PROCEEDINGS
22nd European Conference on Modelling and Simulation ECMS 2008

June 3rd – 6th, 2008
Nicosia, Cyprus

Edited by:

Loucas S. Louca
Yiorgos Chrysanthou
Zuzana Oplatková
Khalid Al-Begain

Organized by:

[ECMS] European Council for Modelling and Simulation

Technically co-sponsored by:

[IEEE] Institute of Electrical and Electronics Engineers UKRI

Hosted by:

University of Cyprus, Nicosia, Cyprus

International co-sponsors:

ASIM: German speaking Simulation Society
EUROSIM: Federation of European Simulation Societies
PTSK: Polish Society of Computer Simulation
LSS: Latvian Simulation Society

ECMS 2008 ORGANIZATION

General Conference Chair

Loucas S. Louca

University of Cyprus

Cyprus

Conference Co-Chair

Khalid Al-Begain

University of Glamorgan

United Kingdom

Programme Chair

Yiorgos Chrysanthou

University of Cyprus

Cyprus

Programme Co-Chair

Zuzana Oplatková

Tomas Bata University in Zlín

Czech Republic

Chairman of European Council for Modelling and Simulation

Khalid Al-Begain

University of Glamorgan

United Kingdom

Managing Editor

Martina-Maria Seidel

St. Ingbert

Germany

INTERNATIONAL PROGRAM COMMITTEE

Simulation in Industry, Business and Services

Track Chair: **Alessandra Orsoni**
University of Kingston, UK

Program Chair: **Serhiy Kovela**
University of Kingston, UK

Agent-Based Simulation

Track Chair: **Eugène J.H. Kerckhoffs**
TU Delft, The Netherlands

Simulation of Complex Systems

Track Chair: **Krzysztof Amborski**
Warsaw University of Technology, Poland

Program Chair: **Jaroslav Sklenar**
University of Malta, Malta

Simulation of Intelligent Systems

Track Chair: **Lars Nolle**
Nottingham Trent University, UK

Track Co-Chair: **Ivan Zelinka**
Tomas Bata University of Zlín, Czech Republic

Program Chair: **Shane Lee**
University of Wales, UK

Vision and Visualization

Track Chair: **Gerald Schaefer**
Nottingham Trent University, UK

Program Chair: **Marina Chukalina**
Institute of Microelectronics Technology RAS, Russian Federation

Simulation Applications in Industry

Track Chair: **Agostino Bruzzone**
MISS DIP University of Genoa, Italy

Program Chair: **Francesco Longo**
University of Calabria, Italy

Finance and Economics

    Track Chair: **Javier Otamendi**
    University of Rey Juan Carlos Madrid, Spain

    Track Co-Chair: **José Luis Montes**
    University of Rey Juan Carlos Madrid, Spain

Simulation, Experimental Science and Engineering

    Track Chair: **Jan Amborski**
    Institute of Aviation, Poland

    Track Co-Chair: **Rafal Kajka**
    Institute of Aviation, Poland

Bond Graph Modeling and Simulation

    Track Chair: **Loucas S. Louca**
    University of Cyprus, Cyprus

    Track Co-Chair: **Wolfgang Borutzky**
    Univ. of Applied Science St Augustin, Germany

Modelling and Simulation Methodologies

    Track Chair: **Claudia Frydman**
    University Paul Cézanne Aix-Marseille III, France

    Program Chair: **Olivier Dalle**
    University of Nice - Sophia Antipolis and INRIA, France

Discrete Event Modelling and Simulation in Logistics, Transport and Supply Chains

    Track Chair: **Gaby Neumann**
    Otto-von Guericke-University Magdeburg, Germany

    Track Co-Chair: **Yuri Merkuryev**
    Riga Technical University, Latvia

    Program Chair: **Edward J. Williams**
    University of Michigan, USA

VR / AR in Modelling and Simulation

    Track Chair: **Angelos Amditis**
    ICCS-NTUA Athens, Greece

    Track Co-Chair: **Yiorgos Chrysanthou**
    University of Cyprus, Cyprus

Modelling, Simulation and Control of Technological Processes

Track Chair: **Petr Dostál**
Tomas Bata University in Zlín, Czech Republic

Track Co-Chair: **Jiri Vojtesek**
Tomas Bata University in Zlín, Czech Republic

Program Chair: **Frantšiek Gazdoš**
Tomas Bata University in Zlín, Czech Republic

Student Track

Track Chair: **Zuzana Oplatková**
Tomas Bata University in Zlín, Czech Republic

IPC Members in Alphabetical Order

Naci Akkok, Oracle Nordics & University of Oslo, Norway

Luis Antunes, University of Lisbonne, Portugal

Monika Bakosova, Slovak University of Technology in Bratislava, Slovakia

Fernando J. Barros, University of Coimbra, Pólo II, Portugal

Marta Basualdo, University National of Rosario, Argentina

Roland Blach, Fraunhofer Institute, Germany

John Bland, Nottingham Trent University, United Kingdom

Fabian Böttinger, Fraunhofer Institute IPA,Stuttgart, Germany

Hüseyin Kemâl Çakmak, Research Institute Karlsruhe, Germany

Petr Chalupa, Tomas Bata University in Zlín, Czech Republic

Carlos E. Christoffersen, Lakehead University, Ontario, Canada

Antonio Cimino, MSC-LES, Italy

Duilio Cursico, University of Calabria, Italy

Torby Dahl, University of Wales, Newport, United Kingdom

Vjekoslav Damic, University of Dubrovnik, Croatia

Geneviève Dauphin-Tanguy, École Centrale de Lille, France

Alejandro Donaire, University National of Rosario, Argentina

Luis Miguel Doncel, University Rey Juan Carlos Madrid, Spain

Raphaël Duboz, CIRAD, Montpellier, France

Georges Dumont, IRISA, France

Andrzej Dzielinski, Warsaw University of Technology, Poland

Bernard Espinasse, LSIS - University Paul Cezanne, France

Miroslav Fikar, Slovak University of Technology in Bratislava, Slovakia

Marco Gilles, UCL, United Kingdom

Héctor Jorquera Gonzalez, Catholic University of Chile, Chile

Pilar Grau, University Rey Juan Carlos Madrid, Spain

Shafqat Hameed, University of Wales, Newport, United Kingdom

David R.C. Hill, ISIMA, Blaise Pascal University, Clermont-Ferrand, France

Martin Ihrig, University of Pennsylvania, USA

Teruaki Ito, University of Tokushima, Japan

Sergiu Ivanov, University of Crajova, Romania

Gerrit K. Janssens, University of Hasselt, Belgium

Sergio Junco, University National of Rosario, Argentina

Nikolaos V. Karadimas, National Technical University of Athens, Greece

Eugene Kindler, Ostrava University, Czech Republic

Joanna Kolodziej, University of Bielsko-Biala, Poland

Petia Koprinkova, Bulgarian Academy of Sciences, Bulgaria

Fabrice Kordon, LIP6, University Pierre and Marie Curie, Paris, France

Marek Kubalcik, Tomas Bata University in Zlín, Czech Republic

Celine Loscos, UCL/Girona, United Kingdom

Susan Lysecky, University of Arizona, USA

Jerzy Malachowski, Military University of Technology, Warsaw, Poland

Wilfrid Marquis-Favre, National Institute of Sciences Appliances of Lyon, France

Galina Merkuryeva, Riga Technical University, Latvia

Despina Michael, UCY, Cyprus

Giannis Karaseitanidis, NTUA, Greece

Joanna Kolodziej, University of Bielsko-Biala, Poland

Lars Michael Kristensen, University of Aarhus, Denmark

Saurabh Mittal, ACIMS & University of Arizona, Tucson, USA

Daniel Mold, University of Hamburg, Germany

Mike Nachtegael, University of Ghent, Belgium

Tomoharu Nakashima, Osaka Prefecture University, Japan

Libero Nigro, University of Calabria, Italy

Dmitry P. Nikolaev, Russian Academy of Sciences, Russia

Jakub Novak, Tomas Bata University in Zlín, Czech Republic

Suzanne Pinson, University Paris-Dauphine, France

Peter Plassmann, University of Glamorgan, United Kingdom

Martin Quinson, University Henry Poincare & Loria, Nancy, France

Napoleon H. Reyes, Massey University, New Zealand

Mathias Röhl, University of Rostock, Germany

Boris Rohal-Ilkiv, Slovak University of Technology in Bratislava, Slovakia

Ronald C. Rosenberg, Michigan State University, USA

Alessandro Rossi, University of Trento, Italy

Leon Rothkrantz, Delft University of Technology, The Netherlands

Toni Ruohonen, University of Jyväskylä, Finland

Jorge Sainz, University Rey Juan Carlos Madrid, Spain

PREFACE

Continuing the tradition and for the 22nd time the European Council for Modeling and Simulation (ECMS) is organizing the 2008 European Conference on Modeling and Simulation (ECMS 2008). The ECMS annual conference is becoming one of the major events in the area of modeling and simulation by attracting European and International researchers. This is confirmed by the contributions of ECMS 2008 with 82 accepted papers by 197 authors from 34 countries (19 European countries).

ECMS 2008 covers a variety of thematic tracks in the general area of modeling and simulation. These include 13 specialized tracks in addition to 1 student track. The contributed papers include topics from more traditional and well-established tracks, but also cover new areas like the Bond Graph Modeling and Simulation or Virtual and Augmented Reality in Modeling and Simulation.

The conference is held in conjunction with the 2008 High Performance Computing & Simulation Conference (HPCS 2008) and the 15th International Conference on Analytical and Stochastic Modeling Techniques and Applications (ASMTA 2008). HPCS 2008 and ASMTA 2008 are contributing another 60 accepted papers. In addition, during the first day of the conference there will be specialized tutorials given by internationally recognized researchers.

It is a great honor to have internationally recognized researchers to address the attendees of the conference as part of keynote talks that will be the opening sessions during each day of the conference. The three keynote speakers are Professor Geoffrey C. Fox from Indiana University Bloomington, Professor Felix Breitenecker from Vienna University of Technology, and Professor Raymond Marie from IRISA.

The success of this conference would not be possible without the contributions from many key people. We would like to thank all these people and more specifically the authors for their contributed manuscripts, the track chairs for encouraging their colleagues and students to submit their work to the conference, and the members of the IPCs for their valuable comments that helped to maintain the high quality of the accepted papers and the conference in general.

For the organization of the conference we would like to thank Martina-M. Seidel for running the ECMS office and making sure that everything was in place in order to have a smooth and problem free conference. Also, we would like to thank Nora Hadjisoteriou for her support in all the local arrangements. Last but not least we would like to thank the University of Cyprus for providing free of charge the facilities and support for all the sessions of the conference.

We wish the attendees of the conference to have many fruitful and interesting discussions, to meet new people and start new collaborations, and to form new research questions after attending the technical sessions. We look forward to seeing you (and some more of your colleagues) again in ECMS 2009 in Madrid

<div align="center">

Loucas S. Louca
General Conference Chair

Khalid Al-Begain
General Conference Co-Chair

Yiorgos Chrysanthou
General Program Chair

Zuzana Oplatková
General Program Co-Chair

</div>

TABLE OF CONTENTS

PLENARY PAPER

SIMULATION IN INDUSTRY, BUSINESS AND SERVICES

AGENT-BASED SIMULATION

SIMULATION OF COMPLEX SYSTEMS

SIMULATION OF INTELLIGENT SYSTEMS

VISION AND VISUALIZATION

SIMULATION APPLICATIONS IN INDUSTRY

MODELLING AND SIMULATION METHODOLOGIES

FINANCE AND ECONOMICS

SIMULATION, EXPERIMENTAL SCIENCE AND ENGINEERING

DISCRETE EVENT MODELLING AND SIMULATION IN LOGISTICS TRANSPORT AND SUPPLY CHAINS

BOND GRAPH MODELLING AND SIMULATION

MODELLING, SIMULATION AND CONTROL OF TECHNOLOGICAL PROCESSES

VR / AR IN MODELLING AND SIMULATION

STUDENT TRACK

ECMS 2008
SCIENTIFIC PROGRAM

Plenary Paper

DEVELOPMENT OF SIMULATION SOFTWARE - FROM SIMPLE ODE MODELLING TO STRUCTURAL DYNAMIC SYSTEMS

Felix Breitenecker
Institute for Analysis and Scientific Computing, Dept. Mathematical Modelling and Simulation
Vienna University of Technology
1040 Vienna, Austria
E-mail: Felix.Breitenecker@tuwien.ac.at

KEYWORDS

Simulation software, simulation systems, object-oriented simulation, CSSL standard, Modelica standard, state space approach, dynamic structures, UML, physical modelling, structural dynamic systems

ABSTRACT

This contribution presents development and trends of simulation software, from the simple structures for 'static' explicit ODE models to modelling of structural dynamic systems with DAEs. Simulation emerged in the 1960' in order to be able to analyse nonlinear dynamic system and to synthesize nonlinear control systems. Since that time simulation as problem solving tool has been developed towards the third pillar of science (beneath theory and experiment), and simultaneously simulation software has been developed further on.

