

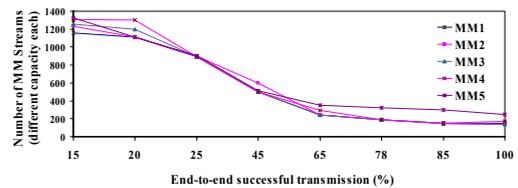


Figure 9: Service tunability for different priority degrees.

<sup>3</sup> Streams included Light Traffic(CBR randomly selected ~4Kbits) and Heavy Traffic(4-Light Traffic)

<sup>4</sup> Multimedia traffic (MM) is prioritized and delay sensitive.

In figure 6 the delay variations (or jitter) for different grade of prioritization of MM packets are shown. Once again the swarm based active scheme enables active prioritization of MM packets which cause significant reduction in the jitter parameter. Jitter is almost the same for low prioritization MM packets (MM-1) and for high prioritization (MM-4, MM-5). Active swarm based scheme prove to be robust in delays where sensitive prioritized packets are scheduled to be transmitted in bounded end-to-end delay.

Figure 7 shows the percentage of bandwidth utilization as the number of hops increases, for MM streams traffic flow versus non-prioritized packets. In figure 7, MM bursty traffic utilizes the available bandwidth reaching 91% of the total available. “Don’t care” packets can not utilize more than 67%. This is caused by the high prioritization of MM packets conflicting with “don’t care packets”. MM packets block “don’t care packets” to reach their destination and balk them from utilizing any link.

In figure 8 we compared the routing overhead generated for 3 different schemes: the generic Ant-Colony-Based Routing Algorithm (ARA), generic Distance-Vector Protocol (DV) and Dynamic Source Routing (DSR). The large number of control packets introduced by other methods enables packet collisions and longer delays which results in insufficient QoS. Significantly less routing overhead is generated in DSR and bandwidth clustering schemes, while the other methods generated doubtless higher overhead packets.

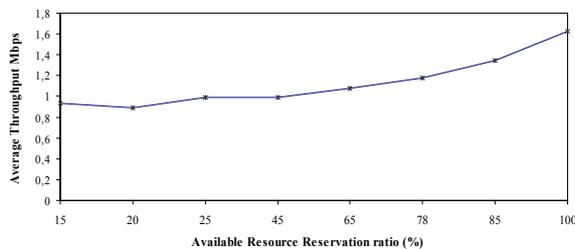


Figure 10: Average throughput as a function of the available resource reservation rate.

Figure 9 shows the service tunability for different priority degrees of MM packets. Each prioritized MM packet has different degree of prioritization to reach a destination. However MM-5 packets behave almost the same as MM-1 packets. This means that the swarm based active scheme enables localized prioritizations of packets to nodes. However nodes are then responsible for scheduling and co-responsible with packets for flow control. Thus when packets are marked as prioritized, swarm based active method “distributes” this prioritization along paths and influences other packets to follow the same path. This is why MM-5 packets behave almost the same as MM-1 packets, while a further examination could be implemented in our future

research for large scale networks where flow rerouting could take place on demand.

Figure 10 illustrates the average throughput response as a function of the available resource reservation rate. When all available resources are reserved, the average throughput response reaches the 1.52Mbps. On the other hand when the available resources are reserved in 15% the average throughput reaches 0.93Mbps. Swarm based active packets are shown to behave better in terms of the average throughput response for high percentage of reserved available resources.

#### 4. CONCLUSIONS AND FURTHER RESEARCH

This work presents a novel method for active swarm based scheme for self-configuration suited for optimized QoS priority routing and the overhead reduction issue. Resources are allocated by using the bandwidth clustering method which is associated with a cooperative learning environment producing a decentralized way capable of adapting quickly to changing capacities. This scheme is applied in a swarm based active network environment where active packets continuously communicate with active nodes by using the SART. This mechanism enables the adaptation of the system to new conditions (bandwidth reservation/capacity allocation), as well as additional information to be passed to neighboring nodes for which information is embodied in transmitted packets. Paths are clustered with re-spect to different levels of bandwidth in order to enable capacity allocation and bandwidth reservation on demand, for any requested traffic. Simulation results reveal that this scheme offers path reliability and enable tunable control in data traffic flow while the same time significantly reduces the generated overhead in the network. Active bandwidth clustering scenario has shown that it can successfully perform optimized bandwidth utilization (avoiding saturated routing or pathlock) and based on the service provided, it successfully offers end-to-end reliability.

Multiclass routing is more than simply putting together routing algorithms designed for individual traffic classes. Scalable interclass resource distribution is essential to achieving high network throughput. Thus this research could be extended for network scalability examination using variants of hybrid-agent based schemes and applied to a Mobile Ad-hoc Network (MANET) where no infrastructure exists. Embedded agents could be entirely responsible for adapting -on demand- the proper resource allocation to the type of service (i.e. voice, multimedia, secure applications for untrustworthy users). Furthermore a scope of interest would be the exploration of different ways in the manipulation of the pheromone quantity.

## 5. REFERENCES

- [1] Chavez, A. Moukas, P. Maes, Challenger: A Multi-agent System for Distributed Resource Allocation, Proceedings of the First International Conference on Autonomous Agents, 1997 pp.323–331.
- [2] D. Tennenhouse, J. Smith, W. Sincoskie, D. Wetherall, and G. Minden, A Survey of Active Network Research. *IEEE Communications Magazine*, 35(1):80-86, January 1997.
- [3] S. Appleby, S. Stewart, Mobile software agents for control in telecommunications networks, *BT Technology Journal*, vol. 12, 1994, pp. 104-113.
- [4] M. Dorigo, V. Maniezzo, A. Coloni, The Ant System: Optimization by a Colony of Cooperating Agents. *IEEE Transactions on Systems, Man, and Cybernetics–Part B*, vol. 26, No. 1, 1996, pp. 29–41.
- [5] E. Guttman, C. Perkins, Service Location Protocol, Version2. RFC2608. <http://www.ietf.org/rfc/rfc2608.txt> (1999)
- [6] R. Schoonderwoerd, O. Holland, J. Bruten, L. Rothkrantz, Ant-based Load Balancing in Telecommunications Networks, *Adaptive Behavior*, vol. 5, 1997, pp. 169-207.
- [7] S. McCanne and S. Floyd, NS network simulator version 2.1b8, June 2001, <http://www-mash.cs.berkeley.edu/ns/>.
- [8] C. X. Mavromoustakis, H. D. Karatza, Bandwidth Clustering for Reliable and Prioritized Network Routing using Split Agent-based method. To appear in the Proceedings of International Workshop on Assurance in Distributed Systems and Networks (ADSN 2005) in conjunction with the 24th International Conference on Distributed Computing Systems Workshops (ICDCS 2005) IEEE Computer Society on Distributed Processing, Columbus, Ohio, USA, June 6, 2005.
- [9] S. Czerwinski, B. Zhao, T. Hodes, A. Joseph, R. Katz, An Architecture for a Secure Service Discovery Service, International Conference on Mobile Computing and Networking, Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking, 1999, Seattle, Washington, United States, pp. 24–35.
- [10] M. Dorigo, L. M. Gambardella, Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem. *IEEE Transactions on Evolutionary Computation*, (1997) 1:53-66.
- [11] W. Adjie-Winoto, E. Schwartz, H. Balakrishnan, J. Lilley, The design and implementation of an Intentional Naming System(INS). Proceedings of the seventeenth ACM symposium on Operating systems principles, Charleston, South Carolina, United States, pp: 186-201, 1999.
- [12] S. Roy, B. Lesieutre, G. Verghese, Resource Allocation in Networks: A Case Study of the Influence Model, Proceedings of the 35th Hawaii International Conference on System Sciences-2002:pp. 220-228.
- [13] L.Zhang S. Berson S. Herzog S. Jamin, Resource Reservation Setup Protocol, official charter maintained by the IETF, 1997-Resource ReSerVation Protocol-Version 1 Functional Specification. RfC: 2205.
- [14] C. X. Mavromoustakis, H. D. Karatza, Split agent-based routing in interconnected networks (SART). *International Journal of Communication Systems (IJCS)*, Wiley, 2004; 17:pp.303-320.
- [15] C. X. Mavromoustakis, H. D. Karatza, An on demand path marking and capacity reservation method using Split Agent. Proceedings of Workshop on Challenges of Large Applications in Distributed Environments (CLADE), in conjunction with the thirteenth IEEE International Symposium on High-Performance Distributed Computing (HPDC-13), Honolulu, Hawaii, 07 June, 2004, pp.128-138.



**CONSTANTINOS X. MAVROMOUSTAKIS**

received a five-year dipl.Eng in Electronic and Computer Engineering from Technical University of Crete and MSc in Telecommunications from University College of London, UK. He is currently a PhD student at Aristotle University of Thessaloniki, Greece, at the department of Informatics. His research interests are in the areas of Multiagent systems, Swarm Intelligence, Epidemic Model for Information Diffusion in wireless networks, Computer network modelling and performance evaluation. His email and web address are <cmavrom@csd.auth.gr> and <agent.csd.auth.gr/~cmavrom>.



**HELEN D. KARATZA**

is an Associate Professor in the Department of Informatics at the Aristotle University of Thessaloniki, Greece. Her research interests include Computer Systems Performance Evaluation, Multiprocessor Scheduling, Mobile Agents, Distributed Systems and Simulation. Her email and web address are <karatza@csd.auth.gr> and <agent.csd.auth.gr/~karatza>.

# Avoid link Breakage in On-Demand Ad-hoc Network Using Packet's Received Time Prediction

Naif Alsharabi, Ya ping Lin, Waleed Rajeh  
College of Computer & Communication  
Hunan University – ChangSha 410082  
Sharabi28@hotmail.com , yplin888@hotmail.com

## KEYWORDS

Ad-hoc Network, AODV, PRT Algorithm, Link breakage.

## ABSTRACT

Most existing on-demand mobile ad hoc network routing protocols continue using a route until a link breaks. During the route reconstruction, packets can be dropped, which will cause significant throughput degradation. In this paper we introduce the Packets Received Time (PRT) to predict the link state, that is, if a particular node is soon to go out of transmission range. The Prediction Algorithm together with our PRT approach enhances the performance of the existing Ad-hoc On-Demand Distance Vector (AODV) protocol. The new approach is compared with original AODV in CBR and TCP traffics using various scenarios. The simulation results showed that our scheme is more efficient, reliable and improves throughput of the Ad-hoc network.

## INTRODUCTION

In most of the current ad-hoc routing protocols like (DSDV, DSR, TORA, DSR etc.) a node will keep using the route until the link is broken. It then has to discover a new route to the destination. During this discovery time the packets are lost and it will cause significant throughput degradation. When the network traffic requires real time delivery (such as voice, or video), dropping data packets at the intermediate nodes can be costly.

In this paper, we propose an algorithm that utilizes PRT (Packet's Received Time) to predict the signal power of the link state and find out if the route is going to break. Our scheme aims at modifying mobile ad-hoc network (MANET) reactive routing protocol (AODV), to give it a proactive behavior to improve its performance. Under our proposed scheme, route maintenance decisions are based on predicted values of 'link-breakage times' (when the next-hop node will move out of transmission range). If a link is about to break, proactive discovery of new routes to all destinations using the next hop node depends on the history of traffic to that destination. In recent years few algorithms and protocols tried to improve performance by using link state information. Associatively-Based Routing (ABR) proposed by Toh [1] favors routes with longer-lived links according to the associatively of the incident nodes. He proposed a link

state prediction model based on the knowledge of mobile nodes' position [1].

Flow Oriented Routing Protocol (FORP) [2] also uses a mobile node's position information that is provided by Global Position System (GPS) to predict link state. The destination node can determine the route expiration time based on the link prediction in the route. It will inform the source node when a link is predicted to be broken, then the source can select the most reliable route to send the rest of the packets.

The more recent preemptive routing protocol proposed by Goof [3] uses signal power strength. It initiates early route discovery by detecting that a link is likely to be broken and builds an alternative route before the link fails. However, their simulation results show that the increase in overhead (The preemptive ratio is used for defining a preemptive zone that is adjacent to the signal strength threshold. Because of is a constant, it implies that for mobile nodes with different relative speeds, the size of the preemptive zone is the same.) could be as high as three times the overhead of the original DSR protocol. The link availability prediction requires that two nodes maintain their movement patterns during the prediction time.

Several papers have proposed the Probability model for the link availability. But the GPS and signal strength methods presented in [4] [5] [6] use physically measured parameters to predict the link status. The node with GPS can know its position directly, but the GPS system currently is not a standard component of mobile devices and the signal is too weak to be received in the metropolitan area and indoors.

This paper concentrates on the PRT prediction approach in ad hoc networks to reduce the data packets that would have been dropped because of link failures. As seen above, in most existing protocols, a mobile host will keep using the route until the link is broken. Our proposed scheme will use power measurement of received packets to predict the topological change in order to rebuild a route prior to the link breakage, thus avoiding the data packets being dropping. Generally, a link failure happens when two mobile nodes A and B

move out of their radio transmission ranges. Node B monitors the packets coming from A, predicts the link breakage time of link A-B, and then sends a warning message to the source node of this active route. The source node can rebuild a new route before the link breaks. The simulation results show that our PRT algorithm can increase the packets delivery ratio and reduce the number of drop's packets due to link failure.

The rest of the paper is organized as follows:

Section 2 discusses the AODV concepts; section three explains our PRT approach, whereas section four gives the simulation method and presents the simulation results. The paper ends with the conclusion and future work to be done in section five.

## 2. AODV CONCEPT

In this section, we present the conceptual details of the AODV protocol. AODV nodes use four types of messages to communicate among each other. Route Request (RREQ) and Route Reply (RREP) messages are used for route discovery. Route Error (RERR) messages and HELLO messages are used for route maintenance. The following sections describe route determination and route maintenance in greater detail.

### 2.1 PRT Route Construction

Our algorithm does not require any modification to the AODV's RREQ (route request) propagation process. When a source needs to initiate a data session to a destination but does not have any route information, it searches a route by flooding a RREQ packet. Each RREQ packet has a unique identifier so that nodes can detect and drop duplicate packets.

An intermediate node upon receiving a non-duplicate RREQ records the previous hop and the source node information in its route table. It then broadcasts the packet or sends back a ROUTE REPLY (RREP) packet to the source if it has a route to the destination. The destination node sends a RREP via the selected route when it receives the first RREQ or subsequent RREQs that traversed a better route (in AODV for instance, fresher or shorter route) than the previously replied route, when the route established the source start send the packet's to the destination through shorter route .

The PRT structure and L\_Prediction are established during RECV (received packet's procedure) and RREP\_ACK phase that we will explain more in section3.

### 2.2 Routing Maintenance and L\_prediction

Data packets are delivered through the primary route unless there is a route disconnection. When PRT detects a packet's received time is bigger than the route life time and does not receive hello packets for a certain period of time the L\_prediction send RERR to the source to initiate a route rediscovery. The reason for

reconstructing a new route is to build a fresh and optimal route that reflects the current network situation and topology. The L\_prediction also mark the disconnect route and delete it from the packet header. Data packets therefore can be delivered through fresh routes and are not dropped when route breaks occur.

### 2.3 Packets Receiving & Packet's Sequence Numbers

The destination node (DN) continues receiving packet's until the link is broken. The DN receives different packet's type from the source node (SN) or upstream node. Each destination (node) maintains a monotonically increasing sequence number, which serves as a logical time at that node. Also every route entry includes a destination sequence number which indicates the "time" at the destination node when the route was created. The protocol uses sequence numbers to ensure that nodes only update routes with "newer" ones.

All RREQ messages include the originator's sequence number and its (latest known) destination sequence number. Nodes receiving the RREQ add/update routes to the originator with the originator sequence number assuming this new number is greater than that of any existing entry. If the node receives an identical RREQ message via another path the originator sequence numbers would be the same, so in this case the node would pick the route with the smaller hop count. If a node receiving the RREQ message has a route to the desired destination then the sequence numbers used to determine whether this route is "fresh enough" to use as a reply to the route request. RREQ messages, RREP messages also include destination sequence numbers. This is so nodes along the route path can update their routing table entries with the latest destination sequence number.

## 3. PRT and prediction algorithm

Two Ray Ground reflection approximations are used as radio propagation model in [9]. The Two Ray Ground model uses formula (1) to calculate signal strength at the receiver's end.

$$\left[ P_r = \frac{P_t * G_r * (h_t * h_r)^2}{d^4} \right] \dots\dots\dots (1)$$

Where:  $P_r$  is the received signal power,  $P_t$  is the transmitted signal power,  $G_t$  is the transmitter antenna gain,  $G_r$  is the receiver antenna gain,  $h_t$  is the transmitter antenna height,  $h_r$  is the receiver antenna height, It is assumed that  $P_t$  is a constant. Assume that the ground is flat to remove dependence of  $h$  and  $d$  values on the geography of the simulation area. So equation above can be simplified under the conditions of ad hoc wireless network simulation to:

$$\left[ P_r = k \frac{P_t}{d^4} \right] \dots\dots\dots (2)$$

Where  $k$  is constant

$$k = G_t * (h_t^2 * h_r^2) \dots\dots\dots (3)$$

This equation shows that the signal power at the receiver node has relation  $\frac{1}{d^4}$  with the distance between the sender node and receiver node.

As we mentioned before, GPS and signal strength methods use physically measured parameters to predict the link status. GPS currently is not a standard component of mobile devices and the signal can be too weak to be received. Supposing the route has already been established and the first packet delivered, our algorithm starts recording packet received times, and based on this data, predicting link breakages using the following formula:

$$R_{pt} = C_t - S_{pt} \dots\dots\dots (4)$$

Where  $R_{pt}$  is the packet received time for the current packet,  $C_t$  is the current time and  $S_{pt}$  is the packet's send time. We suppose to receive 3 packets on destination node to predict the link state. In this case we repeat equation (4) on the future packets and save it in the table using equation (5). In this case we will increment the packet flag  $p\_flag$  by one and save the packet receive time on the receive table as flowing:

$$T_{rpt}[p\_flag] = R_{pt} \dots\dots\dots (5)$$

The packet received time average over all neighborhoods and all time calculated using the following formula:

$$P_{rt} = \sum_{t=0}^T \sum_{n=0}^N \frac{C_t}{T_{rpt}[p\_flag] * LL_t} \dots\dots\dots (6)$$

Where  $P_{rt}$  is packet received Time average on destination node, we are using the  $C_t$  is the current time defined on AODV original protocol calculated during transmitting and receiving packets,  $LL_t$  is link life time,  $T$  total time arrives at the destination, and  $N$  number of hop.

Substituting (6) into equation (2) the received signal power on the distinction node calculated as following:

$$\left[ P_r = k \frac{P_{rt}}{d^4} \right] \dots\dots\dots (7)$$

We added PRT procedure to AODV protocol, in this procedure when the destination node received the first packet, PRT start save packet time on the received packet time table, increment the packet flag, calculate packet signal power and wait for the next packet from the upstream and repeat equation (5), (6), (7) to next packet and compare it with previous packet that is already on the table.

If the current packet's signal power is greater than the pervious packet's signal power, that means the nodes are moving closer to each other otherwise if the current packet signal power is equal to the pervious packet signal power that means the nodes are quiescence so the packet flag will be zero and do not need prediction algorithm.

On the other hand if current packet signal power is weaker than the previous packet signal power, prediction algorithm maintenance marks the current route as idle to delete it from the packet header when a new route is established and send RERR upstream to locally maintain the route, or to the source node to establish RREQ to find a fresh and optimal route to the destination that reflects the current network situation and topology.

In the implementation, each destination nodes will keep an array as showed in (5) of signal info objects. Each table holds three packets with information such as signal power strength and reception time for the same neighboring mobile nodes. When node B receives packets from node A, it updates its table array according to:

$$P_3 \leq P_2 \leq P_1 \quad \text{and} \quad T_3 \leq T_2 \leq T_1 \dots\dots (8)$$

When two mobile nodes are moving closer, the latest signal power strength will be greater than the previous one. In this case, we set  $P_1$  to the latest signal power value and set  $P_2$  and  $P_3$  to zero, no prediction is necessary.

#### 4. Simulation and Analysis

All simulations were run using the NS-2 simulator [13], [14] and numerous simulations were chosen to illustrate the performance advantage gained by using AODV\_PRT over AODV. The simulation experiments can be classified broadly as CBR (UDP) based simulations and simple formal model of TCP over MAC 802.11 based simulations. The routing protocols were tested using both CBR and TCP traffic to get a more complete picture of their performances. Both the CBR and TCP based simulations were run with two mobility models.