The paper first follows roots in the CSSL standard for simulation languages, from simple ODE modelling structures to discrete elements in ODE modelling, using the classical state space approach. Next, the extensions from explicit state space description to implicit model descriptions and their consequences for numerical algorithms and for structure of simulators are discussed, like DAE solvers and implicit model translation. Besides DAE modelling, state event description and state event handling has become a key feature for simulators – sketched by a state event classification and options for implementation.

In the following, the last major steps of the development are presented: a-causal physical modelling, the new Modelica standard for ODE and DAE modelling, state chart and structural dynamic systems. Physical modelling and Modelica is outlined by examples, and for structural dynamic systems a new approach by means of internal and external events is presented – together comfortable state chart descriptions based on UML-RT.

The last section reviews state-of-the-art simulators for availability of extended and structural features necessary for these last developments: DAE modelling, a-causal physical modelling, state events, Modelica modelling, state chart modelling, structural decomposition for structural dynamic systems, and related features. At the end, a table summarises and compares the availability of structural approaches and features.

CSSL STRUCTURE IN CONTINUOUS SIMULATION

Simulation supported various developments in engineering and other areas, and simulation groups and societies were founded. One main effort of such groups was to standardise digital simulation programs and to work with a new basis: not any longer simulating the analog computer, but a self-standing structure for simulation systems. There were some unsuccessful attempts, but in 1968, the CSSL Standard became the milestone in the development: it unified the concepts and language structures of the available simulation programs, it defined a structure for the model, and it describes minimal features for a runtime environment.

The CSSL standard suggests structures and features for a model frame and for an experimental frame. This distinction is based on Zeigler's concept of a strict separation of these two frames. Model frame and experimental frame are the user interfaces for the heart of the simulation system, for the simulator kernel or simulation engine. A translator maps the model description of the model frame into state space notation, which is used by the simulation engine solving the system governing ODEs. This basic structure of a simulator is illustrated in Figure 1; an extended structure with service of discrete elements is given in Figure 3.

In principle, in CSSL's model frame, a system can be described in three different ways, as an interconnection of blocks, by mathematical expressions, and by conventional programming constructs as in FORTRAN or C. Mathematical basis is for the simulation engine is the state space description

$$\dot{\vec{x}}(t) = \vec{f}(\vec{x}(t), \vec{u}(t), t, \vec{p}), \quad \vec{x}(t_0) = \vec{x}_0,$$

which is used by the ODE solvers of the simulation engine. Any kind of textual model formulation, of graphical blocks or structured mathematical description or host languages constructs must be transformed to an internal state equation of the structure given above, so that the vector of derivatives $\vec{f}(\vec{x}, \vec{u}, t, \vec{p})$ can be calculated for a certain time instant $\vec{f}_i = \vec{f}_i(\vec{x}(t_i), \vec{u}(t_i), t_i, \vec{p})$. This vector of derivates is fed into an ODE solver in order to calculate a state update $\vec{x}_{i+1} = \Phi(\vec{x}_i, \vec{f}_i, h)$, h stepsize (all controlled by the simulation engine).
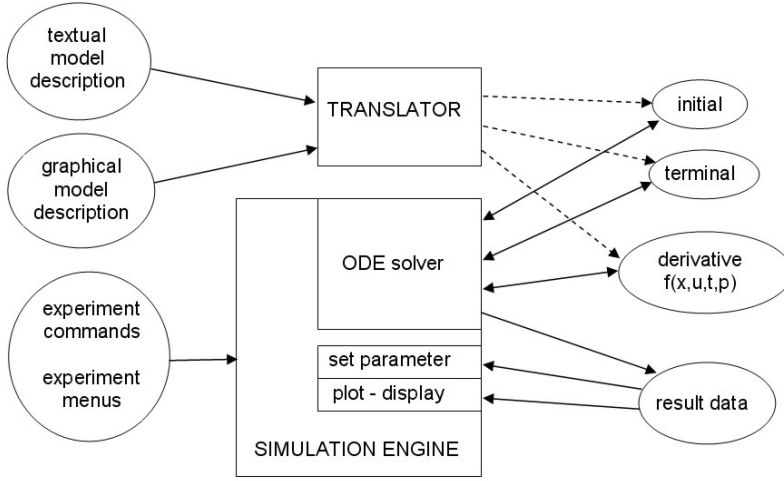
Figure 1: Basic Structure of a Simulation Language
due to CSSL Standard

Essential is CSSL's concept of SECTIONs or RE-GIONs, giving a certain structure to the model description. First, CSSL defines a set of operators like INTEG, which formulates parts of the state space description for the system governing ODEs. Other memory operators like DELAY for time delays, TABLE functions for generating (technical) tables, and transfer functions complete dynamic modelling parts. The dynamic model description builds up the DYNAMIC or DERIVATIVE section of the model description. Mapping the model description onto state space description, requires automatic sorting of the equations (blocks) to proper order of the calculation – an essential feature of the translator.

Sometimes together with the state space equations we also meet parameter equations, parameter dependent initial values, and calculations with the terminal values (e.g. for cost functions in an optimisation). In principle, all this calculations could be done in the dynamic model description, but then they are calculated at each evaluation of the derivative vector of the ODE solver – although they have to be calculated only once.

As example, we consider the model description for a pendulum. The well-known equations (length l, mass m, and damping coefficient d) and initial values, parameter and static relations and dependencies are given by

$$\ddot{\varphi}(t) = -\frac{g}{l}\sin\varphi - \frac{d}{m}\dot{\varphi}, \quad \varphi_0 = \frac{\pi}{n}, \quad \dot{\varphi}_0 = 0,$$

$$d = \sqrt{2D}, \ a = \frac{g}{l}, b = \frac{d}{m}; \quad \varphi^{\deg}(t_f) = \frac{\pi}{180} \ \varphi(t_f)$$

A structured model description, e.g. in ACSL (Table 1), generates more efficient code: only the DERIVATIVE section is translated into the derivative vector function, while INITIAL and TERMINAL section are translated into functions called evaluated separately only once.

It is task of the translator, to recognise the static elements, and to sort them separately from the dynamic equations, so that for the simulation engine dynamic equations (derivative), initial and parameter equations (initial), and terminal equations (terminal), are provided in separate modules.

With graphical window systems, graphical model descriptions became important. Here the roots go back on the one side to analog computation using patching diagrams, and on the other side to control techniques with signal flow diagrams. Consequently, simulation systems offered this kind of model description, either as stand-alone model frame, or as extension.

Table 1: ACSL Structured Textual
Model Description

```
PROGRAM math_pendulum
! --- structured CSSL model --------------------
! --- model parameters --------------------------
  CONSTANT m=1, l=1, d=0.3 ! kg, m, N*s/m
  CONSTANT g=9.81, pi=3.141592653; dphi0=0
  CONSTANT pintel=2
INITIAL ! calculation with parameters-----------
    phi0 = pi/pintel; a = g/l; b = d/m
END ! of INITIAL -------------------------------
  DERIVATIVE ! ODE model ---------------------
    phi = integ ( dphi,        phi0)
    dphi = integ (-gdl*sin(phi)-ddm*dphi, dphi0)
  END ! of DERIVATIVE -------------------------
TERMINAL ! calculations with final states ------
    phi_grad = phi*180/pi
END ! of TERMINAL ------------------------------
END ! of Program -------------------------------
```

However, in a graphical modelling system one disadvantage appears: the graphical structure consisting of directed dynamic signal flow allows almost no structure for dynamic calculations and static calculations. Calculation of static parameter equations are modelled by dynamic blocks – consequently evaluated at each evaluation call of the ODE solver. Figure 2 shows the SIMULINK model for Pendulum example; where one finds the static calculation of d/m as dynamic block in the block-oriented signal flow.

From 2000 on, SIMULINK's graphical model frames were enriched by structures - triggered subsystems can be used for such purposes (Figure 2, triggered subsystems for static calculation of g/l shown in subwindow).

In general, the *experimental frame* has to set parameters, it has to control and perform the "simulation" of the model, and it should support documentation of the results. In the CSSL standard, minimal requirements are availability of certain ODE Solvers (Euler, RK4, RK-Fehlberg, and Gear or BDF algorithms for stiff systems), change of parameters, and documentation of results in a plotting system.
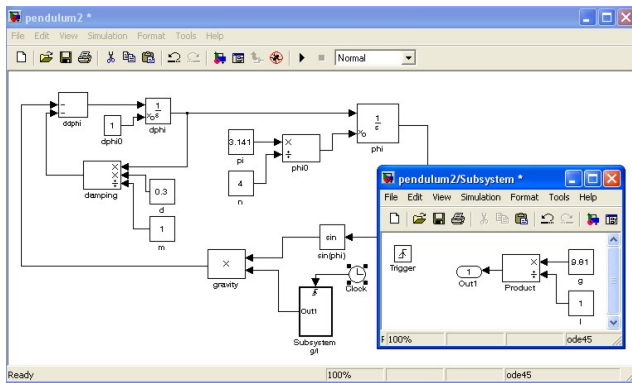
Figure 2: Graphical Model Description of
Pendulum in SIMULINK

From 1980 on all simulation languages tried to meet the CSSL standard. But the implementations were and are different. First, the structure with sections or regions can be given explicitly by definition of these sections, semi-implicitly by type declarations of all variables, and full implicitly – depending on the translator. For instance, SIMNON defines parameters, states, and derivatives, so that for sorting no explicit model structure is necessary; in case of pendulum these definitions are:

```
parameter m, l, g, …….
state      phi,  phidot
derivative dphi, ddphi
```

There are many different ways to implement a simulation system. One meets various mixtures of compiled and interpreted implementations, as well as on the one side strict distinctions of model frame and experimental frame, and on the other hand definitions of model frame and experimental frame in a common deck for compilation. Clearly, there are advantages and disadvantages in all kinds of implementation. Generally holds: the more compiled, the faster, but the more inflexible for changes, and the more interpreted, the slower, but the more flexible for changes.

IMPLICIT MODELS – DIFFERENTIAL-ALGEBRAIC EQUATIONS

For a long time the explicit state space description
$$\dot{\vec{x}}(t) = \vec{f}(\vec{x}(t), \vec{u}(t), t, \vec{p}), \quad \vec{x}(t_0) = \vec{x}_0$$

played the dominant role; additional constraints and implicit models had to be transformed 'manually'. From the 1990s on, the simulators started to take care on these very natural phenomena of implicit structures. Consequently, they started to deal with implicit state space descriptions and constraints, in general with so-called DAE models (differential algebraic equations):

$$F(\dot{\vec{y}}(t), \vec{y}(t), \vec{u}(t), t, \vec{p}) = \vec{0} \quad \vec{y}(t_0) = \vec{y}_0$$

The so-called extended state vector $\vec{y}(t)$ can be splitted into the differential state vectors $\vec{x}(t)$ and into the algebraic state vector $\vec{z}(t)$:

$$\dot{\vec{x}}(t) = \vec{f}(\vec{x}(t), \vec{z}(t), \vec{u}(t), t, \vec{p}) = 0, \quad x(t_0) = \vec{x}_0,$$
$$g(\vec{x}(t), \vec{z}(t), \vec{u}(t), t, \vec{p}) = 0$$

DAE Solvers – DAE Index

The above given DAEs can be solved by extended ODE solvers and by implicit DAE solvers. Three different approaches may be used:

i)   *Nested Approach*, using classical ODE solver
   a. given x_n, solving first numerically
       $$g(x_n, z_n) = 0 \Rightarrow z_n = z_n(x_n) = \hat{g}^{-1}(x_n),$$
       e. g. by modified Newton iteration, and
   b. applying ODE method, evolving
       $$x_{n+1} = \Phi_E(x_n, z_n(x_n), t_n).$$

ii) *Simultaneous Approach*, using an implicit DAE solver;
   given x_n, solving $g(x_{n+1}, z_{n+1}) = 0$ and
   $\Phi_I(x_{n+1}, x_n, z_{n+1}, t_{n+1}) = 0$ simultaneously.

iii) *Symbolic Approach*, determining in advance the explicit form solving
   $g(x, z) = 0 \Rightarrow z = z(x) = \hat{g}^{-1}(x)$ by symbolic computations e.g. within the model translator, and using classical ODE solvers.

The *Symbolic Approach* requires a symbolic inversion of the algebraic equations, which in many cases is not possible or not adequate; furthermore the model translator must not only sort equations, it must be able to perform symbolic manipulations on the equations.

The *Nested Approach* – up to now most commonly used – requires a numerical inversion of the algebraic equations: each evaluation of the vector of derivatives (called by the ODE solver) has to start an iterative procedure to solve the algebraic equation. This approach can be very expensive and time-consuming due to these inner iterations. Here classical ODE solvers can be used.

The *Simultaneous Approach* requires an implicit ODE solver – usually an implicit stiff equation solver. Although also working with iterations, these solvers show much more efficiency and provide more flexibility for modelling (DASSL, IDA-DASSL, and LSODE – solvers).

However, hidden is another problem: the 'DAE index' problem. Roughly speaking, a DAE model is of index *n*, if n differentiations of the DAE result in an ODE system (with an increased state space). The implicit ODE solvers for the *Simultaneous Approach* guarantee convergence only in case of DAE index $n = 1$. Models with higher DAE index must / should be transformed to models with DAE index $n = 1$. This transformation is based on symbolic differentiation and symbolic manipulation of the high index DAE system, and there is no unique solution to this index reduction. The perhaps most efficient procedure is the so-called *Pantelides Algorithm*.

Unfortunately, in case of mechanical systems modelling and in case of process technology modelling indeed DAE models with DAE index $n = 3$ may occur, so that index reduction may be necessary. Index reduction is a new challenge for the translator of simulators, and still point of discussion.

In graphical model descriptions, implicit model structures are known since long time as algebraic loops: the directed graph of signals has one or more signal feedback loops without any memory operator (integrator, delay, etc). Again, in evaluating the problem of sorting occurs, and the model translator cannot build up the sequence for calculating the derivative vector. Some simulators, e.g. SIMULINK, recognise algebraic loops and treat them as implicit models. When a graphical model contains an algebraic loop, SIMULINK calls a loop solving routine at each time step - SIMULINK makes use of the *Nested Approach* described before. This procedure works well in case of models with DAE index $n = 1$, for higher index problems may occur.

In object-oriented simulation systems, like in Dymola, physical a-causal modelling plays an important role, which results in DAEs with sometimes higher index. These systems put emphasis on index reduction (in the translator) to DAEs with index $n = 1$ in order to apply implicit ODE solvers (*Simultaneous Approach*

DISCRETE ELEMENTS IN CONTINUOUS SIMULATION

The CSSL standard also defines segments for discrete actions, first mainly used for modelling discrete control. So-called DISCRETE regions or sections manage the communication between discrete and continuous world and compute the discrete model parts.

In graphical model description, discrete controllers and the time delay could be modelled by a **z-transfer** block. New versions of e.g. SIMULINK and Scicos offer for more complex discrete model parts triggered submodels, which can be executed only at one time instant, controlled by a logical trigger signal.

For incorporating discrete actions, the simulation engine must interrupt the ODE solver and handle the event. For generality, efficient implementations set up and handle event lists, representing the time instants of discrete actions and the calculations associated with the action, where in-between consecutive discrete actions the ODE solver is to be called.

In order to incorporate DAEs and discrete elements, the simulator's translator must now extract from the model description the dynamic differential equations (derivative), the dynamic algebraic equations (algebraic), and the events (event i) with static algebraic equations and event time, as given in Figure 3 (extended structure of a simulation language due to CSSL standard). In principle, initial equations, parameter equations and terminal equations (initial, terminal) are special cases of events at time $t = 0$ and terminal time. Some simulators make use of a modified structure, which puts all discrete actions into one event module, where CASE - constructs distinguish between the different events.

State Events in Continuous Models

Much more complicated, but defined in CSSL, are the so-called state events. Here, a discrete action takes place at a time instant, which is not known in advance, it is only known as a function of the states.

As example, we consider the pendulum with constraints (*Constrained Pendulum*). If the pendulum is swinging, it may hit a pin positioned at angle φ_p with distance l_p from the point of suspension. In this case, the pendulum swings on with the position of the pin as the point of rotation. The shortened length is $l_s = l - l_p$. and the angular velocity $\dot{\varphi}$ is changed at position φ_p from $\dot{\varphi}$ to $\dot{\varphi} l / l_s$, etc. These discontinuous changes are state events, not known in advance.

For such state events, the classical state space description is extended by the so-called state event function $h(\vec{x}(t), \vec{u}(t), \vec{p})$, the zero of which determines the event:
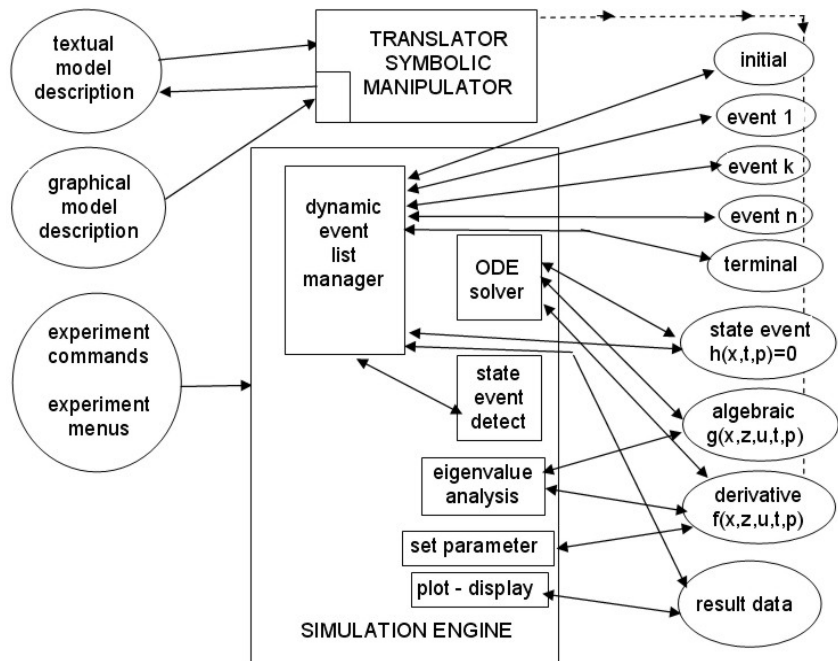
Figure 3: Extended Structure of a Simulation System due to Extensions of the CSSL Standard with Discrete Elements and with DAE Models.

$$\vec{x}(t) = \vec{f}(\vec{x}(t), \vec{u}(t), \vec{p}, t),$$
$$h(\vec{x}(t), \vec{u}(t), \vec{p}, t) = 0$$

In this notation, the model for *Constrained Pendulum* is given by

$$\dot{\varphi}_1 = \varphi_2, \quad \dot{\varphi}_2 = -\frac{g}{l}\sin\varphi_1 - \frac{d}{m}\varphi_2,$$
$$h(\varphi_1, \varphi_2) = \varphi_1 - \varphi_p = 0$$

The example involves two different events: change of length parameter (SE-P), and change of state (SE-S), i.e. angle velocity).

Generally, state events (SE) can be classified in four types:

Type 1 – parameters change discontinuously (**SE-P**),

Type 2 - inputs change discontinuously (**SE-I**),

Type 3 - states change discontinuously (**SE-S**), and

Type 4 - state vector dimension changes (**SE-D**), including total change of model equations.

State events type 1 (SE-P) could also be formulated by means of IF-THEN-ELSE constructs and by switches in graphical model descriptions, without synchronisation with the ODE solver. The necessity of a state event formulation depends on the accuracy wanted. Big changes in parameters may cause problems for ODE solvers with stepsize control.

State events of type 3 (SE-S) are essential state events. They must be located, transformed into a time event, and modelled in discrete model parts.

State events of type 4 (SE-D) are also essential ones. In principle, they are associated with hybrid modelling: models following each other in consecutive order build up a sequence of dynamic processes. And consequently, the structure of the model itself is dynamic; these so-called structural dynamic systems are at present (2008) discussion of extensions to Modelica, see next chapters.

State Event Handling

The handling of a state event requires four steps:

i.   Detection of the event, usually by checking the change of the sign of *h(x)* within the solver step over $[t_i, t_{i+1}]$

ii. Localisation of the event by a proper algorithm determining the time t^* when the event occurs and performing the last solver step over $[t_i, t^*]$

iii. Service of the event: calculating / setting new parameters, inputs and states; switching to new equations

iv. Restart of the ODE solver at time t^* with solver step over $[t^* = t_{i+1}, t_{i+2}]$

State events are facing simulators with severe problems. Up to now, the simulation engine had to call independent algorithms, now a root finder for the state event function *h* needs results from the ODE solver, and the ODE solver calls the root finder by checking the sign of *h*.

For finding the root of the state event function *h(x)*, either interpolative algorithms (MATLAB/Simulink) or iterative algorithms are used (ACSL, Dymola).

Figure 3 (extended structure of a simulation language due to CSSL standard) also shows the necessary extensions for incorporating state events. The simulator's translator must extract from the model description additionally the state event functions (state event j) with the associated event action – only one state event shown in the figure). In the simulator kernel, the static event management must be made dynamically: state events are dynamically handled and transformed to time events. In principle, the kernel of the simulation engine has become an event handler, managing a complex event list with feedbacks. It is to be noted, that different state events may influence each other, if they are close in time – in worst case, the event finders run in a deadlock.

Again, modified implementations are found. It makes sense to separate the module for state event function and the module for the associated event – which may be a single module, or which may be put into a common time event module.

In case of a structural change of the system equations (state event of type 4 – SE-D), simulators usually can manage only fixed structures of the state space. The technique used is to 'freeze' the states that are bound by conditions causing the event. In case of a complete change of equations, both systems are calculated together, freezing one according to the event.

One way around is to make use of the experimental frame: the simulation engine only detects and localises the event, and updates the system until the event time. Then control is given back to the experimental frame. The state event is now serviced in the experimental frame, using features of the environment. Then a new simulation run is restarted (modelling of the structural changes in the experimental frame).

Table 2: *Constrained Pendulum*: Continuous Model with State Events (ACSL)

```
PROGRAM constrained pendulum
CONSTANT m = 1.02, g = 9.81, d =0.2
CONSTANT lf=1, lp=0.7
DERIVATIVE dynamics
  ddphi = -g*sin(phi)/l - d*dphi/m
  dphi  = integ ( ddphi, dphi0)
  phi   = integ ( dphi, phi0)
  SCHEDULE hit   .XN. (phi-phip)
  SCHEDULE leave .XP. (phi-phip)
END ! of dynamics

DISCRETE hit
  l = ls; dphi = dphi*lf/ls
END ! of hit

DISCRETE leave
  l = lf; dphi = dphi*ls/lf
END ! of leave

END ! of constrained pendulum
```

The *Constrained Pendulum* example involves a state event of type 1 (SE-P) and type 3 (SE-S). A classical ACSL model description works with two discrete sections `hit` and `leave`, representing the two different modes, both called from the dynamic equations in the derivative section (Table 2).

Dymola defines events and their scheduling implicitly by WHEN – or IF - constructs in the dynamic model description, in case of the discussed example e.g. by

```
WHEN phi-phip=0 AND phi>phip
THEN l = ls; dphi = dphi*lf/ls
```

In case of more complex event descriptions, the WHEN – or IF – clauses are put into an ALGORITHM section similar to ACSL's DISCRETE section.