The simulations using RW (random waypoint) model were run in a 1500m by 300m area with 20 nodes under varying conditions of mobility and load. The

communication model consisted of 8 CBR connections, with a packet size of 512 bytes for each set of simulations. All statistics were based up on 10,000 data packets and the rate of sending is 0.25, 0.5, 0.75 and 1second. Our simulations were conducted by varying both maximum velocity and pause time. Maximum velocity varied as 1, 4, 8, 12,16, 20 m/s Pause time varied as 0, 50, 100, 150, 200 seconds

Figure 4.1 show that the End-to-End delay for AODV\_PRT has longer delays than AODV for high mobility. The reason for that we measure delays for data packets that survived to the destination. AODV\_PRT delivers more packets, and those packets that are delivered in AODV\_PRT but not in AODV, our explanation when route going to break the AODV\_PRT possibly uses longer alternate paths to deliver packets that are dropped in AODV.

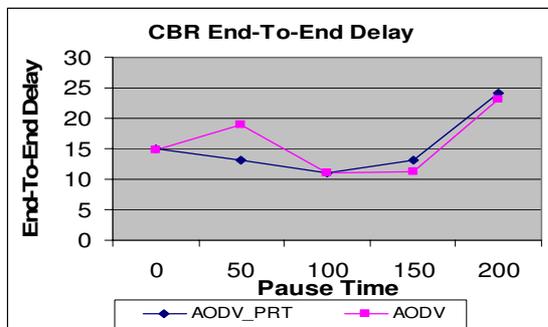


Figure 4.1 End-to-End delays vs. pause time

Figure 4.2 shows that AODV\_PRT delivers more packets on high mobility, when the nodes are moving fast on the topology area, that mean increase the possibility of link breakage. Figure 4.2 shows the AODV with PRT Algorithm will detect the link breakage and delivers those packets lost on AODV protocol.

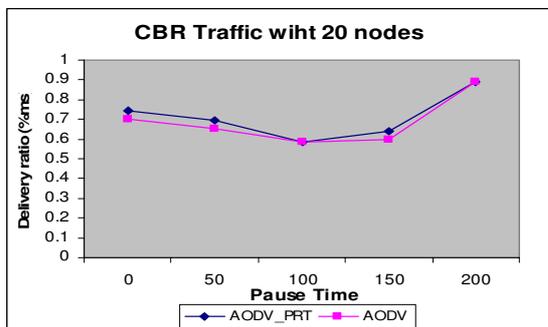


Figure 4.2 Packet delivery ratios vs. pause time

Figure 4.3 shows that the AODV\_PRT losses fewer packets than AODV that's increase the packets delivery thru the AODV\_PRT than AODV.

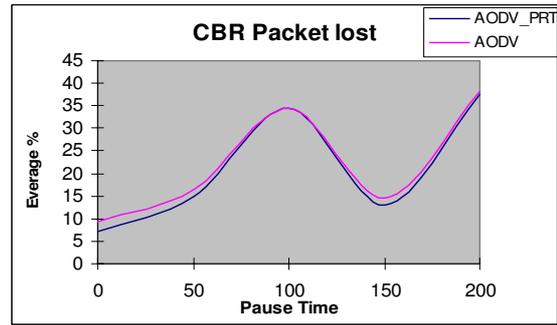


Figure 4.3 CBR packets lost

Figure 4.4 shows that the AODV and AODV\_PRT have delivered almost the same amount of packets in low mobility and when the nodes are moving fast AODV\_PRT delivers more packets. AODV\_PRT has more amounts of control messages. The hop count obtained with CBR traffic is a true measure of the average hop count of all active routes in the simulation, as the traffic source is independent of the network condition, while the hop count obtained with TCP traffic is not. This is because, in the absence of congestion, the rate of TCP transmissions is very sensitive to the number of hops, because the rate depends on the mean round trip time (*rtt*) of each connection, which is largely dependant on the number of hops.

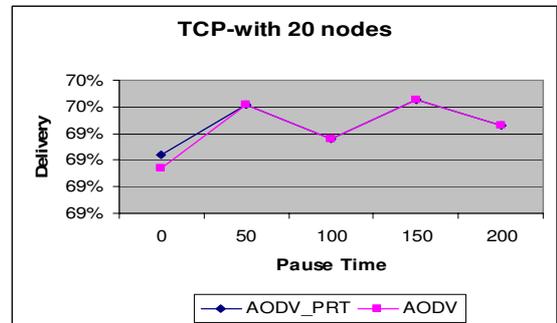


Figure 4.4 TCP Packet delivery ratios vs. pause time

Hence at lower hop counts, TCP transmits at a very high rate, while the rate rapidly drops at higher hop counts. Thus, the average hop count in TCP tends to be similar for all simulations just as the average hop count across all CBR simulations are comparable. Since TCP operates as a feedback system, TCP has a lower average hop count than the average hop count with CBR traffic for the same mobility scenario [6], [7], [8] and [13].

Figure 4.5 shows the delivery packets ratio for TCP && CBR with deferent pause time, figure 4.5 is not compare between TCP and CBR. It shows the packets delivery for each traffic with AODV original and AODV\_PRT.

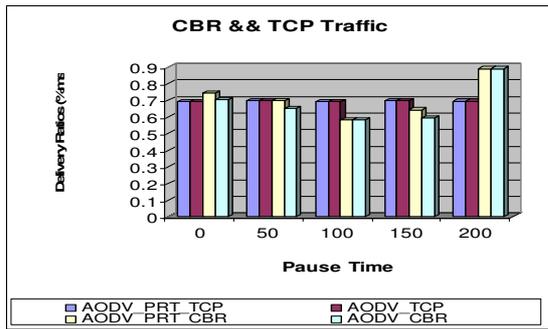


Figure 4.5 Packet delivery ratio with CBR & TCP vs. pause time

Figure 4.6 shows that the End-to-End delay for AODV\_PRT and AODV have the same end to end delay. But AODV\_PRT deliver more packets.

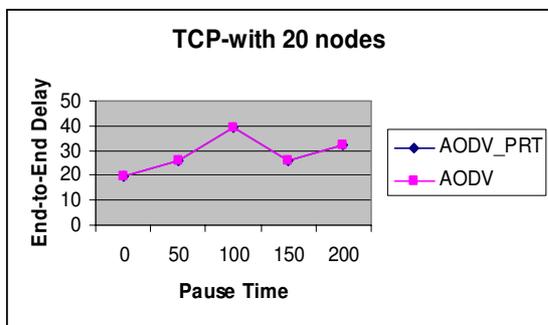


Figure 4.6 TCP End-To-End Delay vs. pause time

Figure 4.7, 4.8 and table [2] are shows that the AODV have More drop packets than AODV\_PRT when no routes because the AODV\_PRT detect the link break and discover a new route to continue send that's packets dropped in AODV, AODV\_PRT more efficient to deliver the data and avoid the packets dropping that's mean its more efficient than AODV.

Table [2] show that the total data dropped and the data dropped because there is no route to the destination during the simulation experiments for the original AODV and AODV\_PRT.

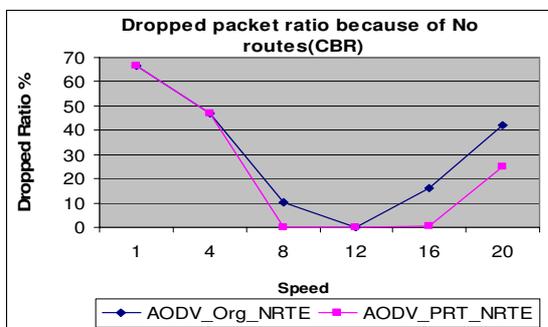


Figure 5.7 Dropped packets on CBR

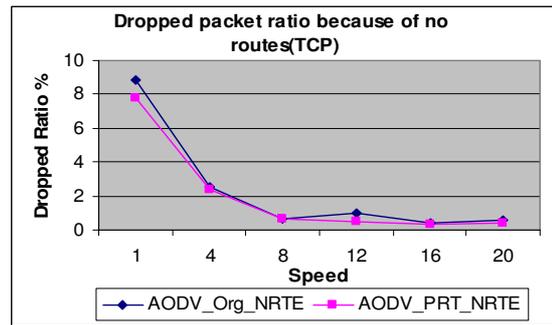


Figure 5.8 Dropped packets on TCP

Protocol/Reason		Speed				
AODV Original		4	8	12	16	20
CBR	Total Dropped Pk	273	277	148	278	376
	Dropped No route	128	28	0	45	159
TCP	Total Dropped Pk	239	2156	402	937	1057
	Dropped No route	6	14	4	4	6
CBR	No route CBR %	46.88 6	10.10 8	0	16.187	42.28 7
	TCP %	2.510 4	0.649 3	0.9950	0.4268	0.567 6
AODV_PRT		4	8	12	16	20
CBR	Total Dropped Pk	273	271	148	332	511
	Dropped No route	128	0	0	1	128
TCP	Total Dropped Pk	255	2156	402	2094	995
	Dropped No route	6	14	2	7	4
CBR	No route CBR %	46.88 6	0	0	0.3012	25.04 8
	TCP %	2.352 9	0.649 3	0.4975	0.3342	0.402 0

Table [2]: Packet Dropped TCP & CBR

## 5. Conclusions and future work

Prediction algorithm is one of the best approaches to avoid link breakage; it has been widely used in schemes aimed at improving performance of ad-hoc networks [4], [5], [6], [7], [9]. As reviewed in this paper, most of this work depends on node density, radio transmission range, and GPS and signal strength. But the GPS and signal strength methods both use physically measured parameters to predict the link status. The performance could be further improved using the received packet signal. This paper has given our new method to improve the performance of ad-hoc network as the following:

- Our simulation compared between CBR traffic OADV original protocol with AODV\_PRT protocol and TCP AODV original protocol with AODV\_PRT protocol. We were not compared

between CBR traffic and TCP traffic performance on simulation results.

- The CBR simulation shows that AODV\_PRT delivers more packets, and those packets that are delivered in AODV\_PRT but not in AODV, take alternate and possibly longer delay more than AODV,
- The TCP simulation shows that AODV and AODV\_PRT have delivered almost the same amount of packets in low mobility, and AODV\_PRT delivers more packets than AODV in high mobility,
- AODV\_PRT has more amounts of control messages. The hop count obtained with CBR traffic is a true measure of the average hop count of all active routes in the simulation, as the traffic source is independent of the network condition, while the hop count obtained with TCP traffic is not,
- Compared with original AODV, our simulation experiments show that our approach with CBR and TCP traffic is more beneficial, delivered more packets, less lost packets and packets dropped.

More still remains to be done to improve the performance on ad-hoc network protocols. Reduce overhead, control messages and also implementing our scheme in the real world scenario.

## References

1. C.-K. Toh. "Associativity-Based Routing for Ad-Hoc Networks", *Wireless Personal Communications Journal, Special Issue on Mobile Networking and Computing Systems*, vol. 4, no. 2, pp. 103-139, March 1997.
2. W. Su and M. Gerla, "IPV6 Flow Handoff in Ad-Hoc Wireless Networks Using Mobility Prediction", *Proc. of IEEE Global Communications Conference, Rio de Janeiro, Brazil*, pp. 271-275, December 1999.
3. T. Goff and N. B. Abu-Ghazaleh etc., "Preemptive Routing in Ad Hoc Networks", *Proc. of the Seventh Annual International Conference on Mobile Computing and Networking, Rome*, pp. 43-52, Italy, July 2001.
4. A. B. McDonald and T. Znati, "A Path Availability Model for Wireless Ad-Hoc Networks", *Proc. of IEEE Wireless Communications and Networking Conference, New Orleans, LA*, pp. 35-40, September 1999.
5. S.M. Jiang, D.J. He and J.Q. Rao, "A Prediction-Based Link Availability Estimation for Mobile Ad-Hoc Networks", *Proc. Of IEEE Infocom, Anchorage, AK*, pp. 1745-1752., April 2001
6. C. E. Perkins. *Ad Hoc Networks*, USA, Addison-Wesley, Dec. 2000.
7. S. R. Das, C. E. Perkins and E. M. Royer, "Performance Comparison of Two On-demand Routing Protocols for Ad Hoc Networks", *Proc. of the IEEE Conference on Computer Communications*.
8. Daehyoung Hong and Stephen Rappaport, "Traffic Models and Performance Analysis for Cellular Mobile Radio Telephone Systems with Prioritized and Non-Prioritized Handoff Procedures", *IEEE Transactions on Vehicular Technology*, Vol. VT-35, No.3, pages 77-92
9. He Dajing, Jiang Shengming and Rao Jianqiang, "A Link Availability PredictionModel for Wireless Ad Hoc Networks", *Proceedings of the International Workshop on Wireless Networks and Mobile Computing, D7-D11, Taipei, Taiwan, April 2000*.
10. B. Narendran, P. Agrawal and D. K. Anvekar, "Minimizing Cellular HandoverFailures without Channel Utilization Loss", *Proceedings of IEEE GlobalCommunications Conference*, pages 1679-1685, vol. 3, December 1994.
11. The Network Simulator - ns-2 <http://www.isi.edu/nsnam/ns>
12. Kevin Fall and Kannan Varahan, editors. *NS Notes and Documentation. The VINT Project*, UC Berkeley, LBL, USC/ISI, and Xerox PARC, November 1997.
13. David A. Maltz, Josh Broch, Jorjeta Jetcheva and David B. Johnson, "The Effects of On-Demand Behavior in Routing Protocols for Multihop Wireless Ad Hoc Networks", *IEEE Journal on Selected Areas in Communications Special Issue on Mobile and Wireless Networks*, pages 1439-1453, August 1999.

# Portable and Scalable Parallel Applications with VCluster

Joohan Lee, Hua Zhang, and Ratan Guha  
School of Computer Science  
University of Central Florida  
Orlando, FL 32816, USA  
E-mails: {jlee, hzhang, guha}@cs.ucf.edu

## KEYWORDS

Middleware, Cluster Computing, Parallel Processing, Message Passing.

## ABSTRACT

Message passing based parallel programming paradigm and associated libraries such as MPI and PVM have proven its novelty and efficiency by successful applications in many diverse areas ranging from scientific computation, simulation, graphics, machine learning, to data mining. Those tools have been de facto parallel programming libraries for cluster computing where a cluster consists of usually homogeneous uni-processor machines. Recent advances in processor technologies made it possible to build a cluster of multi-processor machines more economically. In addition, a new programming language, Java, and its associated technologies opened a door to flourish of new and more efficient development of distributed computing software. Virtual Cluster Computing (VCluster) library is a portable parallel runtime system we have developed to support parallel programs that run on the cluster of heterogeneous uni- and multi-processors. In this paper, we briefly introduce the features of VCluster system and describe how parallel programs can be developed efficiently with VCluster. We also present the experimental results showing the performance of the developed system on two exemplary applications, heat diffusion and back propagation neural network, and compare them with other systems.

## INTRODUCTION

With the availability of high performance microprocessors, high speed networks, and the associated distributed computing tools, cluster computing has become widely used in many diverse areas from fluid dynamics, bioinformatics, simulation, and data mining. Cluster computing provides a low cost parallel computing platform consisting of multiple interconnected PCs or workstations through commodity network technology as a replacement for an expensive parallel computer.

Developing a parallel program over a cluster requires an associated parallel programming model, a language, and a software library to realize that. The most widely used parallel programming model in cluster computing is message passing where each processor executes a

different stream of instructions and exchanges messages when they need to share data or coordinate with other processors. Message Passing Interface (MPI) [Gropp 1995] has been used as a de facto standard for message passing based parallel computing. MPI specifies the necessary point-to-point and advanced collective communication primitives for message passing. MPI and other message passing libraries such as Parallel Virtual Machine (PVM) [Sunderam 1990] and P4 [Butler 1994] have been widely used in developing parallel applications proving its effectiveness due to simplicity and portability over various parallel computing platforms.

However, in cluster computing, lower network speed has been the main bottleneck that impedes the scalability of a cluster. One solution to overcome this obstacle is to improve the bandwidth using faster networking technologies such as Gigabit Ethernet, Myrinet, or ATM. Another solution is to use a cluster of multiprocessors instead of uniprocessor machines. Recently, processor vendors started producing relatively low cost Symmetric Multiprocessor (SMP) machines with usually up to eight processors that share the same memory, I/O devices, and other resources and run different copy of the operating system on each processor. Emergence of low cost SMPs made it possible to build a cluster of multiprocessors economically. With use of multiprocessors, fewer number of machines need to be connected through a slower network and the communication among the processors within the same machine can take place using a much faster bus or an interconnection network. This reduces the amount of communication over the network and improves the scalability of the cluster.

With this configuration, a cluster of multiprocessors, current parallel programming models don't support this new type of cluster computing architecture effectively. With a multiprocessor architecture, shared memory programming models must be incorporated into the programming model while message passing model is still needed for a cluster based parallel computing. A parallel programming model that supports this architecture must address two different parallel computing paradigms in a single framework. In a shared memory multiprocessors, multithreading and synchronization based on shared variables are common techniques. In message passing, communication is defined between two processes rather than threads and explicit message exchanges are used for work

distribution, data sharing, coordination, and synchronization.

While many of the existing parallel programming libraries have been targeted for C/C++ and Fortran programming languages, a new programming language, Java, and its associated technologies opened a door to flourish of more efficient development of distributed computing software due to the platform neutral byte codes, concurrent programming model based on *monitor* concept, object oriented, and inter-process communication mechanisms such as TCP/IP sockets and Remote Method Invocation (RMI). Recently, Java has also strengthened its viability as a distributed computing tool by incorporating Java cryptography and security packages as a part of recent JDKs.

Motivated by these observations, we have developed a new parallel programming model and a parallel runtime system that can address both tightly coupled shared memory multiprocessors and loosely coupled distributed memory multiprocessors, a cluster, in a single framework. VCluster is a prototype realization of the proposed framework. In this paper, we discuss the architecture and features of VCluster and compare them with other related approaches. We present how parallel programs can be developed using VCluster and show experimental results with two parallel applications.

## RELATED RESEARCH

In this section, we discuss the various aspects of high performance cluster computing at different levels and the associated issues. They include communication networks, cluster architecture, parallel programming models and libraries.

In cluster computing, communication latency is the main bottleneck that impedes the scalability of the parallel programs when commodity networking technology such as Fast Ethernet is used. In order to reduce the communication latency, use of faster networking technologies such as ATM network or Myrinet can improve the network performance. However, one critical issue is that they usually use a particular network protocol to utilize the underlying faster network fully rather than using a general TCP/IP protocol, which results in reducing the portability and causing the developed parallel programs to be bound to a specific networking technology.

Recent clusters began to use the Gigabit Ethernet as an alternative networking technology to connect the computers on the cluster. However, the performance of the Gigabit Ethernet doesn't provide much faster networking speed as the name implies. Gigabit Ethernet technology simply increased the network speed to Gigabit per second (Gbps) but was still built on top of the same Ether protocol [Park 1998]. Even with the use of faster networking technologies, it cannot provide enough latency and bandwidth compared with the dedicated interconnection network used in the shared and distributed memory multiprocessors.

As an alternative solution to address scalability, building a cluster using multiprocessors is attractive. It can reduce the communication latency among the processors in the same machine and only the communication between the processors on the different machines needs to take place over the slower network. With the emergence of low cost SMPs and the low cost operating systems running on top of those machines such as LINUX and FreeBSD, SMPs have gained more attention in cluster computing to substitute uni-processor machines. SMP has multiple processors sharing the same system resources. It is symmetric; the processors have the same capability and can access the memory, I/O, and other resources at the same level. And, they are usually connected by a bus with a few processors.

Successful development of parallel programs requires efficient parallel programming models, languages, and libraries. Depending upon the underlying parallel and distributed computing platforms, there can be many different approaches. In a shared memory machine like SMP, multithreading and associated synchronization primitives such as semaphore, mutex, and monitor, are used for developing parallel programs. High level programming languages such as OpenMP provide parallel language constructs that automatically and dynamically create processes and execute some part of the program in parallel such as parallel *for* loops [Chandra 2000].

In cluster computing environment where computers have their own private memories and are loosely connected through a slower network, message passing has been the dominating parallel programming paradigm. Data sharing, coordination, and synchronization are done through explicit message exchanges. In this parlance, MPI and PVM were the most successful implementations and used in developing many different types of parallel applications. Message passing architectures favors coarse-grained parallelism because send and receive operations can be designed to amortize latency over the long messages that are sent infrequently in coarse-grained programs [Jordan 2000].