In graphical model descriptions, we again are faced with the problem that calculations at discrete time instants are difficult to formulate. For the detection of the event, SIMULINK provides the `HIT CROSSING` block (in new Simulink version implicitly defined). This block starts state event detection (interpolation method) depending on the input, the state event function, and outputs a trigger signal, which may call a triggered subsystem servicing the event.

Extended Features of Simulators

Discrete elements with time events and state events and DAEs may change the structure of the model. Explicit type-4 state events (SE-D) and implicit algebraic conditions in DAEs may change the model essentially or may make a need for a different model. In mechanical systems, these changes are equivalent to a change in the degree of mechanical or physical freedom.
Event description (ED), state event handling (SEH) and DAE support (DAE) with or without index reduction (IR) became desirable structural features of simulators, supported directly or indirectly. Table 3 compares the availability of these features in the MATLAB / Simulink System, in ACSL and in Dymola.

Table 3: Comparison of Simulators' Extended Features (Event Handling and DAE Modelling)

	MS - Model Sorting	ED - Event Description	SHE - State Event Handling	DAE - DAE Solver	IR - Index Reduction
MATLAB	no	no	(yes)	(yes)	no
Simulink	yes	(yes)	(yes)	(yes)	no
MATLAB / Simulink	yes	yes	yes	(yes)	no
ACSL	yes	yes	yes	yes	no
Dymola	yes	yes	yes	yes	yes

In Table 3, the availability of features is indicated by 'yes' and 'no'; a 'yes' in parenthesis '(yes)' means, that the feature is complex to use. MS - 'Model Sorting', is a standard feature of a simulator – but missing in MATLAB (in principle, MATLAB cannot be called a simulator). On the other hand, MATLAB's ODE solvers offer limited features for DAEs (systems with mass matrix) and an integration stop on event condition, so that SHE and DAE get a ('yes'). In Simulink, event descriptions are possible by means of triggered subsystems, so that ED gets a '(yes)' because of complexity. A combination of MATLAB and Simulink suggest putting the event description and handling at MATLAB level, so that ED and SHE get both a 'yes'. DAE solving is based on modified ODE solvers, using the nested approach (see before), so DE gets only a '(yes)' for all MATLAB/Simulink combinations.

ACSL is a classical simulator with sophisticated state event handling, and since version 10 (2001) DAEs can be modelled directly by the residuum construct, and they are solved by the DASSL algorithm (a well-known direct DAE solver, based on the simultaneous approach), or by modified ODE solvers (nested approach) – so 'yes' for ED, SHE, and DAE. In case of DAE index $n = 1$, the DASSL algorithm guarantees convergence, in case of higher index integration may fail. ACSL does not perform index reduction (IR 'no').

Dymola is a modern simulator, implemented in C, and based on physical modelling. Model description may be given by implicit laws, symbolic manipulations extract a proper ODE or DAE state space system, with index reduction for high index DAE systems – all extended features are available. Dymola started a new area in modelling and simulation of continuous and hybrid systems.

FROM CSSL TO MODELICA AND VHDL-AMS

In the 1990s, many attempts have been made to improve and to extend the CSSL structure, especially for the task of mathematical modelling. The basic problem was the state space description, which limited the construction of modular and flexible modelling libraries. Two developments helped to overcome this problem. On modelling level, the idea of physical modelling gave new input, and on implementation level, the object-oriented view helped to leave the constraints of input/output relations.

In physical modelling, a typical procedure for modelling is to cut a system into subsystems and to account for the behaviour at the interfaces. Balances of mass, energy and momentum and material equations model each subsystem. The complete model is obtained by combining the descriptions of the subsystems and the interfaces. This approach requires a modelling paradigm different to classical input/output modelling. A model is considered as a constraint between system variables, which leads naturally to DAE descriptions. The approach is very convenient for building reusable model libraries.

In 1996, the situation was thus similar to the mid 1960s when CSSL was defined as a unification of the techniques and ideas of many different simulation programs. An international effort was initiated in September 1996 for bringing together expertise in object-oriented physical modelling (port based modelling) and defining a modern uniform modelling language – mainly driven by the developers of Dymola.

The new modelling language is called *Modelica*. Modelica is intended for modelling within many application domains such as electrical circuits, multibody systems, drive trains, hydraulics, thermodynamical systems, and chemical processes etc. It supports several modelling formalisms: ordinary differential equations, differential-algebraic equations, bond graphs, finite state automata, and Petri nets etc. Modelica is intended to serve as a standard format so that models arising in different domains can be exchanged between tools and users.

Modelica is a not a simulator, Modelica is a modelling language, supporting and generating mathematical models in physical domains. When the development of Modelica started, also a competitive development, the extension of VHDL towards VHDL-AMS was initiated. Both modelling languages aimed for general-purpose use, but VHDL-AMS mainly addresses circuit design, and Modelica covers the broader area of physical modelling; modelling constructs such as Petri nets and finite automata could broaden the application area, as soon as suitable simulators can read the model definitions.

Modelica offers a textual and graphical modelling concept, where the connections of physical blocks are bidirectional physical couplings, and not directed flow.

An example demonstrates how drive trains are modelled. The drive train consists of four inertias and three clutches, where the clutches are controlled by input signals (Figure 4). The graphical model layout corresponds with a textual model representation, shown in Table 4 (abbreviated, simplified).
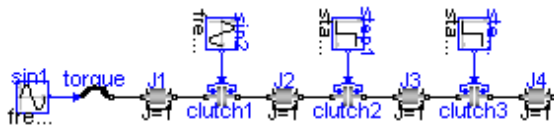
Figure 4: Graphical Modelica Model for Coupled Clutches

Table 4: Textual Modelica Model for Coupled Clutches

```
encapsulated model CoupledClutches; "Drive train"
  parameter SI.Frequency freqHz=0.2; ….
  Rotational.Inertia J1(J=1,phi(ic=0),w(ic=10));
  Rotational.Torque torque;
  Rotational.Clutch clutch1(peak=1.1, fn_max=20);
  Rotational.Inertia J3(J=1); .......................................
equation
  connect(sin1.outPort, torque.inPort);
  connect(torque.flange_b, J1.flange_a);
  connect(J1.flange_b, clutch1.flange_a);
        ...........................................
  connect(step2.outPort, clutch3.inPort);
end CoupledClutches;
```

The translator from Modelica into the target simulator must not only be able to sort equations, it must be able to process the implicit equations symbolically and to perform DAE index reduction (or a way around).

Up to now – similar to VHDL-AMS – some simulation systems understand Modelica (2008; generic – new simulator with Modelica modelling, extension - Modelica modelling interface for existing simulator):

- Dymola from Dynasim (generic),
- MathModelica from MathCore Engineering (generic)
- SimulationX from ISI (generic/extension)
- Scilab/Scicos (extension)
- MapleSim (extension, announced)
- Open Modelica - since 2004 the University of Lyngby develops an provides an open Modelica simulation environment (generic),
- Mosilab - Fraunhofer Gesellschaft develops a generic Modelica simulator, which supports dynamic variable structures (generic)
- Dymola / Modelica blocks in Simulink

As Modelica also incorporates graphical model elements, the user may choose between textual modelling, graphical modelling, and modelling using elements from an application library. Furthermore, graphical and textual modelling may be mixed in various kinds. The minimal modelling environment is a text editor; a comfortable modelling environment offers a graphical modelling editor.

The *Constrained Pendulum* example can be formulated in Modelica textually as a physical law for angular acceleration. The event with parameter change is put into an algorithm section, defining and scheduling the parameter event SE-P (Table 5). As instead of angular velocity, the tangential velocity is used as state variable, the second state event SE-S 'vanishes'.

Table 5: Textual Modelica Model for
Constrained Pendulum

```
equation /*pendulum*/
  v = length*der(phi);
  vdot = der(v);
  mass*vdot/length + mass*g*sin(phi)
  +damping*v = 0;
algorithm
  if (phi<=phipin) then length:=ls; end if;
  if (phi>phipin) then length:=l1; end if;
```

Modelica allows combining textual and graphical modelling. For the *Constrained Pendulum* example, the basic physical dynamics could be modelled graphically with joint and mass elements, and the event of length change is described in an algorithm section, with variables interfacing to the predefined variables in the graphical model part (Figure 5).

```
algorithm
if (revolute1.phi
        <= phipin then
        revolute1.length:=ls;
end if;
if (revolute1.phi
        < phipin then
        revolute1.length:=ll;
end if;
```
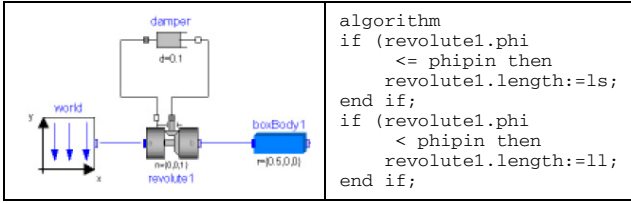
Figure 5: Mixed Graphical and Textual Dymola
Model for *Constrained Pendulum*

MODELLING WITH STATE CHARTS

In the end of the 1990s, computer science initiated a new development for modelling discontinuous changes. The *Unified Modelling Language* (UML) is one of the most important standards for specification and design of object oriented systems. This standard was tuned for real time applications in the form of a new proposal, *UML Real-Time* (UML-RT). By means of UML-RT, objects can hold the dynamic behaviour of an ODE.

In 1999, a simulation research group at the Technical University of St. Petersburg used this approach in combination with a hybrid state machine for the development of a hybrid simulator (*MVS*), from 2000 on available commercially as simulator *AnyLogic*. The modelling language of AnyLogic is an extension of UML-RT; the main building block is the *Active Object*. Active objects have internal structure and behaviour, and allow encapsulating of other objects to any desired depth. Relationships between active objects set up the hybrid model.

Active objects interact with their surroundings solely through boundary objects: ports for discrete communication, and variables for continuous communication (Figure 6). The activities within an object are usually defined by state charts (extended state machine). While discrete model parts are described by means of state charts, events, timers and messages, the continuous model parts are described by means of ODEs and DAEs in CSSL-type notation and with state charts within an object.

An AnyLogic implementation of the well-known *Bouncing Ball* example shows a simple use of state chart modelling (Figure 7). The model equations are defined in the active object ball, together with the state chart ball.main. This state chart describes the interruption of the state flight (without any equations) by the event bounce (SE-P and SE-S event) defined by condition and action.
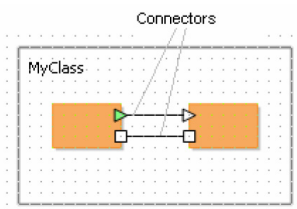
Figure 6: Active Objects with Connectors - Discrete
Messages (Rectangles) and Continuous Signals (Triangles
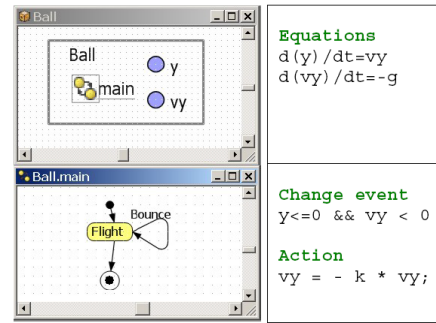
Figure 7: AnyLogic Model for the *Bouncing Ball*

AnyLogic influenced further developments for hybrid and structural dynamic systems, and led to a discussion in the Modelica community with respect to a proper implementation of state charts in Modelica. The principle question is, whether state charts are to be seen as comfortable way to describe complex WHEN – and IF – constructs, being part of the model, or whether state charts control different models from a higher level. At present (2008) a free Modelica state chart library 'emulates' state charts by Boolean variables and IF – THEN – ELSE constructs. A further problem is the fact, that the state chart notation is not really standardised; AnyLogic makes use of the Harel state chart type.

An AnyLogic implementation for the *Constrained Pendulum* may follow the implementation for the bouncing ball (Figure 8). An primary active object (Constrained Pendulum)'holds' the equations for the pendulum, together with a state chart (main) switching between short and long pendulum. The state chart nodes are empty; the arcs define the events (Figure 8). Internally, AnyLogic restarts at each hit the same pendulum model (trivial hybrid decomposition).
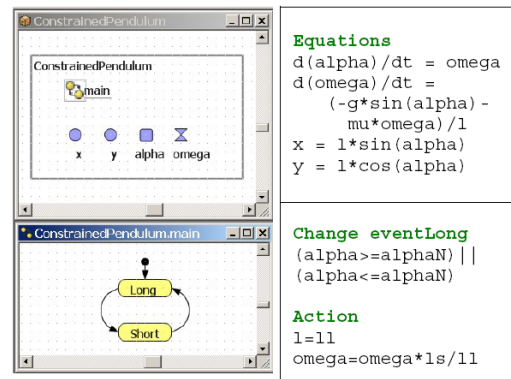
Figure 8: AnyLogic model for *Constrained Pendulum*, Simple Implementation

HYBRID AND STRUCTURAL-DYNAMIC SYSTEMS

Continuous simulation and discrete simulation have different roots, but they are using the same method, the analysis in the time domain. During the last decades a broad variety of model frames (model descriptions) has been developed.

In continuous and hybrid simulation, the explicit or implicit state space description is used as common denominator. This state space may be described textually, or by signal-oriented graphic blocks (e.g. SIMULINK), or by physically based block descriptions (Modelica, VHDL-AMS).

In discrete simulation, we meet very different techniques for the model frame. Application-oriented flow diagrams, network diagrams, state diagrams, etc. allow describing complex behaviour of event-driven dynamics. Usually these descriptions are mapped to an event-based description. On the other side, the simulator kernel is similar for discrete and continuous simulators. The model description is mapped to an event list with adequate update functions of the states within state update events. In discrete simulation, the states are usually the status variables of servers and queues in the model, and state update is simple increase or decrease by increments; complex logic conditions may accompany the scheduling of events. In continuous simulation the state space is based of various laws used in the application area, and usually defined by differential-algebraic equations. DAE solvers generate a grid for the approximation of the solutions. This grid drives an event list with state update events using complex formula depending on the chosen DAE solver and on the defined DAE. Additional time events and state events are inserted into the global event list.

Hybrid systems often come together with a change of the dimension of the state space, then called *structural-dynamic systems*. The dynamic change of the state space is caused by a state event of type SE-D. In contrary to state events SE-P and SE-S, states and derivatives may change continuously and differentiable in case of structural change. In principle, *structural-dynamic systems* can be seen from two extreme viewpoints. The one says, in a maximal state space, state events switch on and off algebraic conditions, which freeze certain states for certain periods. The other one says that a global discrete state space controls local models with fixed state spaces, whereby the local models may be also discrete or static.

These viewpoints derive two different approaches for structural dynamic systems modelling, the

- *maximal state space*, and the
- *hybrid decomposition*.

Maximal State Space for Structural-Dynamic Systems – Internal Events

Most implementations of physically based model descriptions support a big monolithic model description, derived from laws, ODEs, DAEs, state event functions and *internal events*. The state space is maximal and static, index reduction in combination with constraints keep a consistent state space. For instance, Dymola, OpenModelica, and VHDL-AMS follow this approach.

This approach can be classified with respect to event implementation. The approach handles all events of any kind (SE-P, SE-S, and SE-D) within the ODE solver frame, also events which change the state space dimension (change of degree of freedoms) – consequently called *internal events*.

Using the classical state chart notation, *internal state events* **I-SE** caused by the model schedule the model itself, with usually different re-initialisations (depending on the event type I-SE-P, I-SES, I-SE-D; Figure 9). For instance, VHDL-AMS and Dymola follow this approach, handling also DAE models with index higher than 1;

Figure 9: State Chart Control for *Internal Events* of one Model

discrete model parts are only supported at event level. ACSL and MATLAB / Simulink generate also a maximal state space.

Hybrid Decomposition for Structural-Dynamic Systems – External Events

The hybrid decomposition approach makes use of *external events* (E-SE), which control the sequence and the serial coupling of one model or of more models. A convenient tool for switching between models is a state chart, driven by the *external events* – which itself are generated by the models. Following e.g. the UML-RT notation, control for continuous models and for discrete actions can by modelled by state charts. Figure 10 shows the hybrid coupling of two models, which may be extended to an arbitrary number of models, with possible events E-SE-P, E-SE-S, and ESE-D. As special case, this technique may be also used for serial conditional 'execution' of one model – Figure 11 (only for SE-P and SE-S).

This approach additionally allows not only dynamically changing state spaces, but also different model types, like ODEs, linear ODEs (to be analysed by linear theory), PDEs, etc. to be processed in serial or also in parallel, so that also co-simulation can be formulated based on external events.
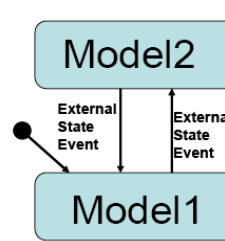
Figure 11: State Chart Control for *External Events* for two Models
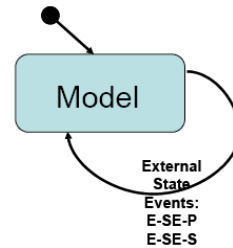
Figure 12: State Chart Control for *External Events* for one Model

The approach allows handling all events also outside the ODE solver frame. After an event, a very new model can be started. This procedure may make sense especially in case of events of type SE-D and SE-S. As consequence, consecutive models of different state spaces may be used.

Figure 12 shows a structure for a simulator supporting structural dynamic modelling and simulation. The figure summarises the outlined ideas by extending the CSSL structure by control model, external events and multiple models. The main extension is that the translator generates not only one DAE model; he generates several DAE models from the (sub)model descriptions, and external events from the connection model, controlling the model execution sequence in the highest level of the dynamic event list.
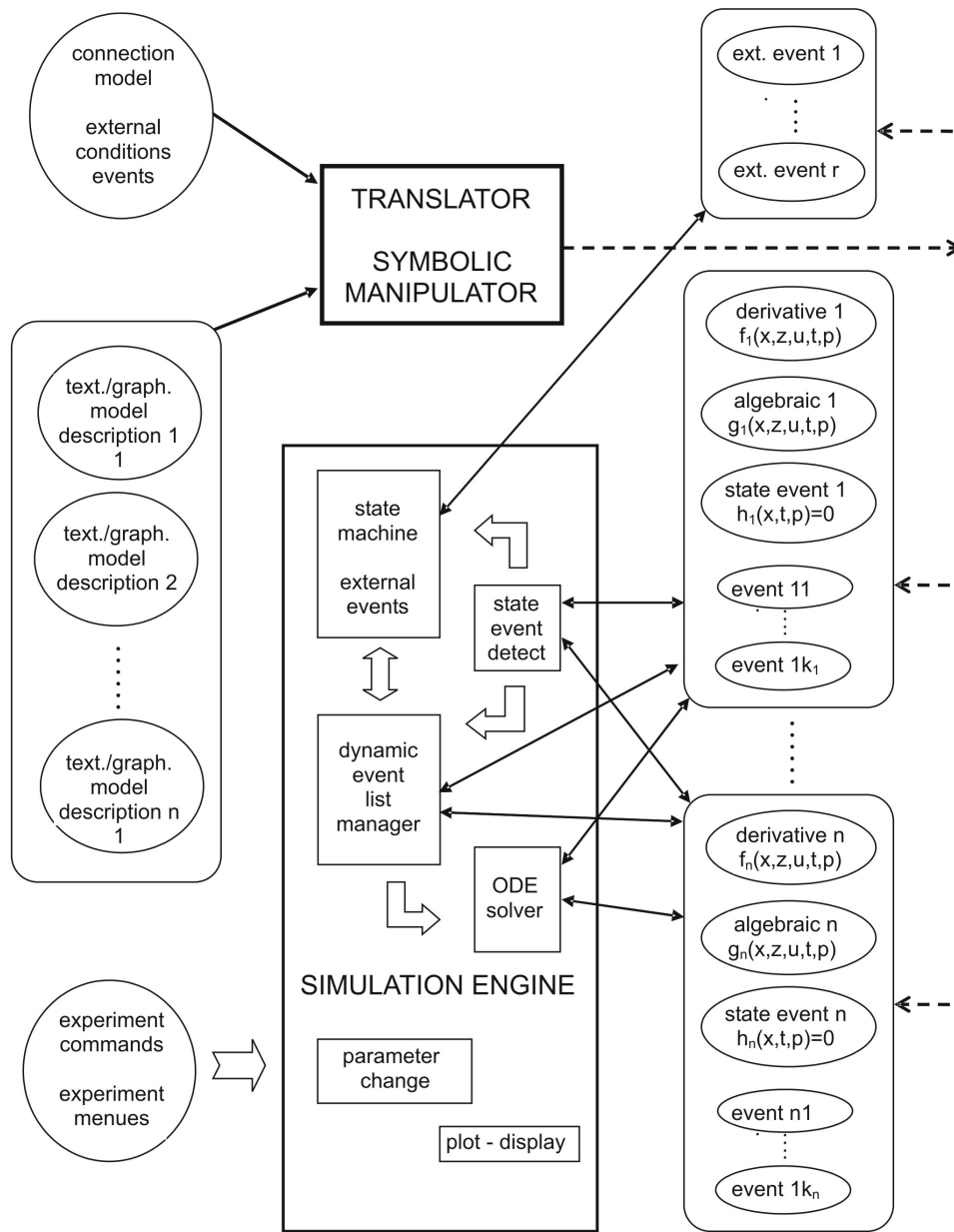
There, all (sub)models may be precompiled, or the new recent state space may be determined and translated to a DAE system in case of the external event (interpretative technique).

Clearly, not only ODE solver can make use of the model descriptions (derivatives), but also eigenvalue analysis and steady state calculation may be used and other analysis algorithms. Furthermore, complex experiments can be controlled by external events scheduling the same model in a loop.

Mixed Approach with Internal and External Events

A simulator structure as proposed in Figure 12 is a very general one, because it allows as well external as ell as internal events, so that hybrid coupling with variable state models of any kind with internal and external events is possible (Figure 13).

Both approaches have advantages and disadvantages. The classical Dymola approach generates a fast simulation, because of the monolithic program. However, the state space is static. Furthermore, Modelica centres on physical modelling. A hybrid approach handles separate model parts and must control the external events.

Consequently, two levels of programs have to be generated: dynamic models, and a control program – today's implementations are interpretative and not compiling, so that simulation times increase - but the overall state space is indeed dynamic.

A challenge for the future lies in the combination of both approaches. The main ideas are:

- Moderate hybrid decomposition
- External and internal events
- Efficient implementation of models and control

Figure 12: Structure for a Simulation System with *External State Events* E-SE and Classical Internal State Events I-SE for Controlling Different Models
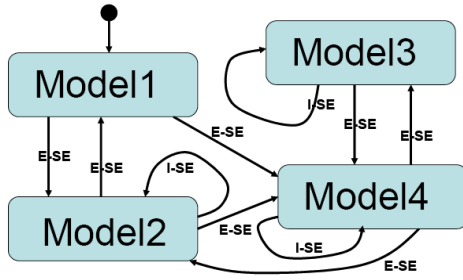
Figure13: State Chart Control for Different Models
with *Internal* and *External Events*

For instance, for parameter state events (SE-P) an implementation with an internal event may be sufficient (I-SE-P), for an event of SE-S type implementation with an external event may be advantageous because of easier state re-initialisation (E-SE-S), and for a structural model change (SE-D) an implementation with an external event may be preferred (E-SE-D), because of much easier handling of the dynamic state change – and less necessity for index reduction.

An efficient control of the sequence of models can be made by state charts, but also by a well-defined definitions and distinction of IF - and WHEN - constructs, like discussed in extensions of Scilab/Scicos for Modelica models.

STRUCTURAL FEATURES OF SIMULATORS

While the *extended features* discussed before address the CSSL-standard, *structural features* characterise features for physical modelling and for structural dynamic systems. This section investigates the availability or of structural features in some simulators, and summarises the results in table xx. Furthermore, is should be discussed, which software structure these simulators use (compared to Figure 3 and Figure 12). The extended features may be classified as follows:

- Support of a-causal physical modelling (sometimes called port-based modelling) at textual (PM-T) or graphical level (PM-G),
- Modelica standard (MOD) for a-causal physical modelling ,
- Decomposition of structural dynamic systems with dynamic features (SD) – features for external events, and
- Support of state chart modelling or a of a similar construct, by means of textual (SC-T) or graphical (SC-G) constructs.