Implementations of message passing libraries were geared towards a cluster of homogeneous uni-processor machines. Even though multithreading can be used with those message passing libraries, however, communication in the message passing model was defined for inter-process communication and it's up to the users how to incorporate threads into MPI programs. While Scalability and better speedup have been the main driving motivations of parallel and distributed processing, we cannot emphasize the importance of portability and maintainability enough in more recent cluster computing environments where a cluster is composed of heterogeneous machines rather than homogeneous ones. The number representation formats can be different, big endian and little endian. Once again, it can also be handled by the user who is aware of

the heterogeneity and converts the number representations accordingly, however, heterogeneity is not hidden from the users and reduces the portability of the developed parallel programs. Several portable message passing libraries to address these issues based on Java have been developed such as JPVM and mpiJava utilizing the platform neutral features of the Java byte codes [Baker 1998, Ferrari 1998]. mpiJava [7] wraps the MPI library with Java using Java Native Interface (JNI). JPVM [Ferrari 1998] implemented the PVM specifications purely in JAVA.

However, simply applying JAVA technologies to the existing message passing models is not an efficient solution. Those message passing models were designed to augment the existing sequential programming languages with additional message passing functionalities. Message passing model was not designed towards object oriented paradigm. Java is an object oriented programming language and the corresponding message passing model has to take that into consideration in its design to be more efficient.

## VIRTUAL CLUSTER COMPUTING (VCLUSTER) LIBRARY

VCluster is our effort to address the issues raised in the previous section. VCluster provides a framework to support a cluster of heterogeneous uni- or multi-processor machines using a portable parallel programming library. This combines the shared memory programming paradigm and the message passing in a single framework. Initial design of VCluster was inspired by SClib and CRLib that are both multithread message passing libraries with richer set of advanced functionalities such as dynamic load balancing, thread mobility, heterogeneous computing, replication, and fault tolerance [Watts 1998, Lee 2002]. They were primarily developed for parallel C programs over a variety of parallel computing platforms.

VCluster is a parallel run time system implemented using 100% Java codes to support better portability exploiting Java's platform neutral byte code technologies. A basic computing unit in the VCluster is a communicating virtual thread that can migrate across the Java Virtual Machines (JVM) and computers. Each thread has a list of associated named states and named communication channels as shown in Figure 1. Unlike MPI message passing model where communication resources are bound to a process, in VCluster, communication resources are associated with an individual thread. A thread creates a set of uni-directional communication channels to communicate with other threads. A thread also associates itself with a set of states of interest that are shared data and methods. The same state can be shared by multiple threads and the atomic and mutually exclusive access to the states is guaranteed.

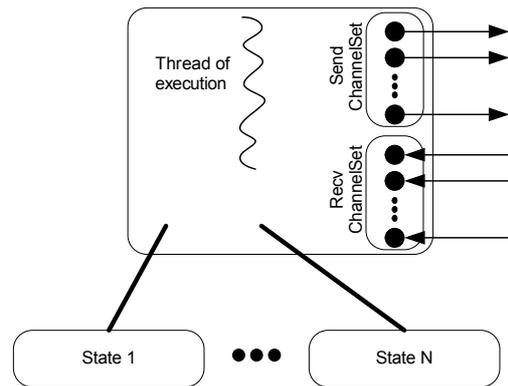


Figure 1. Virtual Thread Structure in VCluster

Java's object oriented paradigm and polymorphism provides useful structure to hide complexity of the software structure and make the parallel program modular and layered software structure more effective. Programmers write the parallel program by extending VCluster thread, state, channel, and channel set classes and inheriting their properties, which hides the internal complexity of those abstract concepts. This architectural features of VCluster make a virtual thread migrate to another machine.

VCluster provides thread to thread message passing functionality. Those communicating threads can be on the same machine or on remote machines. If they are on the same machine, the message will be just copied from one thread buffer to another thread buffer. If they are on the different machines, then the message is actually transmitted over the network. In order to support shared memory programming paradigm without using direct message passing, the threads on the same machine can be bound to the same states and use monitor based coordination and synchronization schemes. In VCluster, a thread is virtual in that the thread can migrate within the system by detaching itself from one computer and move to the remote location with the current status.

Prototype implementation of the proposed VCluster was done using JDK 1.4.2 and used to develop several parallel and distributed applications.

## PARALLEL PROGRAMS WITH VCLUSTER

In this section, we describe how parallel programs are developed using the VCluster library. Figure 2 shows the basic structure of a parallel program. It illustrates how two virtual threads are created, set up the communication links, create states, and exchange messages. Note that two processes on each computer execute the same code but with a different process ID number that tells who is a sender and a receiver. A user creates a virtual thread by extending a VC\_Thread class and needs to implement the run method.

---

```
public class Application
{
    public static void main(String args[])
```

```

{
    VCluster vc;
    VC_ChannelSet vchannelSet;
    VC_Channel vchannel;
    myThread vthr;

    // Initialize
    vc = new VCluster(args);

    // Create two virtual threads and bind them to VCluster
    vthr = new myThread();
    vc.addThread(vthr);

    // Create the states and bind them to the vthreads
    mystate = new myState();
    vthr.addState(mystate);

    // Setup the communication links
    if (vc.getMyPid() == 0) {
        vchannelSet = new VC_ChannelSet("sendChannelSet");
        vchannel = new VC_Channel(1,"ch", vc.VC_WRITE);
    }
    else {
        vchannelSet = new VC_ChannelSet("recvChannelSet");
        vchannel = new VC_Channel(0,"ch", vc.VC_READ);
    }
    // Bind the communication links to virtual threads
    vchannelSet.addChannel(vchannel);
    vthr.addChannelSet(vchannelSet);

    // Start the program
    vc.start();

    // Finalize and exit the program
    vc.finalize();
}

class myThread extends VC_Thread
{
    // User need to implement this run method
    public void run()
    {
        myState mystate;
        VC_ChannelSet vchannelSet;
        int myTid, myPid;

        myPid = getPid();
        myTid = getTid();
        mystate = (myState)getState();

        // Send/Recv a single integer between tow vthreads
        if (myPid == 0) {
            vchannelSet = getChannelSet("sendChannelSet");
            VC_Channel vchannel = vchannelSet.getChannel("ch");
            vchannel.writeInt(888);
        }
        else if (myPid == 1) {
            vchannelSet = getChannelSet("recvChannelSet");
            VC_Channel vchannel = vchannelSet.getChannel("ch");
            int a = vchannel.readInt();
        }

        finalize();
    }
}

```

Figure 2. An Simple Send/Recv Program with VCluster

## EXPERIMENTAL RESULTS

In this section, we examine the performance of the developed system and compare it with other relevant systems. Three relevant systems were compared with VCluster; MPICH, mpiJava, and JPVM. MPICH is a C implementation of MPI by Argonne National Laboratory [MPICH 2004], mpiJava by [Baker 1998], and JPVM by [Ferrari 1998]. MPICH was chosen as the base case that extends the C programming language with message passing functions. It is most widely used and would yield the best performance compared with other Java based libraries. mipJava is simply wrapping the MPICH with Java interface and is expected to be slower than MPICH but faster than JPVM and VCluster that were implemented purely in JAVA.

Our performance analysis was done in two aspects. First we measure the communication overhead. Since JAVA uses the neural byte format for the communication, the communication overhead was expected to be higher than that of C. Fig 3 shows the round trip communication overhead with respect to the varying message size from one byte to one mega bytes. As expected, Java based message passing libraries are slower than C based MPICH. For a larger message sizes closer to one mega bytes, VCluster performed better than the JPVM. We believe that it is because of the VCluster's sending and receiving mechanisms based on separate send/recv threads and polling. Despite the higher communication overhead in VCluster and other Java based systems, it doesn't discourage the use of Java for cluster computing in that coarse grain parallelism is mostly used in cluster computing and the communication overhead can be compensated by the larger computation part of the application.

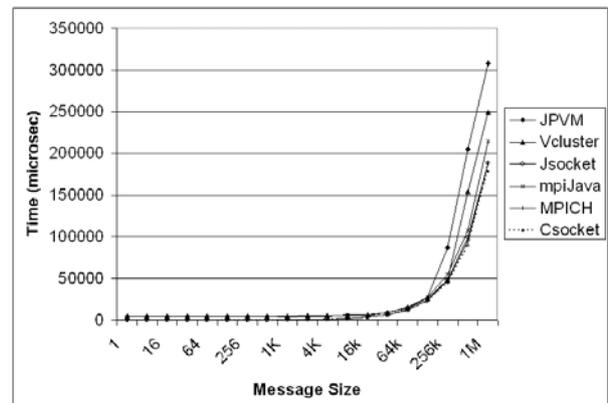


Fig 3. Communication Overhead

Next two experimentations were performed to analyze the real application performance that includes both communication and computation. First application is a

fluid dynamics application, Dirichlet boundary problem. The Dirichlet boundary problem is a simple numerical simulation problem on a two dimensional grid. Each point on the grid has a  $(x, y)$  location and a value representing temperature of some material. At each time step, each point's temperature is averaged with its neighbor's temperatures to find the point's temperature at the end of the time step. This operates for all grid points that are not on the boundary. Boundary grid points are assumed to have a constant value. The workload is uniform in the Dirichlet problem. This allows the domain decomposition technique to be used in dividing up the workload among processes. We have implemented the Dirichlet boundary problem using four message passing libraries introduced above. Our cluster computing environment was composed of a cluster of 32 LINUX PCs equipped with 900MHz AMD Athlon processor, 1 GB memory, and 100BT networking. Figure 4 shows their performance on this cluster with varying number of processors. Performance of three Java based message passing libraries was comparable and the MPICH outperformed them. With a small number of processors, the execution time difference was noticeable, however, with larger number of processors, for example 32, the difference became ignorable. This experimentation results indicate that with enough number of processors Java based message passing libraries can mitigate the associated overhead compared with C based libraries and be an attractive tool for a large scale parallel application development. mpiJava was expected be faster than VCluster and JPVM as it still uses C codes in it, however, its performance was close to the other Java implemented libraries.

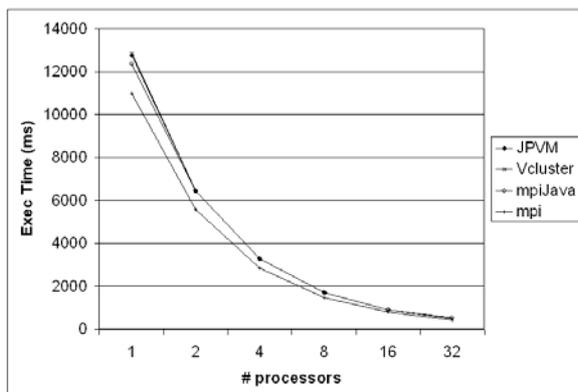


Fig 4. Parallel Dirichlet Problem Performance

The second application was a parallel Back Propagation Neural Network (BPNN) algorithm for network based intrusion detection. BPNN [Rumelhart 1986] is one of the most popular neural network training algorithms and has shown robust performance in many diverse applications. However, computational complexity of the BPNN makes its use challenging especially when the training data set size is huge. BPNN can be parallelized in many different ways depending on the underlying

parallel computing architecture and programming models [Zickenheiner 1994, Nordstrom 1992]. We used a training set partitioning method that reduces the amount of communication needed for distributing data and synchronization. BPNN is trained iteratively until an acceptable mean squared error rate is achieved. In the beginning, the partitions of the data set are distributed to each worker thread. Each iteration is called an epoch. In each epoch, each worker threads work on the fraction of the computation corresponding to the distributed partition and the master thread aggregates the partial computation results. Then, the updated weight vectors are redistributed to all the worker threads for the next epoch.

Figure 5 shows the performance of the parallel PBNN on the same cluster. Once again, three Java based libraries showed the comparable performance. Note that the gap between VCluster and MPICH converges as the number of processors increases. The required amount of computation is much higher than that of Dirichlet boundary problem and it is expected that with use of more processors the gap will further converges as the previous application did.

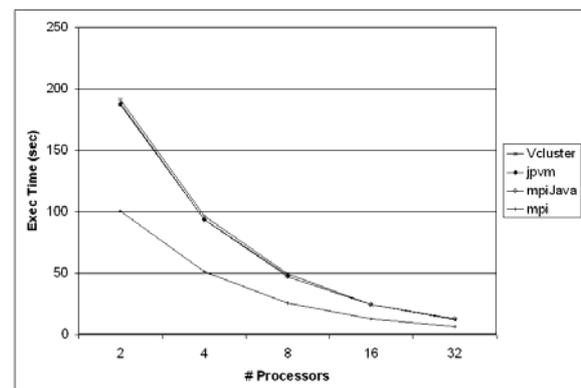


Fig 5. Parallel Neural Network Performance

## CONCLUSION AND FUTURE RESEARCH

In this paper, we have introduced a new portable cluster computing library called VCluster and its features. Preliminary experimentation results showed that the performance of VCluster is comparable to the relevant message passing libraries such as mpiJava and JPVM that are based on Java. However, VCluster provided more convenient and efficient application development environment due to its unique programming model that combines multithreading with communication. Even though Java based parallel programming library has disadvantages with respect to the performance compared with C based implementation, when enough number of processors are available this problem could be mitigated significantly. Currently, we are working on implementing dynamic load balancing using VCluster's thread migration capability.

## ACKNOWLEDGEMENTS

This work was partially supported by ARO under grants DAAD19-01-1-0502,W911NF04110100 and NSF under Grant EIA 0086251. The views and conclusions herein are those of the authors and do not represent the official policies of the funding agencies or the University of Central Florida.

## REFERENCES

- Baker, M., B. Carpenter, S. Ko, and X. Li. 1998. "mpiJava: A Java Interface to MPI", *UK Workshop on Java for High Performance Network Computing* (Sep). Europar.
- Butler, R. and E. Lusk. 1994. "Monitors, Message, and Clusters: The P4 Parallel Programming System", *Parallel Computing*, No. 20, (Apr), 547-564.
- Chandra, R., R. Menon, L. Dagum, D. Kohr, D. Maydan, J. McDonald. 2000. *Parallel Programming in OpenMP*, Morgan Kaufmann.
- Ferrari, A.J. 1998. "JPVM: Network Parallel Computing in Java". *ACM Workshop on Java for High-Performance Network Computing* (Palo Alto, CA, Feb).
- Gropp, W.; E. Lusk; and A. Skjellum. 1995. *Using MPI: Portable Parallel Programming with the Message Passing Interface*. MPI Press.
- Jordan, H.F. and G. Alagband. 2003. *Fundamentals of Parallel Processing*. Prentice Hall.
- Lee, J., S. Chapin, and S. Taylor. 2002. "Computational Resiliency". *Journal of Quality and reliability Engineering International*, No. 18, Vol. 3, 185-199.
- MPICH-A Portable Implementation of MPI. 2004. <http://www-unix.mcs.anl.gov/mpi/mpich>.
- Nordstrom, T. and B. Svensson. 1992. "Using and designing massively parallel computers for artificial neural networks", *Journal of Parallel and Distributed Computing*, Vol. 14, No. 3, 260-285.
- Park, S., J. Lee, and S. Hariri. 1998. "Performance Evaluation of ATM and Gigabit Networks", *IEEE Information Technology Workshop* (Syracuse, NY, August).
- Rumelhart, D. E., G. E. Hinton, and R. J. Williams. 1986. "Learning Representations By Back-propagating Errors", *Nature*, Vol. 323, 533-536.
- Sunderam, V.S. 1990. "PVM: A Framework for Parallel Distributed Computing", *Concurrency: Practice and Experience*, Vol. 2, No. 4, (Dec), 315-339.
- Watts, J., S. Taylor, and S. Nilpanich. 1998. "SCPLib: A Concurrent Programming Library for Programming Heterogeneous Networks of Computers". *IEEE Information Technology Conference* (Syracuse, NY, August). 153-156.
- Zickenheiner, S., M. Wendt, B. Klauer, and K. Waldschmidt. 1994. "Pipelining and parallel training of neural networks on distributed-memory multiprocessors", *IEEE International Conference on Neural Network* (Orlando, FL, June). Vol. 4, No. 27, 2052-2057.

## AUTHOR BIOGRAPHIES

**JOOHAN LEE** is an assistant professor in the School of Computer Science at the University of Central Florida in Orlando, Florida. He received a Ph.d in Computer Science from Syracuse University in 2002. His research interests include parallel and distributed computing, computer and network security, and high performance data mining.

**HUA ZHANG** received a BSc in Computer Science from Shanghai Jiaotong University, Shanghai, China in 1999. He is currently a Ph.d student in the School of Computer Science at the University of Central Florida. His research interests are high performance parallel and distributed computing.

**RATAN GUHA** is a professor at the department of Computer Science, University of Central Florida, with primary interests in Computer Networks, Distributed Computing, Distributed Simulation, and Computer Graphics. He received his Ph.D from the University of Texas at Austin, and his M.Sc. in Applied Mathematics, and B.Sc. with honors in Mathematics from the Calcutta University, India.

# GROUP COMMUNICATION SYSTEM SPECIFICATION AND DESIGN FOR NON-REPLICATED SERVICE

Ruiyong Jia, Yanyuan Zhang, Yong Feng  
School of Computer Science  
Northwestern Polytechnical University  
Mailbox 755, No.127 West Youyi Road, Xi' An City (710072), P.R.China  
E-mail: {jiary,zhangyy,fengyong}@co-think.com

## KEYWORDS

Group Communication System, Storage Area Network, Non-Replicated Service

## ABSTRACT

With the advent and popularization of high-speed storage area network, non-replicated services are applied in more and more high performance distributed storage systems. In this paper, we define the specification of group communication system for non-replicated service and design a novel group communication system GCSLight according to the specification. GCSLight is novel in the following points: (1) it is lightweight in that it removes those complex and extraneous features, such as multicast and virtual synchrony, which exist in many other group communication systems aiming at replicated service. (2) it may obtain optimized performance in failure-free runs by adopting a lazy failure detection protocol. Finally, the preliminary performance of GCSLight is presented.

## 1. INTRODUCTION

Traditionally, server-attached storage (SAS) model dominates in distributed computing systems. Because SAS model limits direct storage sharing among servers, reliability has to be obtained by replicating critical information among multiple clustered servers. In order to support such replicated service, Group Communication System (GCS) middleware (Gregory C. et al. 2001) was advanced. Utilizing the infrastructure, replication-based clustered servers may be built in a two-layers architecture (Y. Amir. 1995):

- Clustered Servers Layer (CSL)

CSL is composed of a group of server processes that deal with application field affairs only, such as serving the query and update requests from clients, balancing the workload among themselves and recovering from possible failure conditions.

- GCS Layer (GCSL)

GCSL underlies CSL and provides the overlying layer with notifications of cluster membership changes, which will trigger the actions of recovery and rebalancing in CSL. Also, some multicast primitives are provided for disseminating messages to total server processes with

required Quality of Service (QoS).

Such a layered architecture has advantages as follows:

- Modularity: Make it possible to separately reason about the guarantees of each layer and the correctness of its implementation.

- Simplicity: With GCS as a toolkit, it will release CSL from those complex tasks, such as detecting various failures and reporting them in a consistent manner, delivering messages to multiple sites with various safety and consistency guarantees. Therefore, the design and implementation of CSL can be simplified greatly.

- Scalability: With the membership service of GCS, CSL will reconfigure automatically, in case of failing, joining, and leaving of cluster members.

In recent years, the advent and popularization of high-speed storage area network has made possible non-replicated clustered servers. The non-replicated service obtains reliability and fails over by a direct take-over manner. For example, suppose that cluster member X is suspected to have crashed, cluster member Y will take over the shared disk of X and do recovery by analyzing the journaling existing on the shared disk. And then Y can take on the work completely charged by X.

The non-replicated service may be applied in many high performance distributed storage systems (J.Menon et al. 2003; Peter J. B. 2004) for shared storage environment. In these systems, a Metadata Server Cluster (MDS Cluster) provides all metadata and locks to clients and the clients access directly the data on shared storage. The size of MDS Cluster is usually very small. For example, a MDS Cluster with ten members is enough for a petabyte-scale distributed storage system (Sage A W. et al 2004).

The non-replicated service has different demands for underlying GCS, compared to replication-based service (R. Golding and O.Rodeh. 2003). In this work, we define a GCS specification suitable for non-replicated service and design a lightweight GCS: GCSLight. It differs from those existing group communication systems in the following points: (1) Simple design. It provides only the necessary GCS functions for non-replicated service. (2) Optimized performance. It eliminates the communication overhead completely in

failure-free runs by employing a lazy failure detection mechanism.

The rest of the paper is structured as follows: Section 2 defines the GCS specification for non-replicated service. Section 3 presents the design and implementation of GCSLight. Preliminary performance of GCSLight is given in section 4. The related work is described in section 5. Section 6 concludes the paper.

## 2. THE SPECIFICATION

Non-replicated service can implement failover by a direct take-over manner. So it is not necessary to replicate information among all cluster members. The only information needing to be kept globally consistent is the group membership. This can be achieved by a strong group membership protocol (Gregory C. et al. 2001) itself.