In principle, each combination of the above features is possible. By means of the maximal state space approach, each classic simulator can handle structural dynamic systems, but a-causal modelling may be supported or not, and state chart modelling may be available or not. Simulators with a-causal modelling may

support hybrid decomposition or not, and state chart modelling may be available or not. Simulators with features for state chart modelling may support hybrid decomposition or not, and a-causal modelling may be offered or not. In general, interpreter-oriented simulators offer more structural flexibility, but modern software structures would allow also flexibility with precompiled models or with models compiled 'on the fly'.

In addition, of interest are also structural features as

- simulation-driven visualisation (with visualisation objects defined with the model objects; VIS),
- frequency domain analysis and linearization for steady state analysis (FA), and
- extended environment for complex experiments and data pre- and postprocessing (ENV).

In the following sections, simulators and simulation systems are investigated in order to check the availability of these structural features. For some of the simulators, implementation templates with the *Constrained Pendulum* are discussed.

MATLAB / Simulink / Stateflow

The mainly interpretative systems MATLAB / Simulink offer different approaches. First, MATLAB itself allows any kind of static and dynamic decomposition (SD 'yes'), but MATLAB is not a simulator, because the model equations have to be provided in a sorted manner, to be called from an ODE solver (MS 'no').
Second, MATLAB allows hybrid decomposition at MATLAB level with Simulink models. There, from MATLAB different Simulink models are called conditionally, and in Simulink, a state event is determined by the hit-crossing block (terminating the simulation). For control, in MATLAB only IF – THEN constructs are available. Table 6 – MATLAB control model, and Figure 15– graphical Simulink model, show a hybrid decomposition of this type for the *Constrained Pendulum*.
MATLAB is a very powerful environment with various modules. Simulink is MATLAB's simulation module for block-oriented dynamic models (directed signal graphs), which can be combined with Stateflow, MATLAB's module for event-driven state changes described by state charts (SC-T and SC-G 'yes').

Table 6: MATLAB Control Model for *Constrained Pendulum* with *External Events*
*S*witching between Long and Short Pendulum

```
if ((phi_p-phi0)*phi_p<0 |
        (phi0==phi_p & phi_p*v>0))
   dphi0=v/ls;
   sim('pendulum_short',[t(length(t)),10]);
   v=dphi(length(dphi))*ls;
else
   dphi0=v/l;
   sim('pendulum_long',[t(length(t)),10]);
   v=dphi(length(dphi))*l;
end
```
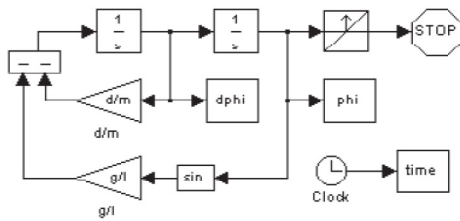
Figure 14: Simulink Model for *Constrained Pendulum* with *External Event* detected by Hit-Crossing Block

At Simulink level, Stateflow, Simulink's state chart modelling tool, may control different submodels. These submodels may be dynamic models based on ODEs (DAEs), or static models describing discrete actions (events). Consequently, Stateflow can be used for implementation of the *Constrained Pendulum*, where the state charts control length and change of velocities in case of hit by triggering the static changes (Figure 15). This implementation makes use of notations from analog circuits: the integrator, the 1/s – block, has not only continuous signal inputs, but also an reset control input and a static IC input, which toggle the velocity at hit. A solely Simulink implementation would make use of a triggered submodels describing the events by AND – and OR – blocks, or by a MATLAB function.

Alternatively, for *Constrained Pendulum* Stateflow could control two different submodels representing long and short pendulum enabled and disabled by the state chart control. Internally Simulink generates a state space with 'double' dimension, because Simulink can only work with a maximal state space and does not allow hybrid decomposition (SD 'no'). As advantage, this implementation would not need the old-fashioned integrator control.
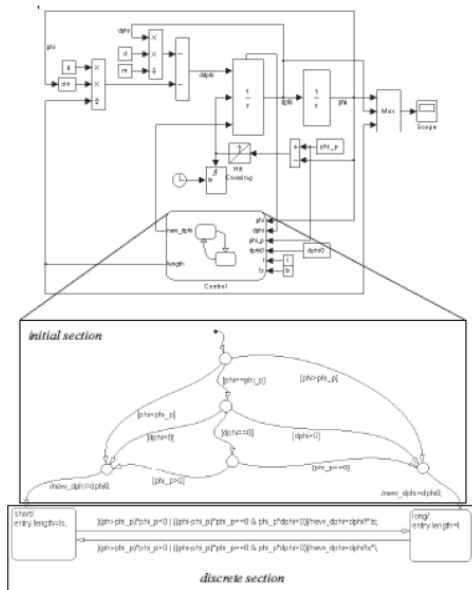
Figure15: Simulink Model for *Constrained Pendulum* with *External Event* detected by Hit-Crossing Block and controlled by Stateflow

Neither MATLAB nor Simulink support a-causal modelling. New MATLAB modules for physical modelling (e.g. *Hydraulic Blockset*) are precompiled to a classical state space (PM-T and PM-G 'no'), and furthermore Modelica modelling is not supported (MOD 'no') – Mathworks developers are working hard on some kind of real physical modelling and on Modelica modelling. For DAEs, MATLAB and Simulink offer modified LSODE solvers (implicit solvers) for the nested DAE solving approach. In MATLAB any kind of simulation – driven visualisation can be programmed and used in MATLAB or Simulink or in both, but not based on the model definition blocks (VIS '(yes)'). From the beginning on, MATLAB and Simulink offered frequency analysis (FA 'yes'), and clearly, MATLAB is a very powerful environment for Simulink, Stateflow, for all other Toolboxes, and for MATLAB itself (ENV 'yes').

ACSL

ACSL – Advanced Continuous Simulation Language – has been developed since more than 25 years. ACSL was strongly influenced by the CSSL standard. ACSL' software structure is a direct mapping of the structure in Figure 3. Implementations of the *Constrained Pendulum* have been shown in the previous sections, as example for modelling due to CSSL standard.

ACSL' development as simulator seems to have ended, as the new developers (Aegis Technologies) concentrate on application-oriented simulation solutions, with models are tailor-made for the specific application. Last extensions were a change to C as basic language (instead of FORTRAN), and DAE features using the nested approach with classical solvers, or direct implicit DAE solving with DASSL Code (DAE 'yes', IR 'no'). From the beginning on, steady state calculation, linearization and frequency analysis was a standard feature of ACSL's simulator kernel (FA 'yes').

Since 2000, the environment has been enriched by modules for modelling and environment modules. The first module was a graphical modeller. Figure 16 shows a graphical friction model ('block on a rough surface') in ACSL's graphic modeller, which seems to make use of physical modelling, but in behind classical state spaces as with Simulink's blocksets for physical modelling is used – PM-T and PM-G 'no'). Furthermore, a simulation-driven visualisation system (third party) is offered (but hard to use) – VIS '(yes)'.
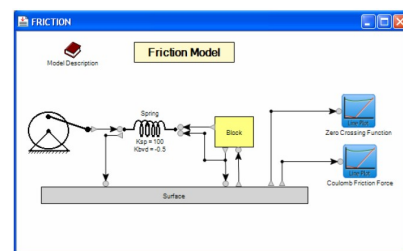
Figure 16: ACSL Graphic Model for Friction System

A very interesting module is an extended environment called ACSLMath. ACSLMath was intended to have same features as MATLAB; available is only a subset, but powerful enough for an extended environment (ENV 'yes'), which can be used for hybrid decomposition of a structural dynamic model in almost the same way than MATLAB does (SD 'yes'). Unfortunately the development of ACSLMath has been stopped. In general, there is no intention to make a-causal physical modelling available, also Modelica is not found in the developers' plans (PM-T, PM-G, and MOD 'no').

Dymola

Dymola has partly been discussed in a section before, together with an implementation for the *Constrained Pendulum* example (Dymola standard implementation, Figure 5, Table 5). Dymola, introduced by F. E. Cellier as a-causal modelling language, and developed to a simulator by H. Elmquist, can be called the mother of Modelica.

Dymola is based on a-causal physical modelling and initiated Modelica; consequently, it fully supports Modelica these structural features (PM-T, PM-G, and MOD 'yes'). Together with the model objects, also graphical objects may be defined, so that simulation based pseudo-3D visualisation is available (VIS 'yes'). A key feature of Dymola is the very sophisticated index reduction by the modified Pantelides algorithm, so Dymola handles any DAE system, also with higher index, with bravura (DAE and IR 'yes'). For DAE solving, modified DASSL algorithms are used. In software structure, Dymola is similar to ACSL, using an extended CSSL structure as given in Figure 3 – with the modification that all discrete actions are put into one event module, where CASE - constructs distinguish between the different events (this structure is based on the first simulator engine Dymola used, the DS-Block System of DLR Oberpfaffenhofen).

Dymola comes with a graphical modelling and basic simulation environment, and provides a simple script language as extended environment; new releases offer also optimisation, as built-in function of the simulator. Furthermore, based on Modelica's matrix functions some task of an environment can be performed – so ENV ('yes') – available, but complex/uncomfortable.

Dymola offers also a Modelica – compatible state chart library, which allows to model complex conditions (internally translated into IF – THEN – ELSE or WHEN constructs - SC-T and SC-G '(yes)'). Figure 17 shows an implementation of the *Constrained Pendulum* using this library.
Up to now (2008) the Modelica definition says nothing about structural dynamic systems, and Dymola builds up a maximal state space. And up to now, Modelica does not directly define state charts, and in Dymola a state chart library in basic Modelica notation is available, but working only with internal events within the maximal state space (SD 'no').
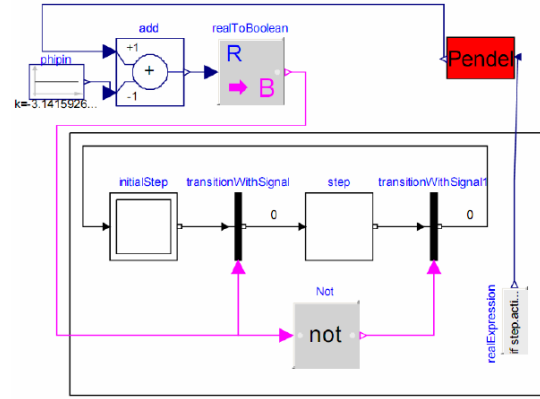
Figure17: Graphical Dymola Model for *Constrained Pendulum* with *Internal Events Managed* by Elements of Dymola's State Chart Library

For Modelica extension, a working group on hybrid systems has been implemented, in order to discuss and standardise hybrid constructs like state charts, and hybrid decompositions (independent submodels).

Another interesting and remarkable development started in 2006 – Modelica's basic static calculation features become notable. These basic features include any kind of vector and matrix operations, and they can be extended by Modelica's generic extension mechanism. In principle, 'static' Modelica defines a MATLAB-like language. Simulators being capable of understanding Modelica, must consequently also support these static calculations (without any DAE around) – so that each Modelica simulator becomes a 'Mini-MATLAB'. In Dymola, such calculations may be performed in a textual Dymola consisting only of an algorithmic section, without any time advance from the simulator kernel.

MathModelica

MathModelica, developed by MathCoreAB, was the second simulation system, which understood Modelica modelling. MathModelica is an integrated interactive development, from modelling via simulation to analysis and code integration. As furthermore the MathModelica translator is very similar to Dymola's model translator, clearly all related features are available, including index reduction and use of implicit solvers like DASSL (all DAE, IR, PM-T, PM-G and MOD 'yes').

Figure 18 shows a drive train model set up with MathModelica's Mechanics Package (Modelica modelling) – due to the Modelica standard this model looks almost exactly like the model in Dymola, SimulationX, etc.
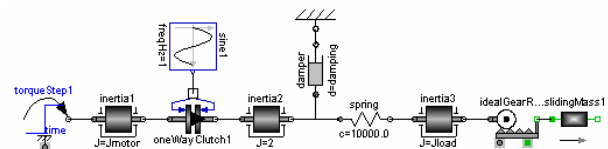
Figure 18: MathModelica Model for a Drive Train

MathModelica follows a software model different to CSSL standard. The user interface consists of a graphical model editor and notebooks. There, a *simulation center* controls and documents experiments in the time domain. Documentation, mathematical type setting, and symbolic formula manipulation are provided via Mathematica, as well as Mathematica acts as extended environment for MathModelica (ENV 'yes') – performing any kind of analysis and visualisation (FA and VIS 'yes'). By means of the Mathematica environment, also a hybrid decomposition of structural dynamic systems is possible, with the same technique like in MATLAB (SD – 'yes').