The group membership problem may be decomposed into two sub-problems: the processor group membership and the server group membership. The first problem is how to achieve agreement on the identity of all correctly functioning processors (or hosts, their incarnations are GCS daemons) that can execute server processes. The latter is how to maintain agreement on the global state of server process group, when server process joins, leaves or crashes.

### 2.1 Assumptions about the Environment

We all processors to use a physical local area network as the communication network; namely, no bridge elements may lie between two processors, such as hubs or routers. The processes communicate by exchanging messages and have access to private hardware clocks whose drift rate are bounded. Neither message delays nor computing speeds can be bounded with certainty. The asynchronous system model allows for the following failures: processes have crash and performance failure semantics.

In addition, we also assume that the system contains a hidden communication channel, namely the storage area network, which allows concurrent access to shared disks by multiple processes.

### 2.2 Failure Detection

Failure detection is fundamental to the adaptation of the processor group membership protocol. It defines the basic means to access availability status of processors and provide inputs to the processor group membership protocol.

We suggest a lazy failure detection protocol for GCS. A processor only checks the availability status of other processors on explicit requests. For example, when a client experiences a communication failure with a cluster member in CSL, it will notify some other available cluster member about the potential failure. At the exact moment, the available cluster member will explicitly require its GCS module to initiate the

processor membership protocol to exclude the processor suspected to have failed. Once one processor is suspected to have failed, it will be deleted from the processor group. This is so-called single site suspicion (M. Hiltunen and R. Schlichting. 1995).

### 2.3 Processor Group Membership Protocol

A processor membership protocol is an agreement protocol for achieving a consistent system-wide view of the operational processors in the presence of member departure, member join and communication failure.

Processor membership protocols could be divided into primary-partition or partitionable membership protocols (Gregory C. et al. 2001). We suggest adopting a partitionable membership protocol, which allows the existence of multiple parallel processor groups at the same point in read-time as a sequence of network partition. It gives applications developers the flexibility of determining how to react when network partitions.

Formally, the processor group membership protocol of GCSLight has properties as follows:

Definitions:

joined (p,g) : holds true if processor p joins group g

LVp (g) : the local view of p about the members of group g.

Historyp : an infinite sequence of LV {LVp1, LVp2, ..., LVpk, ...}

Historyp  $\equiv$  Historyq :

$$\forall j (LVpj = LVqj) \vee (LVpj \cap LVqj = \Phi)$$

(Property 1) Self-Inclusion

If a processor p joins group g, then p is a member of g.

Formally: joined (p,g)  $\rightarrow$  p  $\in$  LVp (g).

(Property 2) Agreement on Group Membership

If processor p and q joins the same group g, both p and q see the same members in the g. Formally:

$$\text{joined (p,g)} \wedge \text{joined (q,g)} \rightarrow \text{LVp (g)} = \text{LVq (g)}$$

(Property 3) Agreement on Group History

All processors have the same history. Formally:

$$\forall p,q (\text{Historyp} \equiv \text{Historyq})$$

## 3. IMPLEMENTATION

### 3.1 Architecture and Interfaces

We implement a GCS (GCSLight) for non-replicated service according to the specification. The GCSLight is based on a daemon-client model where generally long-running daemons provide membership services to clustered servers in CSL. Servers linked with a small library GL\_Lib must connect to the daemon resided on the same processor in order to gain access to the services of GCSLight.

The daemon architecture of GCSlight is presented in the Figure 1. The implementation of GCSlight is inspired by Spread (Yair A. and Jonathan. 1998) and Transis (D. Dolev and D. Malki. 1996).

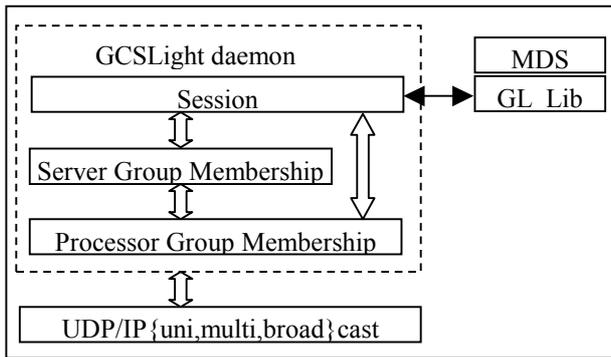


Figure 1: GCSLight Architecture

The GL\_Lib provides the entire client interfaces as depicted in the Figure 2. The connection between the GL\_Lib and the daemon is implemented by IPC mechanism. The Session and Server Group Membership modules manage user connections, server group memberships, and translate processor group membership changes into server group membership changes.

When one server fails, other user application components may call GL\_viewcheck() to update the server group membership consistently. GL\_viewrecv() may be used to receive the changes of server group membership.

```

GL_connect (const char* GCSLight_name,
            mailbox * mbox);
GL_disconnect (mailbox * mbox);
GL_join (mailbox * mbox, char * groupname);
GL_leave (mailbox * mbox, char * groupname);
GL_viewcheck (mailbox * mbox, char * error_report);
GL_viewrecv (mailbox * mbox, int max_mess_len,
            char mess*);
GL_error (int error);

```

Figure 2: GCSLight Application Programming Interfaces

### 3.2 Algorithm for Processor Group Membership

The processor group membership algorithm is based on jahanian's work (Jahanian F. et al. 1993). What different from jahanian's work is that we adopt a lazy failure detection mechanism in the protocol. The updating of new membership is done by a 2-phase protocol: Firstly, the leader of the group sends a "NEW\_MEMBERSHIP" message to all other members; Secondly, the members except the leader will acknowledge the message; Finally, after collecting all "ACK" messages, the leader will send a "COMMIT" messages to all members. The leader of the processor group is the processor, which creates the group. The algorithm is intuitively depicted as follows:

#### (1) Handling joins

When a processor starts, it will broadcast a message "JOIN\_CLUSTER" to all possible processors in the network. If the processor can't discover any other active cluster nodes, it will assume that it is the first processor

to start and other processors will join later, and form a singleton group and set itself as the leader. Otherwise, the leader of the existing cluster will accept the new comer and update the membership change to all cluster members including the new comer.

#### (2) Handling leaves

When a processor leaves the cluster normally, it will broadcast a "LEAVE\_CLUSTER" message. If the processor is a leader, the rest of the cluster will elect a new leader and reform the cluster. Otherwise, the leader of the cluster will update the membership change to all other cluster members.

#### (3) Handling faults

The possible faults of a processor include performance failure (too slow), crash failure or communication failure (message omit or network partition). When one of these failures occurs, the fault will be reported to the corresponding GCSLight module. Then GCSLight will initiate the agreement protocol about the processor group membership. If the processor suspected to have failed is a leader, the rest of the cluster will elect a new leader and reform the cluster. Otherwise, the leader of the cluster will update the membership change to all other cluster members.

Because all processors agree on the sequence in which they join the group, the new leader may be selected locally when the current leader fails. Namely, the processor next to the current leader in the processor group membership is the leader candidate.

The pseudo-codes of the above algorithm are given in the appendix.

### 3.3 Algorithm for Server Group Membership

GCSLight provides the method of managing multiple servers like MDSes as a logical server group. The object of the server group membership protocol is to delivery the server group membership changes consistently to servers, whenever a server joins, leaves or fails.

The server group membership algorithm is triggered when one of the following events appears:

(1) User applications call GL\_join, GL\_leave;

(2) The processor group membership changes;

(3) Servers crashes. GCSLight monitors the status of servers by the services provided by operating system. For example, GCSLight will poll the socket (select()) for possible data, and then call recv() to receive the data. If recv() return 0, this will indicate a crash error of server.

The new server group membership is also promulgated in a 2-phase protocol alike in the processor group membership protocol.

## 4. PRELIMINARY PERFORMANCE

For non-replicated service oriented GCS like GCSLight, the performance metrics only include those of processor group membership protocol. The main metrics of processor group membership protocol (Oliver S. and Flavin C. 1998) are:

- (1) Stability: the number of spurious membership changes per time unit;
- (2) Join processing time: the time between the start of a server  $p$  and the moment a new group that includes  $p$  is formed;
- (3) Failure detection time: the time between the crash or the disconnection of a member server  $p$  and the moment a new group that exclude  $p$  from its membership is formed.

Because the processor group membership protocol of GCSLight is based on lazy failure detection, the suspected failures are just reported by CSL. GCSLight itself will not cause spurious membership changes. Therefore, the stability of GCSLight is dependant on foreign failure detector, so does failure detection time. We only test the join processing time of GCSLight.

The tests were conducted on 10 Pentium II 350Mhz workstations interconnected by a 10 Megabit/sec Ethernet local area network. The workstations are equipped with Windows 2000 Server as operating system. Our measurements approximate the join processing time as the time between the broadcast of the "JOIN\_CLUSTER" message by  $p$  and the moment  $p$  receives the "COMMIT" message.

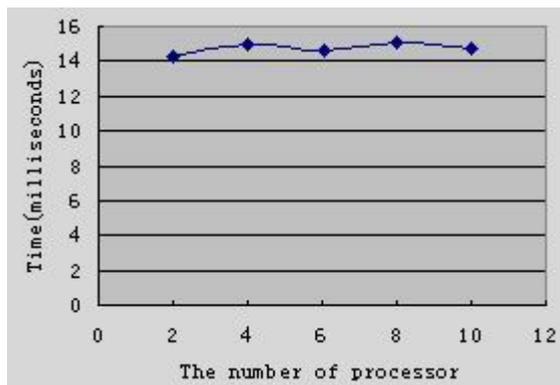


Figure 3: Join Processing Time

Because of the broadcast capability of Ethernet, the "JOIN\_CLUSTER" message and "COMMIT" message are delivered to all processors at virtual the same time. So we can see that the joining processing time is almost constant independent of the number of processor.

The test of GCSLight is just preliminary. In further work, we will compare the joining processing time of GCSLight with those of other GCSes in the same test conditions.

## 5. RELATED OUR WORK TO OTHERS

Most of the existing GCSes were designed toward the end of supporting replicated service. Some of the leading GCSes are: ISIS (Birman, K. P. 1986), Phoenix (Malloth, C. P. et al. 1995), Transis (D. Dolev and D. Malki. 1996), Spread (Yair A. and Jonathan. 1998), etc. we call these GCSes replicated service oriented GCSes.

Replicated service oriented GCSes also provide multicast services and the programming model of virtual synchrony besides membership service (Y. Amir. 1995). The multicast services have different levels of

ordering and reliability, such Safe Delivery, FIFO Order, Causal Order, Total Order. Virtual Synchrony orders membership messages with regard to application messages, so that all active members observe a same messages flow. Both multicast services and virtual synchrony are for ordering and reliability of replication. Therefore, they are redundant for non-replicated service.

To the best of our knowledge, the processor group membership protocols of existing GCSes are based on positive failure detection. For example, a processor sends "are-you-live" messages to other processors periodically and other processors acknowledge the messages. When a processor  $p$  misses a certain number of "ack" messages from  $q$ ,  $p$  will suspect that  $q$  has failed. The shortage of the method is its messages overhead even in the failure-free runs.

We suggest a lazy failure detection-based processor group membership protocol in GCSLight. Lazy failure detection may eliminate the message overhead completely in failure-free runs. Considering that GCSLight is designed to support small clustered servers running on a physical network and the server platforms tend to become more and more reliable today, the failure conditions will appear rarely. Therefore, we argue that lazy failure detection will lead to better average performance.

## 6. CONCLUSION

In this paper, we define the specification of GCS for non-replicated service and design a novel group communication system GCSLight according to the specification. GCSLight is novel in the following points: (1) it provides only the necessary GCS functions for non-replicated service. So it is very lightweight. (2) by adopting a lazy failure detection-based processor group membership protocol, GCSLight itself does not incur any communication overhead in failure-free runs.

Although non-replicated service can also be built on some existing GCSes, like Transis and Spread, these systems are very complicated and hard to understand. Through the practice of designing and implementing the GCSLight, we think that it is worthy of paying out the price of developing a new, yet very simple and efficient GCS, when considering the cost of maintain and testing a very complex GCS with many unrelated features.

## REFERENCES

- A. Bartoli. 2004. "Implementing a Replicated Service with Group Communication", *Journal of Systems Architecture*, Elsevier, Volume 50, Issue 8(Aug), 45-519.
- Birman, K. P. 1986. "ISIS: A System for Fault-Tolerant Distributed Computing". Technical Report TR86-744 (Apr), Cornell University, Department of Computer Science.
- D. Dolev and D. Malki. 1996. "The Transis approach to high availability cluster communication", *Communications of the ACM* 39, No.4 (Apr), 64-70.
- Gregory C. et al. 2001. "Group Communication Specification: A Comprehensive Study", *ACM Computing Surveys* 33,

No.4 (Dec),1-43.

- Jahanian F. et al. 1993. "Processor Group Membership Protocols: Specification, Design and Implementation ". In *Proceedings of the IEEE Symposium on Reliable and Distributed Systems (Oct)*.
- J.Menon et al. 2003. "IBM Storage Tank—A heterogeneous scalable SAN file system". *IBM Systems Journal*, Volume 42, Number 2.
- M. Hiltunen and R. Schlichting. 1995. "Properties of membership services", In *IEEE International Symposium on Autonomous Decentralized Systems (Apr)*.
- Malloth, C. P. et al. 1995. "Phoenix: A toolkit for building fault-tolerant, distributed applications in large scale". In *Workshop on Parallel and Distributed Platforms in Industrial Products (Oct)*.
- Oliver S. and Flavin C. 1998. "Evaluating the Performance of Group Membership Protocols". In *Proceeding of Fourth International Conference on Engineering Complex Computer Systems (Aug)*, Monterey, California
- P. Felber et al. 1999. "Failure detectors as first class objects". In *Proceeding of 1st IEEE Intl. Symp.on Distributed Objects and Applications (DOA'99)*, Edinburgh, Scotland, Sep., 132–141.
- Peter J. B. 2004. "The Lustre Storage Architecture". <http://www.lustre.org/docs/lustre.pdf>, 2004.
- R. Golding and O.Rodeh. 2003. "Group Communication still complex after all these years". <http://www.haifa.il.ibm.com/projects/storage/zFS/public.html>.
- Sage A W. et al 2004. "Intelligent Metadata Management for a Petabyte-scale File System". *Second Intelligent Storage Workshop*, University of Minnesota (Apr).
- Yair A. and Jonathan. 1998. "The Spread Wide Area Group Communication System", Technical Report 98-4, Center for Networking and Distributed Systems, Johns Hopkins University (Jul).
- Y. Amir. 1995. "Replication Using Group Communication Over a Partitioned Network", PhD thesis, Institute of Computer Science, The Hebrew University of Jerusalem, Israel.

### Appendix: The Pseudo-Codes of Algorithm in Section 3.2

*/\* variable definition \*/*

**myid**: the processor id. It may be retrieved from the stable storage, its instance number is incremented by one after every restart.

**CLV**: current local view of a processor, which is a set of processor ids. Initially  $\Phi$ ;

**SLV**: suggested new local view by the processor group membership protocol. Initially  $\Phi$ ;

**number**: the sequence number in the CLV or SLV.

**membership**: Boolean initially false; holds true, if the processor group membership is in execution.

**protocol\_stage**: when waiting for COMMIT message, set it to WAIT\_COMMIT; Else, IDLE;

*/\* initialization \*/*

```
myid←getmyid(); myid←myid+1; setmyid();  
broadcast (joinMes<JOIN_CLUSTER,myid>);  
set join_timer;
```

*/\*main loop \*/*

loop

```
wait (message or timer)  
switch (message.type)  
  case JOIN_CLUSTER:  
    do_member_join(&message);  
    membership←true;  
  case LEAVE_CLUSTER:  
  case VIEW_CHECK:  
    do_member_leave(&message);  
    membership←true;  
  case PREPARE_COMMIT:  
    do_ack (&message);  
  case COMMIT:  
    do_delivery (&message);  
    membership←false;  
  case ACK:  
    do_ack_check(&message) ;  
    .....  
endswitch  
if Join_Timer.expire  
  CLV←CLV ∪ {myid};  
  number←get_number (CLV);  
  Deliver CLV to the layer of server group  
  membership.  
fi  
if Wait_PreCommit_Timer.expire  
  SLV←CLV- {processor ids whose number is larger  
  than that of myid.};  
  broadcast(PreCommitMesM  
    <NEW_MEMBERSHIP,myid,SLV>);  
  set Collect_Ack_timer(timeout_value);  
fi  
if Collect_Ack_timer.expire  
  SLV←CLV-{those members which do not send  
  ACK message};  
  broadcast(PreCommitMesM  
    <NEW_MEMBERSHIP,myid,SLV>);  
  set Collect_Ack_timer(timeout_value);  
fi  
if Wait_Commit_Timer.expire /*leader dies*/  
  SLV←CLV-{ processor ids whose number is larger  
  than that of myid};  
  Broadcast(PreCommitMes  
    <NEW_MEMBERSHIP,myid,SLV>);  
end loop  
  
procedure do_member_join(&message)≡  
  if membership is false  
    SLV←CLV ∪ {message.sender};  
    if number =1 /*is I am leader*/  
      if message.sender is not in CLV  
        broadcast(PreCommitMes  
          <NEW_MEMBERSHIP,myid,SLV>);  
        set Collect_Ack_timer(timeout_value);  
      fi  
    else  
      set Wait_PreCommit_Timer (  
        number*timeout_value);  
    fi  
  fi
```

## AUTHOR BIOGRAPHIES

```
procedure do_member_leave(&message) ≡
  if membership is false
    if number = 1
      if message.failed_member is in CLV
        SLV ← CLV - {message.failed_member};
        broadcast(PreCommitMes
          <NEW_MEMBERSHIP,myid,LV>);
      fi
    else
      set Wait_PreCommit_timer (
        number*timeout_value);
    fi
  fi
```

```
procedure do_ack (&message) ≡
  if myid is in message.SLV
    if message.sender is the leader in CLV
      unicast(AckMes<ACK,myid>);
      protocol_stage ← WAIT_COMMIT;
      unset Wait_PreCommit_Timer;
      set Wait_Commit_Timer(
        number*timeout_value);
    else if protocol_stage is WAIT_COMMIT
      unicast(AckMes<ACK,myid>);
      number ← get_number(message.SLV);
      set Wait_Commit_Timer(
        number*timeout_value);
    fi
  fi
```

```
procedure do_ack_check(&message) ≡
  if receive all "ACK" messages from every member
    broadcast (CommitMes<COMMIT,myid>);
    unset Collect_Ack_timer;
  fi
```

```
procedure do_delivery (&message) ≡
/*according to the change of processor group
membership, check and report the server group
membership to user applications.*/
  if protocol_stage is WAIT_COMMIT
    unset Wait_Commit_Timer.
  CLV ← SLV; number ← get_number (CLV);
  protocol_stage ← IDLE;
  Deliver CLV to the layer of server group
  membership;
  Fi
```



Ruiyong Jia was born in 1976. He received the B.S. Degree in Computer Science from Henan University, China in 1999 and the M.S. Degree in Computer Science from Northwestern Polytechnical University, China in 2002. He is currently a Doctor Candidate at School of Computer

Science, Northwestern Polytechnical University. His research interests include distributed algorithms and fault tolerant systems, distributed storage systems.



Yanyuan Zhang was born in 1954. He is a Full Professor and a head of Software Engineering Institute in School of Computer Science, Northwestern Polytechnical University. He received his Master Degree from Northwestern Polytechnical University in 1987 with specialty of computer

He is a member of the Committee of Information Storage Technology, China Computer Federation. He has published four books, over 20 Journal papers. His research interests cover software engineering, storage management, fault tolerance, and distributed parallel processing.



Yong Feng was born in 1977. He is a Doctor Candidate at School of Computer Science, Northwestern Polytechnical University. He received his Master Degree from Northwestern Polytechnical University in 2002 with specialty of computer software theory.

His research interests are in high performance network storage system and parallel I/O, fault tolerance, parallel and distributed computing, performance evaluation and benchmark.

# EVALUATING PERFORMANCE OF DISTRIBUTED COMPUTING TECHNOLOGIES – HLA AND TSPACES ON A CLUSTER COMPUTER

Ratan Guha, Jooan Lee, Oleg Kachirski  
School of Computer Science  
University of Central Florida  
Orlando, FL 32816, USA  
E-mail: {guha, jlee, oleg}@cs.ucf.edu

## ABSTRACT

Parallel and distributed simulation is concerned with issues introduced by distributing the execution of a discrete event simulation program over multiple computers. For increased performance, a tightly-coupled parallel multiprocessor computing platform is used, containing multiple CPUs communicating over a high-speed interconnection network. Examples include parallel computers. While hardware specifications improve at exponential pace, the software technologies for parallelizing problems have remained relatively unchanged, posing problems due to rapidly increasing data volumes. Message passing has been the most widely used method to parallelize data mining algorithms. In this paper we investigate the performance of two widely used distributed computing technologies, High Level Architecture (HLA) and Tuple Spaces, used to solve the problem of an intrusion detection system utilizing parallel backpropagation neural network (BPNN) for a data mining process in a space-parallel approach.

## INTRODUCTION

Computer simulations have become a useful part of modeling many natural systems in physics, chemistry and biology, human systems in economics and social science and in the process of engineering new technology, to gain insight into the operation of those systems. They have been used in a wide variety of applications, simulating very large-scale systems. Quickly, the processing power of a single computer system became inadequate for certain problems of global scale; and use of super-computer wasn't always an option for many researchers. With the advent of computer networks, and later on computing clusters, the field of parallel and distributed simulation has received a widespread recognition. Beowulf cluster implementation has raised parallel and distributed computing to a new level. Cluster computers were first introduced in 1994 as a result of the Beowulf Project at CESDIS. A Beowulf system is a collection of personal computers constructed from commodity-off-the-shelf hardware components interconnected with a local-area network and configured to operate as a single unit, a parallel computing platform, using an open-source network operating

system (e.g., Linux) (Beowulf Project). The driving design philosophy of a Beowulf system is to achieve the best possible price/performance ratio for a given computing problem. For many problems it's possible to achieve an order of magnitude improvement in price/performance compared with "conventional" parallel supercomputer designs. Cluster computing is used in this paper as a hardware platform for comparing two distributed computing technologies – DMSO's High Level Architecture/Run-Time Infrastructure, and IBM's TupleSpaces.

One of the most recent developments in distributed simulation technologies is called High Level Architecture (HLA), driven by DARPA, which has become a standardized framework (IEEE 1516) for modeling and simulation. HLA is a component-based software architecture that incorporates subsystems for communication and data sharing, synchronization, and time management (Reid 2000). This framework facilitates distributed and multi-platform computing in simulation systems by providing standard integration architecture for separate and remote applications, thus facilitating reuse of simulation components. Simulation objects have to be compliant with the data definition for the simulation, which is described in the Object Model Template. The data shared between the federates within the federation, is defined by the Federation Object Model. Simulation Object Model defines the data that federates share with the federation. The common RTI provides the following services to the federation: federation management, declaration management, ownership management, object management, time management and data distribution management. The flow of simulation and its architecture, as well as providing data processing and meaningful presentation and interpretation of results is still a task of the simulation developer. HLA provides specifications for seamless integration of various simulation components; however, the efficiency of the entire simulation heavily depends on implementation and operational performance of each individual simulator comprising the simulation. At the network level, RTI utilizes reliable multicast protocols to implement inter-federate data exchange.

IBM's TupleSpaces (TSpaces) is a set of network communication buffers called tuple spaces and a set of

APIs for accessing those buffers (Wyckoff 1998). TSpaces allows heterogeneous, Java-enabled devices to exchange data with little programming effort. The package includes server software that implements the buffers and client software for accessing the buffers. The TSpaces server is composed of two main layers. The bottom layer comprises the basic tuple management. This is where tuple sets are stored, updated, indexed, and scanned. The interface to this layer is the Tuple Management API. The top layer comprises the operator component, which is responsible for operator registration and handling, implementation, and management. TSpaces provides group communication services, database services, URL-based file transfer services, and event notification services. With its small footprint, it is ideal for bringing network services to small and embedded systems. TSpaces emulates a shared-memory multiprocessor architecture and reduces the complexity of writing parallel programs accessing shared data. This comes at the expense of message-passing performance, especially in the cluster computer environment, where the data is not physically shared among the processors.

The rest of the paper is organized as follows. First, we highlight the concepts of intrusion detection systems and introduce the problem of parallel training of the neural network classifier for certain intrusion detection systems. We discuss the parallel implementation of the problem using High Level Architecture and TSpaces. Then, we compare the performance of these technologies by executing each implementation on a cluster of computers. Finally, a summary is presented.

## **INTRUSION DETECTION SYSTEMS**

In the recent years we have seen tremendous developments in the field of computer networks. Advances in networking technologies, algorithms and protocols made it possible to think of a network of computers as an information processing unit. With these advances came fundamental questions concerning the security of such systems. It's quite easy to construct a secure centralized system – providing physical security of the equipment, assigning simple admission procedures and authentication mechanism, and providing operating system support for process-level security. Making a computer network secure is much more problematic. Much of the equipment isn't in secure locations, and most of the communication goes over insecure data links.

We should accept the fact that there will be no absolutely secure system for a while, and prepare for system attacks, detecting them as soon as possible and taking appropriate action. And this, in essence, is the task of an Intrusion Detection System (IDS). One of the most utilized ways of detecting intrusions is by analyzing the audit data, generated by system and

application processes. An audit trail is a log of activities on a system that are recorded in a file as they occur; for example – dump of network packets received by a system. The logging process generates very large data files (depending on the system, log files can be hundreds of megabytes in size, e.g. (Lippmann 2001), which are difficult to analyze manually. IDS then automates the task of analyzing the audit trail and pinpointing any suspicious activity (anomalies) which can be classified as a probable intrusion. Using vast resources offered by the distributed systems allows speeding up analysis process, as discussed in the next section. Anomaly Detection model makes use of operational profiles, which represent normal system activity. During the training period, regular system activities are recorded and a normal system behavior profile is established. During system operation, any deviations from a normal profile are considered anomalous, and an alert is raised.

Neural networks offer alternative means of maintaining a model of expected normal user behavior. They offer a more efficient, less complex, and better performing model than statistical models of system and user behavior (Lee and Siddiqui 2004). Neural network techniques may be found to be more efficient and less computationally intensive than conventional rule-based systems. A lengthy, careful training phase is required with skilled monitoring. After the training period, the network tries to match actual commands with the actual user profile already present in the net. Any incorrectly predicted events actually measure the deviation of the user from the established profile.

## **KNOWLEDGE DISCOVERY PROBLEM FOR INTRUSION DETECTION**

As discussed in the previous section, the problem of data mining within large datasets places high demand on computational resources. This makes it a viable problem to measure the performance of a distributed system. Specific application is considered in this paper – a backpropagation neural network training algorithm for an intrusion detection system (BPNN). Given a large initial training data file, the intrusion detector learning task is to build a predictive model (i.e. a classifier) capable of distinguishing between "bad" connections, called intrusions or attacks, and "good" normal connections. A standard set of data to be audited, which includes a wide variety of intrusions simulated in a military network environment, was provided by Lincoln Labs (Lippmann 2001) in the form of a network trace file incorporating known labeled attacks. A connection is a sequence of TCP packets starting and ending at some well defined times, between which data flows to and from a source IP address to a target IP address under some well defined protocol. Each connection is labeled as either normal, or as an attack, with exactly one specific attack type (KDD Data 1999).

The training process proceeds as follows. A finite dataset of 2,000,000 records is partitioned equally among N processors. Each processor is an independent self-contained computer that is part of a computing cluster (further referred to as a Node). One node is designated as a Master and is responsible for task allocation and result unification; while other nodes are designated Workers for the backpropagation neural network. BPNN (Williams et al 1986) is one of the widely used neural network training algorithms and has shown robust performance in many applications (Frasconi et al 1993). Several parallelization techniques have been introduced for BPNN (Lee and Siddiqui 2004; Klauer et al 1994; Svensson et al 1992). Our choice for a cluster computer is a training method based upon set partitioning and epoch-based weights update schemes. This allows us to reduce the amount of inter-processor communication needed to distribute data, which is beneficial for a cluster computer environment with a relatively slow inter-communication links. BPNN is trained iteratively, until an acceptable mean square error rate is achieved. At the beginning of each iteration (called epoch), master process creates a series of task requests and forwards them to the workers via a communication technology of choice. Each request contains information on a fraction of the entire dataset to be processed. Worker processes work on the corresponding fractions of the dataset and communicate results back to the master. Master process then aggregates all partial results. Once data mining step is complete, the quality of training result is evaluated. Quality is measured by the classification rate with regard to the training dataset and the new testing dataset, instances of which were not part of the training dataset. Usually, good classification rate on the training dataset is regarded important; but good results on a testing dataset are also significant, as they imply that the data mining process was able to extract knowledge that represents not only the patterns in the training dataset, but also those of unseen patterns.

## IMPLEMENTATION AND RESULTS

The BPNN parallel neural network classifier problem for intrusion detection was implemented using two distributed communication technologies – High Level Architecture, and TSpaces. Hardware platform used was a cluster computer with 16 individual PCs equipped with 900 MHz AMD Athlon processors, 1GB memory and Fast Ethernet, interconnected via Cisco switch, running Linux OS.

RTI implementation involved defining user interaction classes and devising communication strategy among federates. First, the Worker federates were started, that would join RTI federation execution, publish and subscribe to interaction classes and partition the training file. Then each Worker waits for a work order from a Master process, and upon receipt, computes BPNN weights. After sending results back to the Master, the Worker awaits further work orders in an infinite loop. The Master process retrieves initial weights, allocates portions of work order to each Worker process, then sends out work orders and awaits results. When all the individual results have been received and combined, an error is computed. If within the threshold (e.g., 0.01), the result is accepted and Master displays total run-time. Otherwise, a new set of work orders is distributed among the Workers. The total work order is divided in a way that each worker receives equal part of the training file for processing, since the program is executed in a homogeneous parallel environment with all nodes having approximately equal system resources.

The TSpaces implementation differs from RTI implementation in that the Master doesn't communicate with a specific Worker process, but rather places a work order in a virtual shared memory. Since task processing time of Worker process is much larger than network communication time, and having the number of Workers equal to the number of work orders, the problem is evenly distributed among available processors. The algorithms for Master and Worker classes are given below:

```
Class Worker {  
  
    Connect to RTI;  
    Create / Join Federation Execution;  
    Publish / Subscribe to Interaction Classes;  
    Retrieve Training Data;  
  
    Loop {  
        Receive Work Order;  
        De-serialize Parameters;  
        Build / Verify Work Order Set;  
        Compute BPNN Weights;  
        Serialize Parameters;  
        Send Result to Master;  
    }  
}
```

```

Class Master {

    Connect to RTI;
    Create / Join Federation Execution;
    Publish / Subscribe to Interaction Classes;
    Read Initial Weights;
    Get Start Time;

    Loop {
        Compute New BPNN Weights {
            Send Out Work Orders to Workers;
            Receive All Results;
        }
        Test Result {
            If Result Successful, Display Runtime and Quit;
            Else Continue;
        }
    }
}

```

The difference in implementation using RTI and TSpaces libraries is in Send and Receive methods. While RTI publishes interaction and transmits message via multicast to all federates, TSpaces implementation puts a data tuple into virtual shared memory and notifies all clients that data is available for pickup (via signaling or polling). Each client then picks up a tuple from tuple space, effectively removing it from shared memory. Since there is no addressing scheme and no way to specify a particular processor to send data to, the tuple space clients continuously scan for data that matches specific template.

The results of running these implementations on a varied number of cluster nodes, is given in the table 1, and plotted in figure 1. Two cluster computers were used for HLA/RTI implementation – Scerola cluster (contains 16 900MHz Athlon processors), and Ariel cluster (has 32 dual-P4 2.6GHz processors); the rest of the hardware is equivalent in both clusters. Even though

the HLA implementation required considerably more effort from software engineering perspective, the results show a considerable overall decrease in the total execution time, attributed to faster inter-node communications. The performance of HLA is quite different in the case of only one worker. This may be attributed to the overhead of HLA as a more heavyweight (but also more robust) solution to distributed system problems. As far as the overall expense of the distributed system solution goes, the total processor time (on all nodes) plus inter-node communication demonstrates an added overhead of a distributed system. For instance, using RTI with one worker yields 2129 processor-seconds execution time. When using 16 processors increases the total time to 3267 processor-seconds per problem. In situations where the processor time is an expensive resource, a careful cost analysis and planning has to be performed prior to implementing the distributed system.

<b>Nodes/Exec Time (sec)</b>	<b>HLA/RTI (S)</b>	<b>TSpaces (S)</b>	<b>HLA/RTI (A)</b>
1	2129.37	1837.13	417.83
2	1115.98	1323.58	251.16
4	601.24	697.64	188.95
8	345.02	380.83	134.93
16	204.23	243.49	105.32

\* (S) = Scerola Cluster (16 x 900MHz Athlon, 1GB RAM), (A) = Ariel Cluster (16 x dual P4 2.6GHz, 1GB RAM)

Table 1: Parallel BPNN Program Execution Times

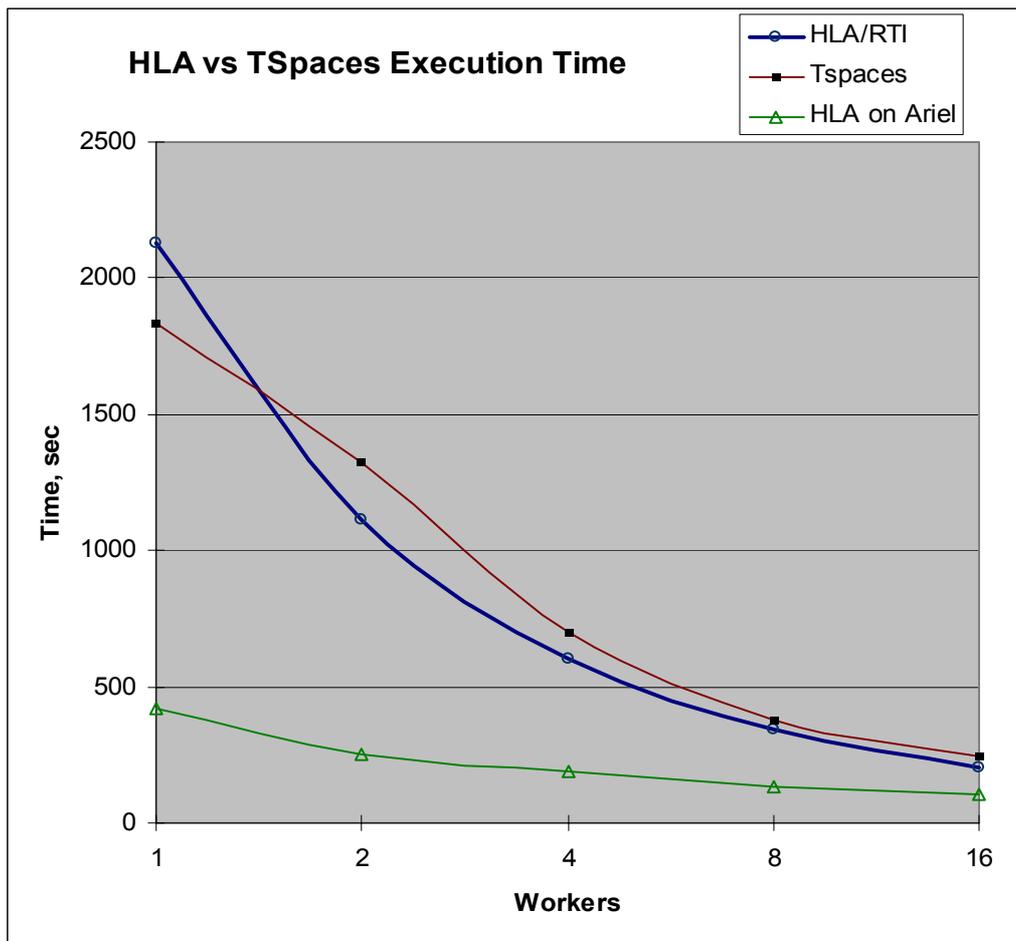


Figure 1: Parallel BPNN Program Execution Times

Examining performance of HLA/RTI implementation on both a single-processor and dual-processor cluster computers, we confirm that Java Virtual Machine efficiently utilizes dual-processor configuration and dramatically speeds up computationally-intensive tasks without further modeling by the researcher. This demonstrates cost-effectiveness of using a dual-processor cluster configuration, since processing time is considerably reduced for the same-size task, compared to single-processor cluster configuration. In our tests of BPNN algorithm on all 32 nodes of Ariel cluster, the execution would occasionally deadlock. This is due to very small processing times on the client processors, resulting in increased utilization of communication channel. As the number of processors increases in a parallel system, problem implementation has to be planned and evaluated carefully to avoid deadlock problems, as network communication becomes the bottleneck.

## CONCLUSION

In this paper we have presented comparison of implementations of a data mining problem using two

distributed technologies – HLA and TSpaces. The results demonstrate that the knowledge discovery and data mining algorithms can be easily implemented over a distributed tuple space; however, this comes at an expense of communication efficiency. HLA is a more heavyweight solution, requiring more effort in implementation and more overhead. This is noticeable in the results, when only one worker is used. However, when the amount of network communication increases, RTI implementation shows scalability in communication time. Even though total Master execution time comprises processing and communication among nodes, all other code and data being identical, the results demonstrate particular difference in network communication. Given the significant difference in communication component, we conclude that the High Level Architecture/RTI is a better choice for mission critical distributed applications with considerable amount of inter-node communications. The results of this study will be used during the implementation of our distributed simulation system for wireless networks project.

## ACKNOWLEDGEMENTS

This work was partially supported by ARO under grants DAAD19-01-1-0502, W911NF04110100 and NSF under Grant EIA 0086251. The views and conclusions herein are those of the authors and do not represent the official policies of the funding agencies or the University of Central Florida.

## AUTHOR BIOGRAPHIES



**Ratan Guha** is a professor at the department of Computer Science, University of Central Florida, with primary interests in Computer Networks, Distributed Computing, Distributed Simulation, and Computer Graphics. He received his PhD from the University of Texas at Austin, and his M.Sc. in Applied Mathematics, and B.Sc. with honors in Mathematics from the Calcutta University, India.



**Joohan Lee** is an assistant professor in the School of Computer Science at the University of Central Florida in Orlando, Florida. He received a PhD in Computer Science from Syracuse University in 2002. His research interests include parallel and distributed computing, computer and network security, and high performance data mining.



**Oleg Kachirski** is currently a PhD student in the School of Computer Science at the University of Central Florida. He received his M.Sc. in Computer Science from the University of Central Florida in 2002. His research interests include high performance parallel and distributed computing, modeling and simulations.

## REFERENCES

- Beowulf Project, <http://www.beowulf.org/index.html>
- M. Reid, "An Evaluation of the High Level Architecture (HLA) as a Framework for NASA Modeling and Simulation". In *Proceedings of the 25<sup>th</sup> NASA Software Engineering Workshop*, Goddard Space Flight Center, Maryland, November 2000.
- P. Wyckoff, "T Spaces". *IBM System Journal*, 37:3, August 1998.
- J. Haines, R. Lippmann, et al, "1999 DARPA Intrusion Detection Evaluation: Design and Procedures". *Lincoln Laboratory Technical Report 1062*, Massachusetts Institute of Technology, 2001.
- J. Lee and M. Siddiqui, "High Performance Data Mining for Network Intrusion Detection". In *Proceedings of IASTED International Conference on Parallel and Distributed Computing and Systems (PDCS 2004)*, MIT Cambridge, November 2004.
- R.J. Williams et al, "Learning Representations by Back-Propagating Errors". *Nature*, Vol. 323, pp. 533-536, 1986.
- P. Frasconi et al, "Successes and Failures of Backpropagation: A Theoretical Investigation". *Progress in Neural Networks*, Vol. 5, pp. 205-242, 1993.
- B. Klauer et al, "Pipelining and Parallel Training of Neural Networks on Distributed-Memory Multiprocessors". In *Proceedings of IEEE World Congress on Computational Intelligence*, 4(27), pp. 2052-2057, 1994.
- B. Svensson et al, "Using and Designing Massively Parallel Computers for Artificial Neural Networks". In *Journal of Parallel and Distributed Computing*, 14(3), pp. 260-285, 1992.
- KDD Cup 1999 Data,  
<http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html>

# WEB SERVICES COMPOSITION: A PRAGMATIC VIEW OF THE PRESENT AND THE FUTURE

Dhaval Kumar Thakker, Taha Osman, David Al-Dabass  
School of Computing & Informatics,  
Nottingham Trent University,

Burton Street, Nottingham NG1 4BU, United Kingdom.

Email: dhaval.kumar.thakker@students.ntu.ac.uk, taha.osman@ntu.ac.uk, david.al-dabass@ntu.ac.uk

## KEY WORDS

Web Services, Business Process Management, BPEL, CDL.