Mosilab

Since 2004, Fraunhofer Gesellschaft Dresden develops a generic simulator *Mosilab*, which also initiates an extension to Modelica: multiple models controlled by state automata, coupled in serial and in parallel. Furthermore, Mosilab puts emphasis on co-simulation and simulator coupling, whereby for interfacing the same constructs are used than for hybrid decomposition. Mosilab is a generic Modelica simulator, so all basic features are met (ED, SEH, DAE, PM-T, and PM-G 'yes', and MOD '(yes)' – because of subset implementation at present, 2008). For DAE solving, variants of IDA-DASSL solver are used.

Mosilab implements extended state chart modelling, which may be translated directly due to Modelica standard into equivalent IF – THEN constructs, or which can control different models and model executions (SC-T, SC-G, and SD 'yes'). At state chart level, state events of type SE-D control the switching between different models and service the events (E-SE-D). State events affecting a state variable (SE-S type) can be modelled at this external level (E-SE-S type), or also as classic internal event (I-SE-S). Mosilab translates each model separately, and generates a main simulation program out of state charts, controlling the call of the precompiled models and passing data between the models, so that the software model of Mosilab follows the structure in Figure 12. The textual and graphical constructs for the state charts are modifications of state chart modelling in AnyLogic.

Mosilab is in developing, so it supports only a subset of Modelica, and index reduction has not been implemented yet, so that MOD gets a '(yes)' in parenthesis, and IR gets a '(no)' – indicating that the feature is not available at present (2008), but is scheduled for the future. Index reduction at present not available in Mosilab, but planned (IR '(no)') - has become topic of discussion: case studies show, that hybrid decomposition of structural dynamic systems results mainly in DAE systems of index $n = 1$, so that index reduction may be bypassed (except models with contact problems).

Mosilab allows very different approaches for modelling and simulation tasks, to be discussed with the *Constrained Pendulum* example. Three different modelling approaches reflect the distinction between internal and external events as discussed before.

Mosilab Standard Modelica Model. In a standard Modelica approach, the *Constrained Pendulum* is defined in the MOSILAB equation layer as implicit law; the state event, which appears every time when the rope of the pendulum hits or 'leaves' the pin, is modelled in an `algorithm section` with if (or when) – conditions (Table 7).

Table 7: Mosilab Model for *Constrained Pendulum* – Standard Modelica Approach with *Internal Events* (I-SE-P)

```
equation /*pendulum*/
 v = l1*der(phi); vdot = der(v);
 mass*vdot/l1 + mass*g*sin(phi)+damping*v = 0;
algorithm
 if (phi<=phipin) then length:=ls; end if;
 if (phi>phipin) then length:=l1; end if;
end
```

Mosilab I-SE-P Model with State Charts. MOSILAB's state chart approach models discrete elements by state charts, which may be used instead of IF - or WHEN - clauses, with much higher flexibility and readability in case of complex conditions. There, Boolean variables define the status of the system and are managed by the state chart. Table 8 shows a Mosilab implementation of the *Constrained Pendulum*: the state charts initialise the system (`initial` state) and manage switching between long and short pendulum, by changing the length appropriately.

Table 8: Mosilab Model for *Constrained Pendulum* – State Chart Model with *Internal Events* (I-SE-P)

```
event Boolean lengthen(start=false),
 shorten(start = false);
equation
lengthen=(phi>phipin); shorten=(phi<=phipin);
equation /*pendulum*/
 v = l1*der(phi); vdot = der(v);
 mass*vdot/l1 + mass*g*sin(phi)+damping*v= 0;
statechart
 state LengthSwitch extends State;
 State Short,Long,Initial(isInitial=true);
transition Initial -> Long end transition;
transition Long -> Short event shorten action
 length := ls;
end transition;
transition Short -> Long event lengthen action
 length := l1;
end transition; end LengthSwitch;
```

From the modelling point of view, this description is equivalent to the aforementioned description with IF - clauses. The Mosilab translator clearly generates there an implementation with different internal equations.

Mosilab E-SE-P Model. Mosilab's state chart construct is not only a good alternative to IF - or WHEN - clauses within one model, it offers also the possibility to switch between structural different models. This very powerful feature allows any kind of hybrid composition of models with different state spaces and of different type (from ODEs to PDEs, etc.). Table 9 shows a Mosilab implementation of the *Constrained Pendulum* making use of two different pendulum models, controlled externally by a state chart. Clearly, in case of this simple model, different models would not be necessary.

Here, the system is decomposed into two different models, `Short` pendulum model, and `Long` pendulum model, controlled by a state chart. The model description (Table 9) defines now first the two pendulum models, and then the event as before. The state chart creates first instances of both pendulum models during the initial state (`new`). The transitions organise the switching between the pendulums (`remove`, `add`). The `connect` statements are used for mapping local to global state.

Table 9: Mosilab Model for *Constrained Pendulum* – State Chart Switching between Different Pendulums Models by *External Events* (E-SE-P)

```
model Long
equation
 mass*vdot/l1 + mass*g*sin(phi)+damping*v = 0;
end Long;
model Short
equation
 mass*vdot/ls + mass*g*sin(phi)+damping*v = 0;
end Short;
event discrete Boolean lengthen(start=true),
 shorten(start = false);
equation
 lengthen =
 (phi>phipin);shorten=(phi<=phipin);
statechart
state ChangePendulum extends State;
 State Short,Long,startState(isInitial=true);
transition startState -> Long action
 L:=new Long(); K:=new Short(); add(L);
end transition;
transition Long->Short event shorten action
 disconnect ….; remove(L); add(K); connect …
end transition;
transition Short -> Long event lengthen
 action
 disconnect …;
 remove(K); add(L);
 connect ……
end transition; end ChangePendulum;
```

Mosilab offers also strong support for simulator coupling (e.g. MATLAB) and time-synchronised coupling of external programs. This feature may be used for any kind of visualisation not based the model definition (VIS '(yes)').

External events driven by external states charts open possibilities, which were not planned at begin of Mosilab development, but which became obvious during development. It turned out, that complex experiments can be defined and performed by means of external state charts - as well as a simple parameter loop, which makes use of the same model in each state change (change of parameter value). Furthermore, at level of the 'main' model, any kind of static calculations due to Modelica standard should be possible. There, Mosilab mixes model frame and experimental frame and sets up a common extended environment (ENV 'yes'), where also frequency analysis can be implemented (FA '(no)').

Open Modelica

The goal of the *Open Modelica* project is to create a complete Modelica modelling, compilation and simulation environment based on free software distributed in binary and source code form.
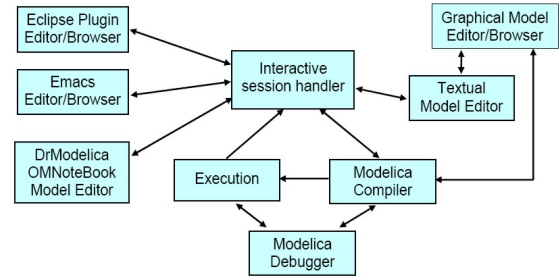
Figure 19: Software Modules of Open Modelica.

The whole OpenModelica environment consists of open software (Figure 19): *OMC* – the Open Modelica Compiler translates Modelica models (with index reduction); *OMShell* as interactive session handler is a minimal experiment frame; Modelica models may be set up by a simple text editor or by a graphical model editor (here, for teaching purposes the model editor of MathModelica is allowed to be used!); the purpose of *OMNotebook* is to provide an advanced Modelica environment and teaching tool; the *DrModelica* notebook provides all the examples from P. Fritzson's book on Modelica; the other modules support environment interfacing and Open Modelica development.

Open Modelica is a generic Modelica simulator, so all basic features are met (ED, SEH, DAE, PM-T, PM-G, IR and MOD 'yes'; for DAE solving, variants of DASSL solver are used). P. Fritzson, the initiator of Open Modelica puts emphasis on discrete events and hybrid modelling, so documentation comes with clear advice for use of IF – and WHEN – clauses in Modelica, and with state chart modules in DrModelica – so SC-T gets 'yes'. Figure 20 shows the equivalence of a state chart and the correct definition as Modelica submodel. For graphical state chart modelling the experimental Modelica state chart library can be used – so SC-G '(yes)'.

The notebook features allow interfaces and extensions of any kind, e.g. for data visualisation and frequency analysis – FA and VIS '(yes)'; they allow also for controlled executive of different models, so that hybrid decomposition of structural dynamic systems is possible – SD '(yes)'.


```
partial model SimpleBacklash
Boolean backward, slack, forward;
parameter ……
equation
phi_dev = phi_rel - phi_rel0;
 backward = phi_rel < -b/2;
 forward = phi_rel > b/2;
 slack = not (backward or forward);
 tau = if forward then
          c*(phi_dev – b/2)
      else (if backward then
              c*(phi_dev + b/2)
          else 0);
end SimpleBacklash
```
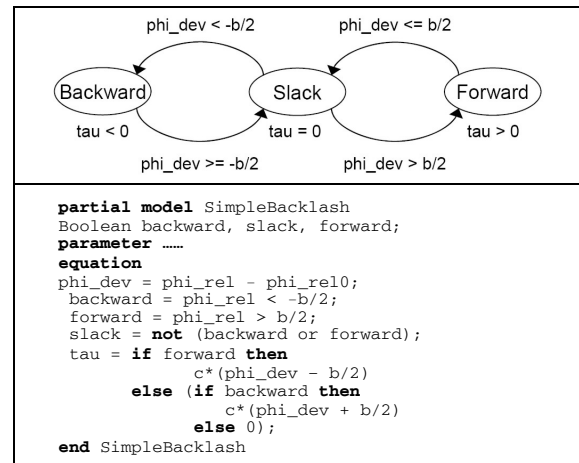
Figure 20: OpenModelica State Chart Modelling

SimulationX

SimulationX is a new Modelica simulator developed by ITI simulation, Dresden. This almost generic Modelica simulator is based on ITI's simulation system ITI-SIM, where the generic IT-SIM modelling frame has been replaced by Modelica modelling. From the very beginning on, ITI-SIM concentrated on physical modelling, with a theoretical background from power graphs and bond graphs. Figure 21 shows graphical physical modelling in ITI-SIM – very similar to Modelica graphical modelling.

The simulation engine from ITI-SIM drives also SimulationX, using a sophisticated implicit integration scheme, with state event handling. Consequently, all features f related to physical modelling are available: (ED, SEH, DAE, PM-T, PM-G, and MOD 'yes'; index reduction is not really implemented – IR '(no)'.

State chart constructs are not directly supported (SC-T 'no'), but due to Modelica compatibility the Modelica state chart library can be used (SC-G – '(yes)'. SimulationX (and ITI-SIM) put emphasis on physical application – oriented modelling and simulation, so frequency analysis is directly supported in the simulation environment (FA – 'yes'), which offers via additional modules (e.g. interfaces to multibody systems) connectivity to external systems (ENV – '(yes)'). The simulation engine drives also pseudo-3D visualisation (VIS – 'yes').
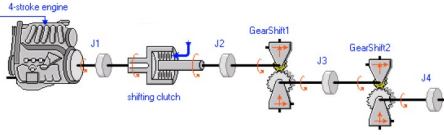
Figure 21: Physical Modelling in ITI-SIM / SimulationX

AnyLogic

AnyLogic – already discussed in a previous section) is based on hybrid automata (SC-T and SC-G - 'yes'). Consequently, hybrid decomposition and control by external events is possible (ED, SD 'yes'). AnyLogic can deal partly with implicit systems (only nested approach, DAE '(yes)'), but does not support a-causal modelling (PM-T, PM-G - 'no') and does not support Modelica (MOD - 'no'). Furthermore, new versions of AnyLogic concentrate more on discrete modelling and modelling with System Dynamics, whereby state event detection has been sorted out (SEH '(no)'. On the other hand, AnyLogic offers many other modelling paradigms, as System Dynamics, Agent-based Simulation, DEVS modelling and simulation. AnyLogic is Java-based and provides simulation-driven visualisation and animation of model objects (VIS 'yes') and can also generate Java web applets.