## ABSTRACT

Web Service is loosely coupled, highly accessible distributed computing technology that can expose applications beyond the firewall. Composition of Web Services has received much attention from the business and the research community. Composition techniques are classified as static, dynamic and semi-automatic composition, each addressing different application areas and requirements. Here, we evaluate such approaches from two perspectives: as Business Process Management (BPM) solution and the facilitation they provide to the composition participants: end users, developers and composers in their present form. In this paper, we use the present workflow-based composition standard WS-BPEL, explore its merits and limitations and analyze another static composition standard WS-CDL which has complimentary role to BPEL for the business process management. We also discuss dynamic composition as the future work and what it can offer.

## 1. INTRODUCTION

The last decade has witnessed an explosion of application services delivered electronically, ranging from e-commerce to information service delivered through the World Wide Web (WWW) to the services that facilitate trading between business partners, better known as Business-to-Business (B2B) relationships. Traditionally these services are facilitated by distributed technologies such as RPC, CORBA and more recently RMI. Web Services is the latest distributed computing technology. It is a form of remote procedure call like other distributed computing technology, but uses XML extensively for the messaging, discovery and description. The use of XML messaging makes Web Services platform and language neutral. Web Services use SOAP (Simple Object Access Protocol)[1] for XML messaging, which in turn uses ubiquitous HTTP for the transport mechanism. HTTP is considered as a secure protocol thus it allows the Web Services to be exposed beyond the firewall. The Web Service messages and

operations with invocation details are described using a platform-independent language WSDL (Web Services Description Language)[2]. Web Services can be published and discovered using UDDI (Universal Description Discovery and Integration)[3] protocol. The Web Services architecture centred on WSDL, UDDI and SOAP is an instance of Service Oriented Architecture (SOA). Using this architecture services can be published using UDDI, with WSDL based description, and can be searched, called and bind at run time making it loosely coupled and highly accessible.

To take advantage of these features of Web Services, network applications services have to be developed as Web Services or converted into Web Service using the wrapping mechanism [4]. Moreover, multiple Web Services can be integrated either to provide a new, value-added service to the end-user or to facilitate co-operation between various business partners. This integration of Web Services is called "Web Services composition" and is feasible to achieve because of the Web Services advantages of being platform, language neutral and loosely coupled.

The logic for the composition mainly involves two activities: selection of the candidate Web Services that fulfil the requirement in accumulation and flow management. Flow management is further categorized into control and data flow, where control flow is the order in which Web Services operations are invoked, while the data flow is the order in which the messages are passed between the Web Services operations. The level of automation provided in performing selection of services and flow management classifies composition into static, semi-automatic and dynamic. Static composition involves prior hard coding of the service selection and flow management. Performing selection and flow management on the fly, in machine-readable format leads to dynamic composition. In semi-automatic composition, service composer is involved at some stage.

The focus of the discussion is to contrast these approaches keeping in mind the feasibility of their

implementation today and in the near future. The comparison perspective focuses on the ease of use and facilitation provided to the participating parties, to the end user who is going to use the composed service, to the service developer (service provider) and to the service composer. The latter either can be software developer, the business analyst or logic based agent programmes. We will investigate Web Service composition using above mentioned criteria for the both the cases where the existing application is non-web service based or Web Service has no WSDL file or description available.

The structure of the paper is as follows: section 2 provides more detail on the Web Service Composition and discusses composition using WS-BPEL, a prominent industry standard. Section 3 provides insight on the futuristic approaches to the composition. In section 4 we conclude with our analysis of these approaches.

## **2. OVERVIEW OF THE WEB SERVICES COMPOSITION**

Traditional techniques approach Web Services composition as the Business Process Management (BPM) solution. Business process can be considered as the group of activities to carry out business goals [5]. Business applications represent such activities in the business processes, for example a customer order fulfilment process will include individual applications for the activities: customer placing an order, checking account status, verifying order and despatch. BPM deals with achieving the integration of these individual applications to achieve business process view.

Business process can have scope within inter and intra organization relations. EAI (Enterprise Application Integration) is the BPM solution to achieve intra-organization business applications integration, while B2B integration software (e.g. Electronic Data Interchange) addresses the problem for inter organization business application integration. Traditional EAI and B2B integration solutions are very complex, proprietary and presumes many details about the participating applications making them tightly coupled. As the business applications are now being developed using the Web Services, the BPM problems (EAI, B2B) are being addressed with the composition of Web Services, mainly to utilize SOA based Web Service features.

Main industrial standards to achieve such composition of Web Services as BPM solution are WS-BPEL (Web Services Business Process Execution Language) [6], WS-CDL (Web Services Choreography Description Language) [7] and BPML (Business Process Modelling

Language) [8]. These approaches use WSDL extensively and build on top of it. WSDL based operations and messages with the data types are the main details based on which the flow management and other essential requirements for composition can be built upon. But to achieve business process view of the composition, composition specification needs to be built on the explicit process model [9]. This process model addresses requirements for describing flow management in composition, handling business transaction with roll back facility, state management for business interaction support, and also handling exception and errors. The process model and the extent to which these features provided, differentiates these standards.

### **2.1 Composing services using BPEL**

WS-BPEL (also called BPEL) specification - enhances and replaces existing standards XLANG from Microsoft and WSFL from IBM. Apart from being based on WSDL, it uses workflow management as process model to achieve the formalization for control and data flow. WS-BPEL facilitates static composition, as the selection of services and decision on flow management is done priori. All the participant services in BPEL are modeled as partners. Partners contribute to the total processing capability of the BPEL process. BPEL process also has its own processing capability for dataflow, control flow, data manipulation, fault and event handling and state management. The significance of BPEL architecture is that the process itself is published as a Web Service. This composed BPEL service can be treated as a single Web Service and can be used for further composition hence facilitating recursive composition.

BPEL standard aims to be the de facto standard for inner-organization (EAI) and inert-organization (B2B) Web Services integration. Hence, BPEL is categorized into executable BPEL (BPELe) and abstract BPEL (BPELa), where BPELe maps well to the requirements of inner-organization based Web Services integration and BPELa is aimed to cover the requirements for achieving integration in cross-organizational domain. BPELe represents the orchestration feature of BPEL, where the Web Service operations are orchestrated from a single party view. BPELa uses same language constructs and semantics as the executable but rather presents the aspects that are more suitable for Business-to-Business interaction.

We describe a Web Service composition scenario implemented using IBM BPEL runtime, BPWS4J [10]. The scenario is based on a travel agent service, which manages the reservation of airline and hotel for the customer trip. Travel agent is implemented as BPEL process, which is the composition of four Web Services: AirFrance service, AirUSA service, HotelRating service and HotelService service. Process logic for the travel

agent is (fig 1): to check the availability of flight service from two competing airlines AirFrance and AirUSA, make flight reservation, and then retrieve hotel ratings from the HotelRating service at the destination city and make the reservation using HotelService Web Service at the selected hotel. The reservations are made based on customer preference (Air Line price preference: cheapest, medium or high and hotel rating preference: A, B or C).

### 2.1.1 Travel Agent Example

As BPEL is built on top of WSDL, WSDL file of partner business services are required for the composition process. This fact is described in BPEL using partnerLinkType. The portType of such a Web Service defines the role of partner in the composition. Listing 1 shows AirFrance and AirUSA web services as partners and the role they play in the composition using portTypes (i.e. fr: is the unique identifier for the AirFrance WSDL file). Figure 1 is the sequence diagram for the travel agent service. 1.1.a and 1.1.b are two activities for checking the flight between source and destination city is available or not, performed in parallel. The BPEL syntax for this using <flow> to achieve parallel execution is shown in the listing 2. Both invocations are executed in parallel.

Similarly other operations for checking the possibility of reservation are performed on AirFrance and AirUSA, and reservation is made after comparing the price (activities 1.2.a, 1.2.b, 1.3.a, 1.3.b in figure 1). The payment details are omitted to keep the example simple. Listing 3 shows the code where the user has specified the cheapest flight reservation in his preference.

```
<plnk:partnerLinkType name="airFrancePLT">
  <plnk:role name="AFcheckServices">
    <plnk:portType name="fr:AirFrance"/>
  </plnk:role>
</plnk:partnerLinkType>
<plnk:partnerLinkType name="airUSAPLT">
  <plnk:role name="AUcheckServices">
    <plnk:portType name="usa:AirUSA"/>
  </plnk:role>
</plnk:partnerLinkType>
```

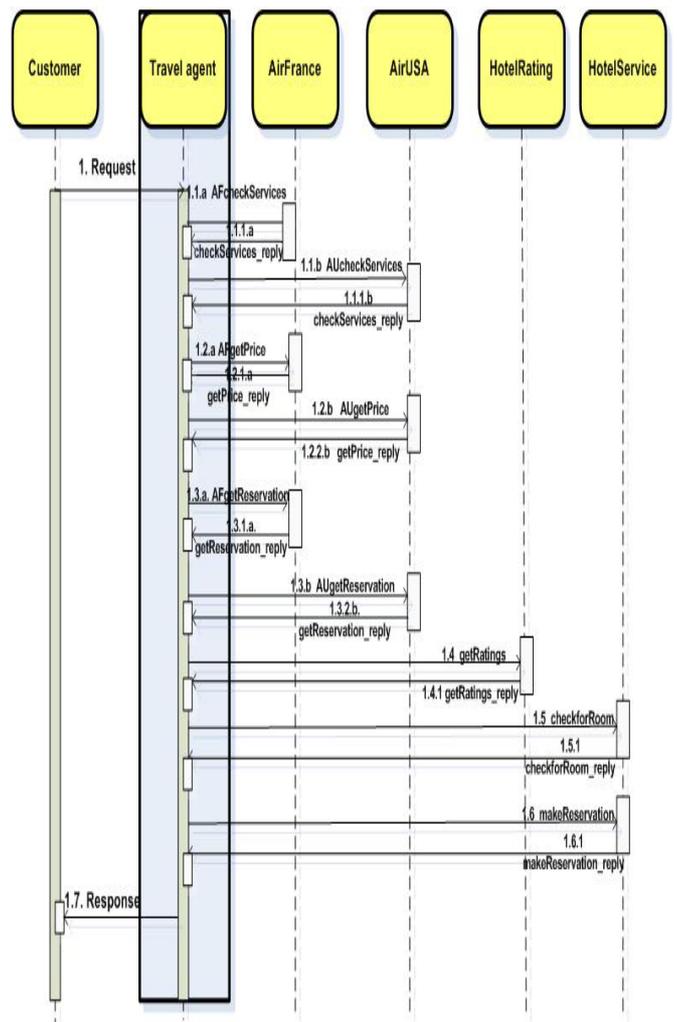
**Listing 1. Partners in BPEL process**

```
<flow>
<invoke name="invokeAirFrancecheckServices"
partnerLink="AFcheckServicesPL"
portType="fr:AirFrance".....>
<invoke name="invokeAirUSACheckServices"
partnerLink="AUcheckServicesPL"
portType="fr:AirUSA".....>
</flow>
```

**Listing 2. Concurrency using <flow>**

```
<switch name="comparePrices">
<case condition="bpws:getVariableData
('compInfo','PriceAirUSA') &lt;
bpws:getVariableData('compInfo','PriceAirFrance')
">
<invoke name="AUinvokegetReservation" </case>
<otherwise>
<invoke name="AFinvokegetReservation"
<partnerLink="AFgetReservationPL" .....>
</otherwise>
</switch>
```

**Listing 3. Selecting the cheapest AirLine<switch>**



**Fig 1. Sequence diagram for the composition [Travel Agent viewpoint]**

Implementation of travel agent example shows the expressiveness of BPEL as composition language. In the section 2.2 we will consider the merits and limitations of BPEL architecture when applied to the composition problems from inter or intra organization domain. We will use our travel agent example for the discussion.

## 2.2 Discussion on BPEL facilitated composition

We will first consider BPELe for the enterprise application integration and Business-to-Business Integration relations. The architecture of the BPELe process model assumes the selection of the services to be done manually, for this reason the interpretation of the requirement expected from the service is according to the understanding of the composer. This assumption is easy to achieve for the inner-organization relations, where the composer has access to all the internal Web Services details, which makes the achieved solution tightly coupled. This is sufficient for the enterprise application integration, as the change in the process logic and addition or removal of services (Business Process Reengineering) can be done in-house, in private domain.

This approach has serious limitations when it comes to B2B integration, where candidate services can be from public, external or cross organizational domain. To illustrate this in our travel agent example, which is a B2B relation, lets consider a scenario, where the new business (AirUK) wants to join this composition. It is really difficult to add new businesses that are not part of the old mix, because it is required that they as a minimum implement the following operations: 1) an operation to check whether flight service is available between two cities 2) an operation to check the trip expense between these two cities.3) an operation for making reservation. Hence, the addition can be made successful if the travel agent business publishes this information as an agreement i.e WSDL file or text description and the AirUK business has or newly implements the service with the above operations and then makes WSDL file available for the composition.

Apart from being tightly coupled, BPELe assumes the B2B integration from the single party viewpoint as the requirement specified above is from the travel agent business logic viewpoint. Real world B2B integrations are peer-to-peer in place of being centralized. It is more like contract in terms of performing responsibility in the collaborative work, as a result such integration requires a notion to specify “ I will provide this functionality to achieve this from all my partners (other businesses)”.

BPEL specification claims to facilitate the B2B integration using abstract BPEL, where each party describes their B2B participant functionality using BPELa file. Hence all such businesses can publish their own BPELa files and can work together. But again if these BPEL files are not developed in collaborative way at first place, then the consumer-producer relationship cannot be achieved using BPELa. For this reason, the bottom-up approach i.e. implementation first and then description adopted by BPELa is not adequate for the B2B relations. Therefore, the top-down approach that

describes functionality first and considers implementation at later stage is better suited for such B2B domain Web Services composition. Overall, BPELa provides notion to specify only “this is the functionality I provide” in place of required collaborative notion “I will provide functionality to achieve this from all my partners (other businesses)” making BPELa inadequate for B2B integration.

To consider the facilitation provided to the composition participants, in case of service provider if provider wants to make their service available for composition then they need to provide minimum functionality required by the business logic from the single party perspective in the integration. Considering new AirUK service for travel agent composition, AirUK to be part of the composition, options for the service provider of the new AirUK service will be:

a) If the AirUK has web service but does not implement required functions, then the service needs to be modified to accommodate the required functionalities. WSDL file for this service can be made available to the travel agent via UDDI. This WSDL file can contain only those operations useful to the agent service.

b) If the AirUK has a non-web service application, with the required functionality already built-in, then just a WSDL file is required to be created without modifying the existing non-web service application. As such Non-Web Services, which have WSDL, can still be composed in BPEL. BPEL execution engine uses WSIF (Web Services Invocation Framework)[12] for the Invocation of such non web-services [13].

Considering the case of service composer, they mainly encounter problems in parameter mismatch during the flow management, as the output of one service operation has different format from the input of next service operation in the flow logic. The other case might be the bit of missing functionality, which probably is already available as legacy code. To address both the case, BPEL implementations [11] provides a mechanism where the higher language code(C#, Java) can be inline in the BPEL process. BPEL-J [14] is another such an industrial effort to combine BPEL and Java, where BPEL is for defining Business processes and Java to provide general programming language expressiveness, allowing each language to do what it does best.

To illustrate the case of end-user, in our travel agent example, the BPEL agent process is published using JSP technology. This service can be retrieved using this simple web page or can be retrieved from the public UDDI registry, where the access point in the business service binding information points to this JSP page. In

such B2C interactions, it is totally transparent from the end-user that the service is a Web Service, composition of multiple Web Services, could be implemented on heterogeneous platforms or using different programming languages. As Semantic Web [15] introduced a scenario, where the intelligent software programmes -agents work on behalf of the end user, and can use composition to satisfy user requirement or task however BPEL, in its' present form has nothing to offer to facilitate this approach. The similar way it has little to do with automatic service composition.

To conclude, BPELe is the best candidate for composing private, inner-organization Web Services making suitable for EAI. Business analysts and developers can work together and can compose such Web Services manually using BPELe. The composition is hard coded and the developers should have the explicit knowledge of all the details of participating business services. The control and data flow logic also should be known in advance. BPELe can also facilitate tightly coupled B2B integration; in contrast BPELa is poor candidate for describing peer-to-peer B2B collaborations.

### 3. WEB SERVICES COMPOSITION APPROACHES IN THE NEAR AND THE DISTANT FUTURE

#### 3.1 Composition using WS-CDL

During our discussion over BPELa as B2B integration language, we highlighted two main requirements adequate for B2B integration. First, a description language is required to facilitate top-down, using which collaborative functionality in B2B integration is described, agreed first and the respective parts can be implemented by each partners. Second, the description should be from peer-to-peer point of view, since natural B2B integrations are peer-to-peer collaborative relationships, not governed by a single party. Recently promoted working draft WS-CDL [6] from W3C Web Services choreography working group confirmed our conclusions that more work on BPEL is required to make it adoptable for B2B integration by addressing these requirements and adopting approach specific to the B2B relationship.

WS-CDL is a description language, using which the B2B integration partners can first describe the collaborative functionality. This description document is considered as a contract and each party can implement their own part. The WS-CDL document describes common and complementary behaviour of all the parties involved, making the viewpoint global, peer-to-peer [6]. Travel agent B2B integration previously illustrated in figure 1 transforms into figure 2 under WS-CDL architecture, where travel agent is no longer the

controller of the integration, in view of the fact that the respective functionality and the ordering of the activities performed is from the perspective of all the parties involved and agreed by all in the CDL document.

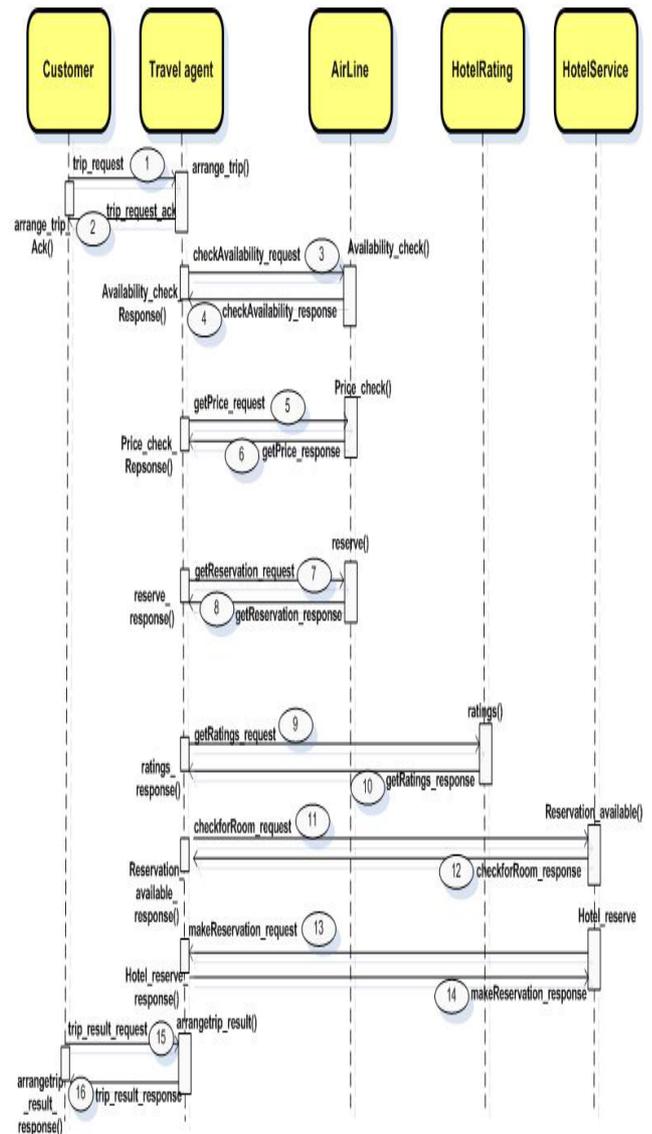


Fig 2. Travel Agent B2B integration scenario using CDL [global viewpoint]

CDL document describing this scenario becomes contract between participating parties in terms of the functionalities they agree to provide. For example, considering the interaction between businesses Travel Agent (TA) and AirLine (AL) Services in activity (5,6): TA interacts with AL service for checking the price (activity 5) for the required flight; for that reason AL provides a web service operation Price\_check, So that TA can send getPrice\_request. And TA has the Price\_check response operation, which AL uses to send the response message.

Listings 4 and 5 describe this interaction in CDL. CDL has the notion of role and relationship for the participating parties in particular interaction. The interaction PriceCheckRequestInteract in figure 9, documents the fact that it is the fifth activity in sequence, the participant in the interaction are those who have TravelAgentAirLineBinding relationship, operation affected is the AirLine Web Service operation Price\_check and getPrice\_requestAtTravelAgent (Unique Identifier for getPrice\_request) message exchanged from the role TravelAgent to AirLine.

```

<roleType name="AirLine">
  <behavior name="airline"
    interface="AirLinePT"/>
</roleType>

<roleType name="TravelAgent">
  <behavior name="TravelAgentForAirLine"
    interface="TravelAgentForAirLinePT"/>
    .....
</roleType>

<relationshipType
  name="TravelAgentAirLineBinding">
  <role type="AirLine"/>
  <role type="TravelAgent" behavior
    ="TravelAgentForAirLine ">
    .....
</relationshipType>

```

**Listing 4. Roles and Relationships**

```

<interaction name="PriceCheckRequestInteract"
  operation="Price_check" ..... >

<participate relationship="Travel
  AgentAirLineBinding"
  fromRole="TravelAgent"
  toRole="AirLine"/>
  <exchange name="request"
    action="request">
  <send variable="getPrice_requestAt
    TravelAgent"/>
  <receive variable="getPrice_reqeustAtAirLine"/>
  </exchange>

</interaction>

```

**Listing 5. Interaction**

The other aspect of CDL architecture is that the internal business logic of each party remains hidden from the business partners. i.e. in this travel agent application, after getting price from airlines, travel agent can have internal application to implement the business logic for the air line selection based on certain criteria, while the external detail described in CDL document is just an

operation to make reservation at particular airline. This internal logic can be an EAI application composed using BPEL.

If considering the facilitation provided to the participants in CDL, service composer designs the global interface CDL and as the other parties follow the interface, composer does not have to deal with individual service providers and can easily accommodate them once providers follow the global interface. Having a global interface also liberate composer from problems of functionality and technicality mismatch. CDL is still descriptive language but can play the role like WSDL, to create stub files so that each party service provider can have blue print of what they are supposed to implement [16]. Consider the benefit of having such descriptive language with top-down approach when the integration takes place between large numbers of Web Services.

Overall, CDL is designed to address the requirements for B2B integration and compliments BPEL, which is a better candidate for EAI. Consequently, CDL and BPEL together address the problem of BPM by facilitating static composition.

**3.2 Dynamic Composition**

Commercial Institutions are focusing their efforts on standardizing the static composition techniques in preparation for their wider adoption amongst the business community. In contrast, research community foresee that there is a better futuristic potential in the semi-automatic and automatic or dynamic composition of Web Services. Dynamic composition achieved can serve a great range of business domains. In such kind of composition participating services can be external and public. User can specify parameters for the successful composition and the composition is performed at the run-time. The solution addresses the problems of identifying candidate services, composing them, and verifying closely that they satisfy the request.

Dynamic Web Services composition is the topic of our ongoing research. As per our definition of composition automation, a semantic based language specifying the capability of Web services is required so that services can be selected on the fly for the composition according to user parameters. Semantic web based OWL-S [17][18] can be utilized to achieve this. A layer on top of OWL-S is required for: automating flow management, interpreting the semantic web service based capability, and to manipulate data before invoking operations and to carry out execution according to the user requirements. Artificial intelligence planning, workflow management, and intelligent agents etc. are the available options.

Using dynamic solution, users will get the maximum flexibility, as composed services will be an optimum mix based on the user specified input parameters. The service providers will be able to participate in the composition to their benefit with minimal effort, as the human developer will be taken out of the loop.

#### 4. CONCLUSIONS

Web Services composition approaches are characterized as static, semi-automatic and dynamic. Well-known industrial standards like BPEL and CDL facilitate composition as the BPM solution. BPEL categorized into BPELe and BPELa is designed to address intra and inter organization BPM integration problem. BPELe architecture makes it better suited for enterprise level integrations while BPELa is a poor candidate for B2B integrations. CDL is an effort to overcome the limitations of BPELa and covers from where the BPEL has left. These standards satisfy current business requirements by adopting static composition.

In this paper, we also briefly explored the research efforts into dynamic composition of Web Services, and noted that future composition solutions based on OWL-S can be applied to wider range of business applications to facilitate machine-readable, agent based automatic Web Services composition scenario.

#### REFERENCES

- Gudgin M et al. 2003. "Simple Object Access Protocol(SOAP) Version 1.2", W3C recommendation, 2003, <http://www.w3.org/TR/soap12-part1/>
- Christensen E et al. 2001. "Web Services Description Language( WSDL) version 1.1", W3C recommendation, <http://www.w3.org/TR/wsdl/>
- Hately A et al. 2003. "Universal Description Discovery and Integration (UDDI), Version 2.0" OASIS Specification, <http://www.oasis-open.org/committees/uddi-spec/doc/tcspecs.htm#uddiv2>
- Osman T et al, 2005. An Integrative Framework for Traffic Telematics Web Services, to be appeared in the *Parallel and Distributed Computing Networks Conference(PDCN 2005)*.
- Leymann F at al. 2005. "Web Services and business process management". *IBM Systems Journal*, Volume 41-2, 2002, 198-211.
- Andrews T et al, 2003. "Business Process Execution Language for Web Services, Version 1.1", <http://www-128.ibm.com/developerworks/library/wsbpel/>
- Kavantzias N et al, 2004. Web Services Choreography Description Language (WS-CDL) Version 1.0, <http://www.w3.org/TR/2004/WD-ws-cdl-10-20041217/>
- Arkin A, 2003. "Business Process Modelling Language Version 1.0", <http://www.bpmi.org/bpml-spec.htm>.

Staab S et al, "Web services: been there, done that?" *IEEE Intelligent Systems*, Vol. 18, Issue. 1, Jan 2003, 72- 85.

BPWS4J, "Business Process Execution Language for Web Services Java Run Time". <http://www.alphaworks.ibm.com/tech/bpws4j>.

Oracle BPEL Process Manager (PM), <http://www.oracle.com/technology/products/ias/bpel/index.html>.

Web Services Invocation Framework, Apache Software Foundation Web Services project, <http://ws.apache.org/wsif/>

Afshar M et al, 2004. "WSIF and JSR-208: Flexible binding frameworks for today and tomorrow", *Web Services Journal*, <http://www.sys-con.com/story/?storyid=46558&DE=1>

Blow M et al, 2004 "BPELJ: BPEL for Java", White paper from BEA and IBM, <http://www-106.ibm.com/developerworks/webservices/library/ws-bpelj/>

Lee B at al, 2001. "The Semantic Web", *Scientific American*

Austin D et al, 2004. "Web Services Choreography Requirements", *W3C working draft*

Martin D et al, 2004. "Semantic Mark-up for Web Services, OWL-S version 1.1"

Martin D et al, 2004, "Bringing Semantics to Web Services: The OWL-S Approach", *Proceedings of the First International Workshop on Semantic Web Services and Web Process Composition (SWSWPC 2004)*.

#### AUTHOR BIOGRAPHIES

**DHAVALKUMAR THAKKER** took his Masters degree in Data Communication Systems at Brunel University, London-UK. He also received his bachelors degree in engineering from India. He is a PhD student of Nottingham Trent University, School of computing and Informatics. His current research interests are distributed computing technologies, Web Services composition, Semantic Web and Ontologies. His email address is [dhavalkumar.thakker@students.ntu.ac.uk](mailto:dhavalkumar.thakker@students.ntu.ac.uk).

**TAHA OSMAN** is senior lecturer at the Nottingham Trent University, UK. He received a B.Sc honours degree in Computing from Donetsk Polytechnical Institute, Ukraine in 1992. He joined the Nottingham Trent University in 1993 where he received an MSc in Real-time Systems in 1994 and a PhD in 1998. His current research concerned investigation of Fault-Tolerance in Open Distributed Systems. His email address is [taha.Osman@ntu.ac.uk](mailto:taha.Osman@ntu.ac.uk).

**DAVID AL-DABASS** is professor at the Nottingham Trent University, UK. His research work explores algorithm and architecture for machine intelligence. His email address is [david.al-dabass@ntu.ac.uk](mailto:david.al-dabass@ntu.ac.uk) and web page is <http://ducati.doc.ntu.ac.uk/uksim/dad/webpage.htm>.

# DISTRIBUTED IMPLEMENTATION OF A HETEROGENEOUS SIMULATION OF URBAN ROAD TRAFFIC

Camelia Avram  
Technical University of Cluj-Napoca  
Ghe. Baritiu str., No. 26 -28,  
400027, Cluj Napoca, Romania  
[camelia.avram@aut.utcluj.ro](mailto:camelia.avram@aut.utcluj.ro)

René Boel  
Universiteit Gent  
Technologiepark - Zwijnaarde 9,  
9052 Zwijnaarde Gent, Belgium  
[rene.boel@ugent.be](mailto:rene.boel@ugent.be)

## KEYWORDS

Road traffic models, road traffic simulation, distributed programming

## ABSTRACT

Efficient control of urban traffic requires the implementation of a sufficiently accurate model allowing prediction of the effects of various control actions (such as adaptation of red-green phases at different intersections). Given the size of the plant it is important that one can use a distributed implementation of this simulation model. The computational efficiency is also improved by using a heterogeneous model, where some parts of the network are represented by a macroscopic model, other parts by a microscopic model. Long road sections with fairly homogeneous traffic conditions are represented by time driven macroscopic models describing the evolution over time of flow, speed and density of vehicles in different locations. In other parts of the plant it is more efficient to use event driven microscopic models representing the times at which individual cars cross certain boundaries. This is the case for short sections of road and for intersections. This paper presents a Java implementation of such a modular distributed simulator for a large system built up from many interacting components of long road segments, short road segments, and intersections, together with the boundary conditions at the edges of the plant. Each component represents some randomness in the evolution of the plant. We also discuss how to connect time driven and event driven components to each other in a computationally efficient way.

## INTRODUCTION

On-line feedback control of urban traffic networks forms a challenging problem, both due to the size of the plant and due to the complexity of the dynamics to be taken into account. It is quite natural to try to use model based MPC controllers. This requires the use of a simulation package that is capable of evaluating the effects of several options for the setting of the control actuators (mainly traffic lights in the urban traffic control problem) faster than real time. The size of the plant imposes the use of a distributed simulation package. This paper describes a simple client-server based Java version of such a simulation package that achieves the goal of faster than real time simulation for a reasonably sized road network (even when

implemented on a single processor - future implementations on multi-processor systems can achieve state predictions for very large metropolitan networks).

We are looking for models that are modular, i.e. that can be developed by interconnecting simple, standardized representations of various components. In this paper we use as components:

- long segments (road sections that are several 100m long, without intersections, where the behavior within one segment is fairly homogeneous). For these long segments we use a time driven, macroscopic model, describing the evolution of aggregated variables density (vehicles per km) and average speed);
- short segments (road sections between two intersections close together); we use event driven models for these short segments, describing the event times "a vehicle enters the segment" and "a vehicle leaves the segment".
- intersections where several streams of vehicles interact, and influence each others behavior; we use event driven models for intersections. The events to be modelled are all the possible arrivals, and all the possible turns, and all the possible departures at the intersection. The model must represent all the mutual constraints between these events, like in a complicated multi-server queuing system.

Large networks can be obtained by composing many different components, the outflow of an upstream component being the inflow of the downstream component. Short segments and long segments only have one downstream neighbor, and one upstream neighbor; intersections on the other can have many upstream and downstream neighbors. This object oriented modelling approach (with as classes long and short segments, and intersections) leads to very simple development of simulation programs, that are easily adaptable to modifications to the road network (e.g. whenever road works occur one only needs to modify the model of the one single segment involved in the road works).

In section 2 below we describe in more details how the mathematical models of long and short segments, and of intersections, are constructed, and how these segments interact via the inflow and outflow of vehicles at their boundary. The only major difficulty is how the time driven long segments (outputting a stream of numbers describing the flow of vehicles in veh/minute crossing a boundary) can be translated into a stream of event times that serves as input to an event driven short segment. In section 2 we also emphasize the random distributions of

the model variables, so that our simulator can be used in a particle filtering state predictor (in (Mihaylova and Boel 2004) this idea is applied for a model of a freeway consisting only of long segments; the extension to an urban traffic state predictor is under development).

Section 3 then describes how different components are implemented efficiently in Java, while section 4 describes the client-server distributed Java program. Finally section 5 presents some results obtained with our simulation tool.

## COMPOSITIONAL MODELLING OF URBAN TRAFFIC

The mathematical model that forms the basis for our distributed simulation consists of many interconnecting components, the outflow of an upstream component being the inflow in the next down-stream component. For road networks where the distance between intersections is big, one often can assume that the traffic state is roughly the same over segments of a length of several 100m. This allows the use of macroscopic models; describing the evolution of the following aggregated variables (no individual vehicles are represented):

- the flow  $q(x, t)$  of vehicles (vehicles/minute) at time  $t$  and at location  $x$ ,
- the density  $\rho(x, t)$  of vehicles at time and at location  $x$ , in (vehicles/km) and
- the average speed  $v(x, t)$  of these vehicles, in km/h.

Classical macroscopic models develop partial differential equations for the evolution of these aggregated variables, and simulate the behavior of the network by time and space discretisation See e.g. the METANET simulator (Papageorgiou 2002).

A more modular approach, useful for distributed simulations of large systems, is the cell transmission model CTM of Daganzo (Daganzo 1994). This forms the basis for the simulator in this paper. The CTM framework divides the road network into cells of given length, and calculates only the number of cars  $N_{n,k}$  in the different cells at consecutive points in time  $t_k$ . The evolution of  $N_{n,k}$  depends on two functions:

- the sending function  $S_{n,k}$  measuring how many vehicles would cross the boundary between cell  $n$  and cell  $n+1$  in the interval  $[t_k, t_{k+1})$  provided cell  $n+1$  would be empty.  $S_{n,k}$  only depends on the state of cell  $n$  at time  $t_k$ .
- the receiving function  $R_{n,k}$  describing how many vehicles are allowed to enter cell  $n+1$  from cell  $n$ , taking into account that cell  $n+1$  can never contain more vehicles than allowed according to the speed dependent safety distance between vehicles.  $R_{n,k}$  only depends on the state of the cell  $n+1$  at time  $t_k$ .

The actual flow  $Q_{n,k}$  of vehicles from cell  $n$  to cell  $n+1$  during the time interval  $[t_k, t_{k+1})$ :

$$Q_{n,k} = \min(S_{n,k}, R_{n,k}). \quad (1)$$

The number of cars in cell  $n$  is updated according to the conservation of vehicles:

$$N_{n,k+1} = N_{n,k} + Q_{n-1,k} - Q_{n,k} \quad (2)$$

In (Boel and Mihaylova 2004) this CTM model is extended in two ways. First of all since speed

measurements are typically also available for feedback control, (Boel and Mihaylova 2004) includes a dynamic update equation of the average speed  $v_{n,k}$  in segment  $n$  at time  $t_k$ , taking into account speed convection, and speed adaptation by drivers who observe the density ahead of them. See (Boel and Mihaylova 2004) for details of this update equation from  $v_{n,k}$  to  $v_{n,k+1}$ .

Moreover (Boel and Mihaylova 2004) describes in detail how the update of  $(v_{n,k}, N_{n,k})$  to  $(v_{n,k+1}, N_{n,k+1})$  is a random function with well defined probability distributions. This allows the use of these update equation in developing a recursive Bayesian state estimator (Daganzo 1994), making the simulator a useful tool for model predictive control.

In the simulator for urban traffic presented in this paper, the model of (Boel and Mihaylova 2004) is used as a model for the long segments.

Short segments are represented using cellular automata, as in (Neubert 1999). Vehicles enter a short segment  $n$  through its upstream boundary at the event time "j-th vehicle enters segment  $n$ ". Vehicles are then propagated via a sequence of consecutive cells, moving to the next cell as soon as they can have driven a distance equal to the length of the cell (this time is calculated using the current speed of the vehicle), and when the next cell is free (not blocked by another vehicle, nor forbidden by safety constraints imposed by downstream vehicles). The speed of the vehicles is updated at the time "leave cell" starting with the speed they had when entering the cell. If the next cell is blocked the vehicle will slow down until the next cell becomes free. If the driver sees several cells ahead of him/her that are empty, and if the current speed is low, then the driver will speed up trying to reach an equilibrium speed. This description represents the progress of the vehicle through the short segment, until the vehicle leaves the short segment at the time of the event "vehicle leaves segment  $n$ ". This event time is then used as an inflow event "vehicle arrives at segment  $n+1$ " is segment  $n+1$  is a short segment. Note that in the distributed program implementation the events corresponding to crossing cell boundaries are local to the cell model, while the events "vehicle enters short segment" and "vehicle leaves short segment" are global variables.

If segment  $n+1$  is a long segment then the vehicles departure times from short segment  $n$  are aggregated during the time interval  $[t_k, t_{k+1})$  and this event count is used as the inflow  $Q_{n,k}$  into segment  $n+1$  in the interval  $[t_k, t_{k+1})$ . If segment  $n$  is a long segment, and segment  $n+1$  is a short segment, then the  $Q_{n,k}$  vehicles from segment  $n$  entering segment  $n+1$  during the time interval  $[t_k, t_{k+1})$  must be distributed as  $Q_{n,k}$  random events "arrival of vehicle into segment  $n+1$ ". Distributing the  $Q_{n,k}$  events uniformly and independently – rejecting random event times that violate the minimum safety distance with already accepted event times – is physically incorrect, and leads to an explosion of the computation time under dense traffic conditions (because most event times that are generated are rejected, and the generation of event times must be continued until  $Q_{n,k}$  events have been accepted). Hence we have developed a more realistic model for the event time generation, taking into account that vehicles

typically drive in platoons, the size of a platoon being a random variable with mean depending on the traffic density. For more details see section 3.

Note that for long segments we do not distinguish the lane a vehicle is in, while for short segments the cells represent both location and lane.

Finally we need a 3rd category of components, intersections. This component uses the event times "vehicle enters intersection i from stream j". Stream j not only specifies the inflow port but also the exit port where the vehicles of stream j must leave intersection i. Stream j events are generated by the outflow from the short segment just upstream from the intersection; destinations are obtained by randomization according to turning ratios observed at that intersection.

Intersections consist of many cells. Vehicles move through intersection i according to rules similar to those used in short segments, but now one must also specify to which downstream cells the vehicle goes, representing the various turns that vehicles can take in the intersection. The stream identifier specifies which next cell(s) the vehicle can choose from, thus avoiding that vehicles would travel in circles. The progress of the vehicles is constrained by both the state of the traffic light, by the right of way priority rules, and of course by the availability of the cells the vehicle wants to enter.

By interconnecting short and/or long segments and intersections one can represent an arbitrary urban traffic network consisting of one-way lanes. Overtaking, using empty spaces in a short or long segment representing traffic in the opposing direction, is not yet included in our simulation. This will be possible provided the opposing direction segment is part of the subnetwork simulated in the same client of the distributed simulation implementation.

## JAVA IMPLEMENTATION OF SHORT SEGMENTS, LONG SEGMENTS, AND INTERSECTION COMPONENTS

The simulation program reads the information about the simulated map from text files describing the map and the initial state. The map file contains information about the length of the segment, traffic light, next segment, no. of lanes and type of the segment. The initial state file contains information for each segment about the number of vehicles and the time they need to pass through the segment.

All the events are described in an "event list" (like a calendar) including for each event an execution time and a label. The event list is organized as a queue.

### Short segment

For short segments we model traffic as if we would be looking at the each car with a traffic camera. The short segment is defined like this:

$a: b: c: d: e: f: g: h: i: p: pA$ , where:

- o a – Type (for example 0 for short segment);
- o b – nameSegment = a string representing the name of the segment;
- o c – No. lanes;

- o d – Capacity of the segment (this variable give the number of cells);
- o e – Length of the segment;
- o f – direction of traffic;
- o g – traffic light;
- o h – speed (can be changed during the simulation);
- o i – Neighbors  $x, y$ ;
  - $x = \text{string representing the name of the left neighbor}$ ;
  - $y = \text{string representing the name of the right neighbor}$ ;
- o p – Probability that one vehicle from the current segment will drive to one of the next segments (when we have two consecutive segments the probability is 1; in case of an intersection we have different values for probability);
- o pA – parking area; (in case there is a parking area that can be entered from this segment)

The segment is split into several cells – parallel cells containing vehicles driving next to each other in parallel lanes - for updating the location and the speed of the vehicles. We assume that the maximum speed is 50 Km/h and the minimum speed is dictated by the traffic congestion. When vehicles enter a short segment the program first tries to assign them to the first cell of the right lane, and if that is not possible to the first cell in the other lanes.

The algorithm for changing the color of the traffic light could be modified on line, by default a static algorithm is setup (fixed time for green, yellow and red color).

When a vehicle leaves a cell the program determines if the downstream cell is free, and if yes the vehicle moves forward at its current speed (a long vehicle will occupy more than one cell; if the traffic speed is high then we require that at least one cell should be free between two cars). If this downstream cell is not free and if the downstream cell in the neighboring lane is free then the vehicle will change the lane (except when lanes are reserved for turning traffic). A car can't move to the next cell if the next cell is occupied; i.e. only one car can be inside of a cell at a time. This means that the car may have to slow down. In the short segment we consider only two possible values for the speed (maximum speed allowed and a minimum speed). Hence we need to consider only two possible time delays between the time when a vehicle enters a cell and the time the same vehicle leaves this cell.

To obtain sample values of the time delay needed to pass through the short segment we consider a special type of cars (observable cars) that send a message to an observer each time they enter or exit from the segment. This type of car exists only in short segments (here we start a new thread for each car). This approach provides an accurate value for the time needed to pass through the short segment.

The thread technique from java programming language allows several tasks to be executed in parallel even on a single processor computer with a technique call "time slicing" (the time interval is split into several small time intervals and for each slice of the time one active thread

will have the opportunity to be executed). This was implemented in the simulation program presented here.

### Long segment

On this type of segment the amount of information is smaller than in the case of a short segment. Long segments are characterized by modeling of the aggregated traffic flow. Here is implemented a function (exchange function) used to update at every time step (synchronous updating) the number of cars that are driving through the segment. This exchange function represents the amount of cars that want to leave the segment and which of them actually can enter the next segment. All the variables are recalculated every time interval based on the last values.

The long segment is defined as follows:

*type: name: noL: len: cap: traff: nextN: prob: park*

Where:

- type – the type of the segment;
  - 3; in case of the long segment;
- name – a string, representing the name of the segment;
- noL – number of lane;
- len – length of the segment;
- traff – traffic light;
  - 0 if we don't have a traffic light;
  - 1 in case that we have traffic light;
- nextN – a string, representing the name of the next segment;
- prob – the probability that a car from the current segment when leaving the segment enters the next segment;
- park – parking area;
  - 0 if we don't have a parking area;
  - 1 if we have parking area, in this case we have another variable giving the capacity of the parking area;

In case of the long segment we could calculate the average time needed to pass the segment. To transfer cars from one long segment to another long segment we implement the exchange functions:  $E_i$  (the integer number of cars that can leave the downstream segment of the upper segment during one time update interval) and  $E_c$  (the integer number of cars that can enter the segment from the previous segment).

The exchange function ( $E_i$ ) is a random number and can be maximum  $N_i$  and is influenced by the number of cars that can actually enter on the next segment, so  $E_i$  also depends on the response from the next segment.  $E_c$  cars enter the segment from the previous segment. After the program determines the value for  $E_i$  and  $E_c$  function it updates the value for  $N_i$ .

### Intersection

Intersection represents a collection of several short segments with some rules used to "drive" the cars. In case of an intersection without traffic light the priority of the main road is applied and the program also

checks if the gaps between consecutive cars, for each lane that enter on the intersection, is "safe".

The intersection is defined as:

*I: CRS1: 4: a: b: c: d,*

where

CRS1 – name of the intersection;

4 – the number of streets that form the intersection;

a, b, c, d – streets

The street is defined as:

*Str1, n, nameSeg1, nameSeg2, nameSeg3, nameSeg4, nameSeg5*

- Str1 – name of the street
- n – type of the street
  - 0 - forward
  - 1 - forward and right
  - 2 - left, forward and right
  - 3 - left and right
  - 4 - forward and left
- nameSegX (X = 1 ... 5) – name of the segment
  - nope – in case of one way street
  - a string – representing the name

Figure 1 presents an intersection of four streets. The cells  $M_i$  ( $i = 1 \dots 4$ ) represent the "core" of the

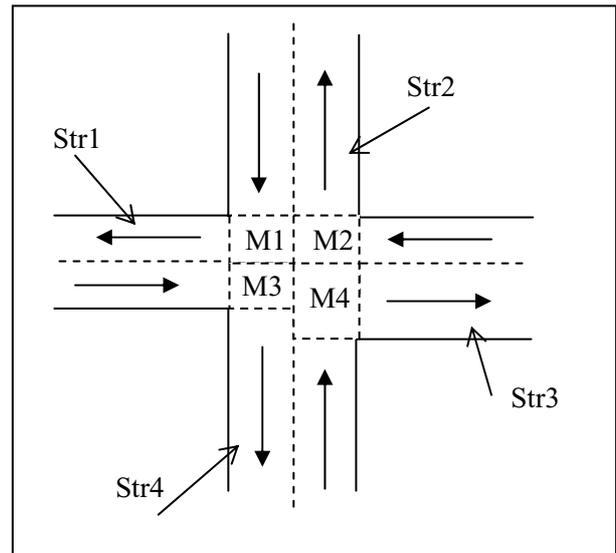


Figure 1. Intersection of four streets

intersection.

Based on information about the topology of the intersection we provide a list describing the interconnection between segments.

For example street "Str 1":

*nameSeg1* – the segment entering the intersection

*nameSeg2* – the segment exiting the intersection

$n = 2$ , means that from *nameSeg1* there are 3 next segments: *nameSeg3*, *nameSeg4* and *nameSeg5*.

In conclusion we have a list with values like:

*nameOfTheSegment* → *vectorOfNextSegments[i]* ( $i \geq 1$  represents the number of possible choices)

The length of "vectorOfNextSegments" vector is variable but it has at least one position even in the case

of a one way street. (e.g.  $segment1 \rightarrow \{nope\}$ , for one way street and  $segment1 \rightarrow \{segmentOne, segmentTwo, segmentThree\}$ )

The "core" of the intersection represents a collection of matrices that are constructed using information about each segment that forms the intersection.

$M_i$  ( $i = 1 \dots 4$ ) – matrix  $n \times m$

$n = SegmX.noOfLanes$

$m = SegmY.noOfLanes$

$M_i[i,j] = 0$  or  $1$

0 – for a free position

1 – Non empty position

### Connecting two segments

This part of the application solves the problem of connecting a short segment or intersection (the event driven microscopic model) with a long segment using a time driven, synchronous, macroscopic model.

1. Short segment – short segment

The output sensor from the first segment is the input sensor from the second segment.

2. Short segment – long segment

In this case we transform the individual cars into an amount of cars. The long segment reads the number of cars leaving the short segment during each state update interval.

3. Long segment – long segment

Here the program implements the exchange function ( $E_i$  and  $E_c$ ).

4. Long segment – short segment

Transform the amount of cars into individual cars. The program implements several algorithms to generate random time interval describing the distance between two consecutive cars entering the short segment. The minimum length of the interval is 1 second. Let  $n$  be the number of cars leaving the long segment during one interval of time  $\Delta$  ( $> n$ ).

#### Algorithm 1

Split the time interval  $\Delta$  into  $n$  equidistant time intervals (larger than 1 second). This algorithm is simple but it cannot represent platoons of cars.

#### Algorithm 2

Repeat until  $n$  cars have been located in the interval of length  $\Delta$ :

*{Generate random number and check if it is not in the safety distance of an already generated and accepted neighboring car}. End*

The advantage of this algorithm is that the time intervals are random but for  $n$  close to  $\Delta$  the computational time is too big.

#### Algorithm 3

This algorithm is a combination of the other two algorithms. First we generate a random number, then we check in the time interval left how many cars can fit in and this is repeated until we generate the times for all the cars.

## CLIENT-SERVER IMPLEMENTATION OF SIMULATION FOR LARGE NETWORKS

The simulation program is distributed which means that we can simulate a large map, each client simulating a part of the map and for the border

segments (the segments that have the next Segment in another client) the client establish the communication with the other client. The border segments must be long segments because this way it's easy to maintain the clock synchronization for all the different clients. Otherwise, if some border segments were short segments we must implement a list (like a "history of the system") to reload it when the system has been "desynchronized" (this situation could appear very often when we have many short segment on the border). This would require extra time for calculation and resynchronization.

The scheme of the simulation program is presented in figure 2.

During the simulation the program can start several

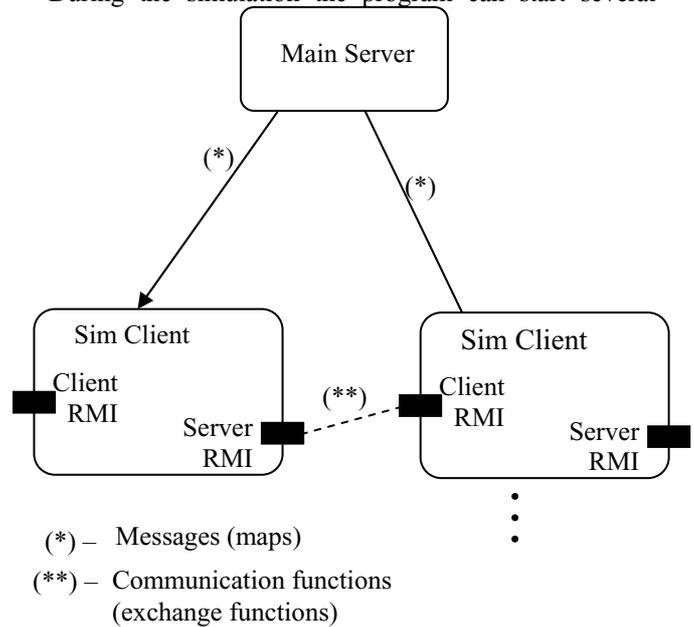


Figure 2. The simulation program

clients and all act the same. First the client must request for a connection, after the communication channel is established the server sends the initial data to start the simulation (several text files), the client send a message in case that everything was OK (receive all the files) and then the communication channel is close.

Each car (in case of the short segment) is simulated using the Thread techniques from java.

```
public class MyCar extends Thread
```

```
{
```

```
    String name;
```

```
public MyCar(String n)
```

```
{
```

```
    name = n;
```

```
}
```

```
public void run()
```

```
{
```

```
    //this method runs when start() is invoked on the thread
```

```
    System.out.println( "car with the name: "+name+", is driving");
```

```
}
```

Each thread must implement the *run()* method which will be invoked during the simulation.

Using threads we could implement concurrent application.

This type of implementation is recommended when we want to simulate several event producers in parallel. The thread technique allow several task to be executed in parallel even on a single processor computer with a technique call “time slicing” (the time interval is split into several small time interval and for each slice of the time one active thread will have the opportunity to be executed). The number of threads that can be started and active at the same time on the system depend on the computer configuration and operating system.

All the method that can be access by several threads must by synchronized (this way we avoid the redundancy in the program) and the threads are “thread safe”.

When the simulation is started each client asks the server for the map, for the initial state of the simulated map and for the list with all the clients (IP address and port).

JavaSoft provides the RMI-IIOP (Remote Method Invocation – Internet Inter – ORB Protocol - set of libraries) which allows Java/RMI applications to use OMG's IIOP instead of Sun's proprietary Java Remote Method Protocol (JRMP). This protocol give the possibility to write Java/RMI Servers that can be accessed not only by regular Java clients, but also from CORBA clients (this clients can be written on different programming language, for ex. C++, ...) and vice versa.

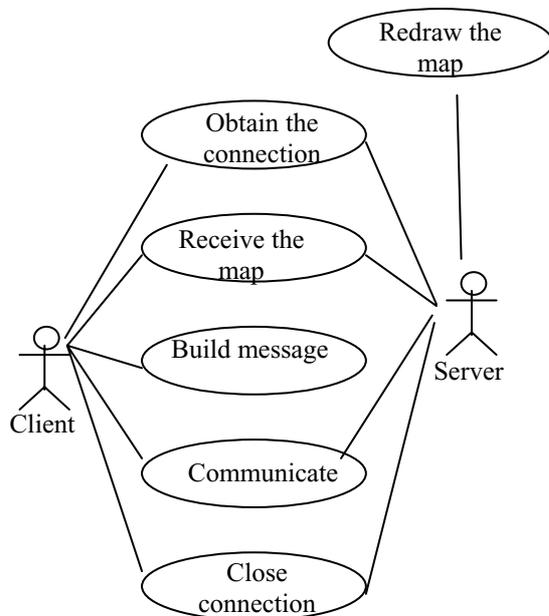


Figure 3. Use case UML diagram for client – server communication

The application is distributed and we use the RMI – IIOP java technique for implementation.

For each simulation client we have a client module and a server module implemented with RMI – IIOP. The client module receive cars from the neighbors (can be one or more) and the server module send the cars to the neighbors. RMI – IIOP allows server objects to receive messages from multiple threads, so they should be thread - safe. RMI – IIOP technique implies that we should implement several java files (Stub and Skel – predefine names) where the communication between the client and the server application is described and the permissions (the client right, for ex.: read, write, all ...).

Use case UML diagram for client server communication (for sending the initial state of the system and the map) is presented in figure 3. This diagram represents the activities of the client object and the interaction with the server object.

For each simulation client the “Main Server” will start a new Thread to treat all the clients in same time.

### SIMULATION RESULTS

At the end of the simulation, the program generates a file with measurement data for each segment and a file with all the events that occurred during the simulation. This data file contains the time, speed and the number of cars per each segment (an average between all the lanes in case of the short segment).

The output data file is used in a Matlab program to draw the time evolution on the segment during the simulated period of time. This graphical representation is not done in java because it is much easier to draw in Matlab. So in java we generate the files and in Matlab we analyze them. For example the simulation result has been used to analyze the delay of the cars during the simulation, because we know the time needed to pass the short segment and for the long segment we know the average speed and the length of the segment. This is sufficient to calculate the time needed to drive from one point on the road map corresponding to the simulation, to any other point on this map. To increase the speed of the simulation, no visual results are printed or drawn during the execution of the simulation. Several situations were simulated (e.g. blocking a lane or an entire segment by reducing the capacity of the segment or by putting a traffic light permanently on red, simulating an incident). The simulation results corresponded to the intuitively expected results, showing e.g. correctly how the blockage was propagated upstream.

In figure 4 is presented a situation simulated with this program, an intersection of 3 streets is considered. All the segments have 2 lanes. The time unit is second. We represent the number of cars (that leave the segment) function of time.

We simulate several situations:

1. (\*) The segment 3 is blocked (both lanes) which implies that the previous segment will be full in the next few seconds. The segment 1 is long segment (the capacity is bigger).

2. Between (\*) and (\*\*) only one lane (from segment 3) is open. From segment 2 the driver could turn to left on the segment 4.
3. (\*\*) again all the lanes from segment 3 are blocked.

4. After (\*\*) at the beginning no constraints are imposed and after a while a speed limitation is imposed on the segment 3.

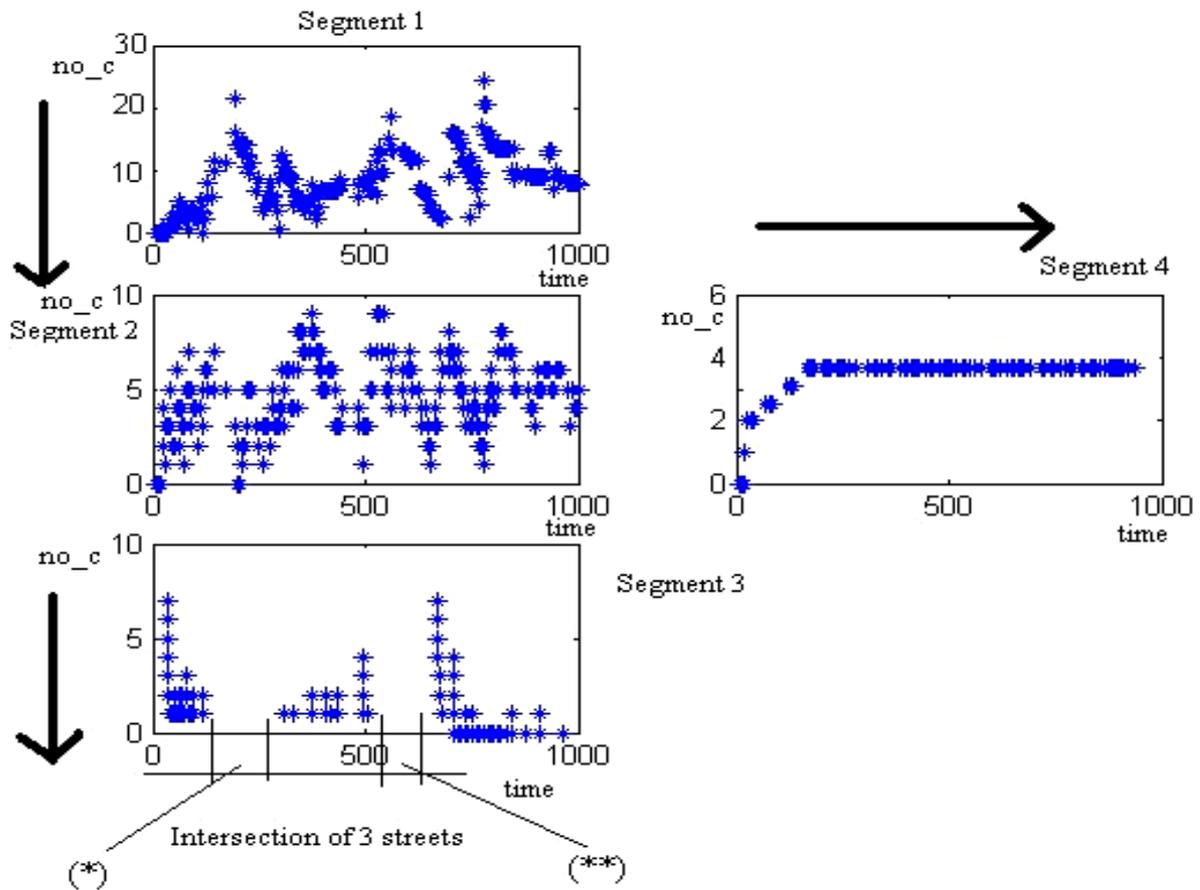


Figure 4. Intersection of 3 streets – simulation result

## REFERENCES

- Boel R., L. Mihaylova, "Modelling Freeway Networks by Hybrid Stochastic Models", *Proc.2004 IEEE Intelligent Vehicle Symposium* (Parma) pp.182-187, 2004.
- Daganzo, C. F., "The Cell Transmission Model", *Transp. Res.B*, 28B, No. 4, pp. 269-287, 1994
- Mihaylova L., R. Boel: "A Particle Filter for Freeway Traffic Estimation", *43rd IEEE CDC*, WeB08.4, 2004
- Neubert, L., L. Santen, A. Schadschneider and M. Schreckenberg, "Single-Vehicle Data of Highway Traffic", *A Statistica, Analysis, Physical Review A*, Vol. 60, No. 6, pp. 6480-6490, 1999.
- Papageorgiou, M. et al., "Freeway Ramp Metering: an Overview", *IEEE-T- Intelligent Transport. Systems*, Vol. 3, No. 4, pp.271-280, 2002.

## ACKNOWLEDGMENT

The work of the first author was supported by a Marie Curie Fellowship of CTS. Part of the research was carried out in the framework of the DWTC Project PODO CP/40 on "Sustainability Effects of traffic management strategies".

## AUTHOR BIOGRAPHIES

**Camelia Avram** received the automation engineering degree from the Technical University of Cluj - Napoca, Romania, in June 2001 and the MSc. Degree from the Technical University of Cluj - Napoca. She is a PhD student to the Technical University of Cluj - Napoca. She was a research fellow to the Gent University for 7 months in 2004.

**Rene K. Boel** received the electro-mechanical engineering degree and the nuclear engineering degree from the Ghent University, Ghent, Belgium, in 1969 and 1970 resp., and the M.Sc. and Ph. D. degrees from the University of California, Berkeley, in 1972 and 1974 resp. His doctoral dissertation on "Optimal control of Jump processes" was supervised by Prof. P. Varaiya. He was a Research Fellow of F.W.O.-Vlaanderen (Flemish Foundation for Scientific Research) until September 2000. Since October 2000 he is an associate professor at the School of Engineering of Ghent University, and at the Department of Mathematics of the Katholieke Universiteit Leuven. He has held temporary appointments at the Department of Electrical Engineering and Computer Sciences,

University of California, Berkeley, at the Department of Technical Cybernetics of the Technical University Trondheim, Norway, at the Department of Mathematics and Statistics, Bell Telephone Laboratories, Murray Hill, at the Department of Computing and Control of Imperial College, London, at the Department of Systems Engineering, Australian National University, Canberra, at the Electrical and Electronic Engineering Department, Melbourne University, and at the Department of Electrical and Computer Engineering, University of Newcastle, N.S.W.

His current research interests are in stochastic control and its applications to communications networks, and in the control of discrete event systems and hybrid systems, with applications to traffic and power systems.