In AnyLogic, various implementations for the *Constrained Pendulum* are possible. A classical implementation is given in Figure 8, following classical textual ODE modelling, whereby instead of IF – THEN clauses a state chart is used for switching (I-SE-P, I-SE-S).
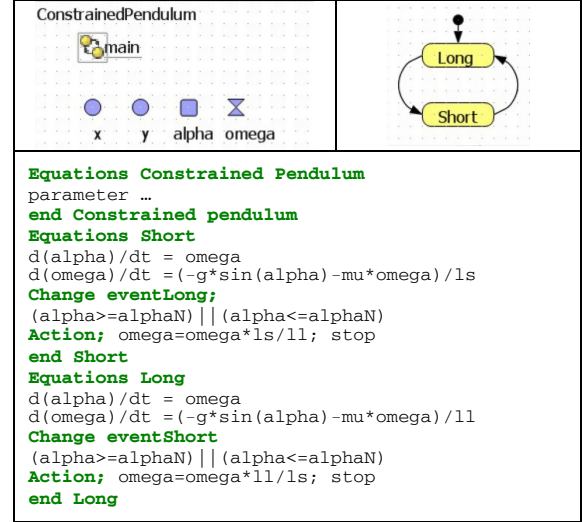
Figure 21: AnyLogic Model for *Constrained Pendulum*, Hybrid Model Decomposition with two Pendulum Models and *External Events*

AnyLogic E-SE-P Model with State Charts. A hybrid decomposed model may make use of two different models, each defined in substate / submodel Short and Long. – both part of a state chart switching between these submodels. The events defined at the arcs stop the actual model, set new initial conditions and start the alternative model (Figure 21).

AnyLogic E-SE-P Model with Parallel Models. AnyLogic works interpretatively, after each external event state equations are tracked and sorted anew for the new state space. This makes it possible, to decompose model not only in serial, but also in parallel. In *Constrained Pendulum* example, the ODE for the angle, which is not effected by the events, may be put in the main model, together with transformation to Cartesian coordinates (Figure 22), which seems to run in parallel with different velocity equations.
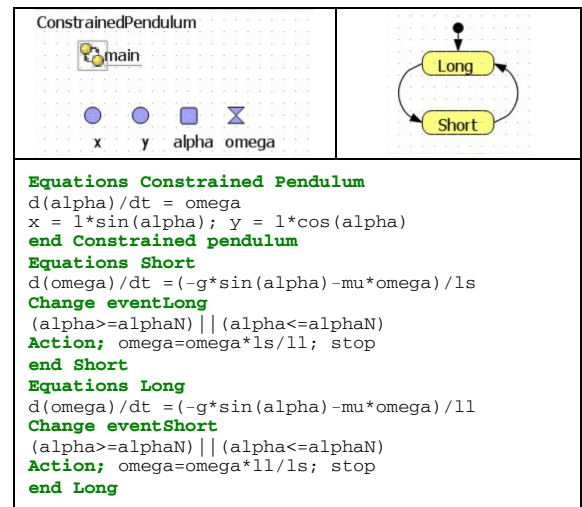
Figure 22 AnyLogic Model for *Constrained Pendulum*, Hybrid Model Decomposition with Two Models for Angular Velocity and Overall Model for Angle

From software engineering view, AnyLogic is a programming environment for Java, with special features for ODE simulation. At each level Java code can be entered, and Java modules linked and called. The main module may be arbitrarily extended by Java code, stating not only the (predefined) simulation engine, but also frequency analysis packages, etc., with programming effort – so ENV '(yes)'

Model Vision Studium MVS

Model Vision Studium (MVS) – is an integrated graphical environment for modelling and simulation of complex dynamical systems. Development of MVS started in the 1990ies at Technical University of St. Petersburg; for end of 2008, an English version is announced.
Basis of MVS are hybrid state charts (SC-T, SC-G - 'yes'), allowing any parallel, serial, and conditional combination of continuous models, described by DAEs, and controlled and interrupted by state events (ED, SHE - 'yes'). State models itself are objects to be instantiated in various kinds, so that structural dynamic systems of any kind can be modelled (SD - 'yes'). Textual physical and DAE modelling is supported by an editor capable of editing mathematical formula (DAE and PM-T 'yes', PM-G no), but no Modelica compatibility (MOD – 'no').
For MVS, a subset of *UML Real Time* was chosen and extended to state chart activities (Java – based). Other modules (simulation kernel, environment) are linked modules (e.g. C-modules), e.g. Java-base simulation driven visualisation (VIS - 'yes'). In principle, MVS and AnyLogic have been developed in parallel. The continuous elements in AnyLogic have been taken from MVS, because AnyLogic started as pure discrete simulator.

State charts are similar to AnyLogic, consisting of different implicit state space descriptions – and also defining complex experiments (calling different models; ENV – 'yes), but without frequency analysis (FA – 'no').

As example, two states pendulum and flight, and a state chart handling the external event of type E-SE-D (Figure 23) describe a breaking pendulum.
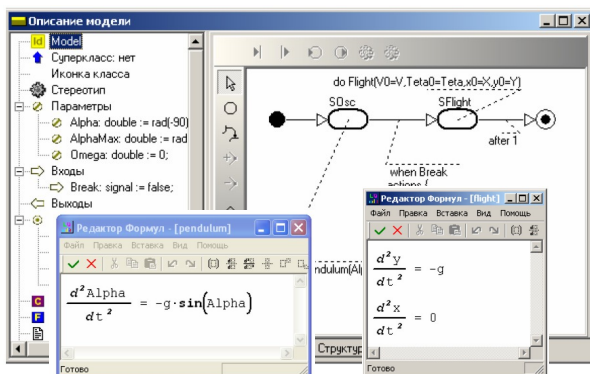
Figure 23: MVS Model for Breaking Pendulum
- Hybrid Model Decomposition into Pendulum
and Flight Model

SCILAB / SCICOS

Scilab is a scientific software package for numerical computations with a powerful open computing environment for engineering and scientific applications. Scilab is open source software. Scilab is now the responsibility of the Scilab Consortium, launched in May 2003. *Scicos* is a graphical dynamical system modeller and simulator toolbox included in Scilab.
Scilab / Scicos is an open source alternative to MAT-LAB / Simulink, developed in France. Consequently, Scilab as MATLAB – like tool has nearly the same features than MATLAB: no equation sorting– MS – 'no'!; DE, IR, PM-T, PM-G, MOD, SC-T, and SC-G – 'no'; SEH, DAE, and VIS – '(yes)', remarkably – SD, FA and ENV – 'yes'. Similarly, Scicos has extended features ED, SEH, and DAE – 'yes'.

The developers of Scicos started early with a kind of physical modelling – Figure 24 shows an electrical modelling palette of Scicos (PM-T, PM-G – yes). They are working on extensions in two directions:

- extending the model description by full Modelica models (textually and graphically) –
  - so MOD and IR '(yes)' (subset)
- refining the IF-THEN-ELSE – and WHEN – clause introducing different classes of associated events, resulting 'state chart clauses' - so SC-T – 'yes'

In Scicos, the Modelica state chart library allows graphical state chart modelling. Standalone Scicos has no features for frequency analysis, structural decomposition and extended environment (FA, SD, ENV – 'no'), but limited visualisatiasion (VIS – '(yes)'); Scicos controlled by Scilab has all these features (VIS, FA, SD, ENV – 'yes').
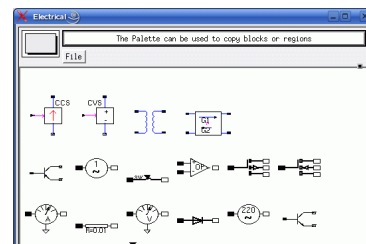
Figure 24: Scicos Physical Modelling Palette for
Electrical Applications

Maple

Maple – developed by Maplesoft, Canada, is working on a toolbox *MapleSim*, which will understand Modelica models (PM-T, PM-G, and MOD – 'yes'). Maple acts as environment and provides sophisticated DAE solvers, as well as symbolic algorithms for index reduction (DAE, IR, ENV, VIS, FA – 'yes').
In development are constructs for events and event handling (ED – 'yes'), SEH – '(no)'); textual state chart modelling has not been discussed yet, graphical state chat notation may com from the experimental Modelica state chart library (SC-T – 'no', SC-G – '(yes)').

Availability of Structural Features

Table 10 provides an availability comparison of the discussed features within the presented simulators. Clearly such comparison must be incomplete, and using simple 'yes' and 'no' might be too simple. Consequently, it should be a hint for further detailed feature comparison.

REFERENCES

As a really adequate reference list, with details on structures, features, and detailed developments and background would cover again 10 pages, alternatively the list is restricted to only few main sources. For information modelling approaches, it is referred to the journal SNE – Simulation News, where regularly benchmarks, also for Modelica modelling, are published (http: sne.argesim.org, ww.argesim.org). For simulator information, see webpages of distributors / developers.

F. Breitenecker F., and I. Troch. 2004. 'Simulation Software – Development and Trends'. In *Modelling and Simulation of Dynamic Systems / Control Systems, Robotics, and Automation*. H. Unbehauen, I. Troch, and F. Breitenecker (Eds.). Encyclopedia of Life Support Systems (EOLSS), UNESCO, Eolss Publishers, Oxford ,UK, www.eolss.net.

Cellier, F.E. (1991). *Continuous System Modeling*. Springer, New York.

Cellier, F.E., and E. Kofman. 2006. *Continuous System Simulation*. Springer, New York.

Fritzson, P. 2005. Principles of Object-Oriented Modeling and Simulation with Modelica. Wiley IEEE Press.

Fritzson, P., F.E.Cellier, C. Nytsch-Geusen, D. Broman, and M. Cebulla, Eds. 2007. *EOOLT'2007 - Proc. 1st Intl. Workshop on Equation-based Object-oriented Languages and Tools*. TU Berlin Forschungsberichte, Vol. 2007-11.

Nytsch-Geusen C., and P. Schwarz. 2005. 'MOSILAB: Development of a Modelica based generic simulation tool supporting model structural dynamics'. In *Proc. 4th Intern. Modelica Conference,* G. Schmitz (Ed.), Modelica Association - www.modelica.org, 527 – 535.

Strauss J. C. 1967. 'The SCi continuous system simulation language (CSSL)'. Simulation 9, SCS Publ. 281-303.

AUTHOR BIOGRAPHIES

Felix Breitenecker studied 'Applied Mathematics' and acts as professor for Mathematical Modelling and Simulation at Vienna University of Technology. He covers a broad research area, from mathematical modelling to simulator development, from DES via numerical mathematics to symbolic computation, from biomedical and mechanical simulation to process simulation. He is active in various simulation societies: president and past president of EUROSIM since 1992, board member and president of the German Simulation Society ASIM, member of INFORMS, SCS, etc. He has published about 250 scientific publications, and he is author of two books and editor of 22 books. Since 1995, he is Editor in Chief of the journal Simulation News Europe.

	MS - Model Sorting	ED -Event Description	SEH -State Event Handling	DAE - DAE Solver	IR - Index Reduction	PM-T - Physical Modelling -Text	PM-G - Physical Modelling -Graphics	VIS – 'Onlie' - Visualisation	MOD – Modelica Modelling	SC-T – State Chart – Modelling - Text	SC-G – State Chart Modelling - Graphics	SD – Structural Dynamic Systems	FA – Frequency Analysis	ENV – Extended Environment
MATLAB	no	no	(yes)	(yes)	no	no	no	(yes)	no	no	no	yes	yes	yes
Simulink	yes	(yes)	(yes)	(yes)	no	no	(no)	(yes)	no	no	no	no	yes	(yes)
MATLAB / Simulink	yes	yes	yes	(yes)	no	no	(no)	(yes)	no	no	no	yes	yes	yes
Simulink / Stateflow	yes	yes	yes	(yes)	no	no	(no)	(yes)	no	(yes)	yes	no	yes	(yes)
ACSL	yes	yes	yes	yes	no	no	(no)	(yes)	no	no	no	no	yes	yes
Dymola	yes	yes	yes	yes	yes	yes	yes	yes	yes	(yes)	(yes)	no	(no)	(yes)
MathModelica	yes	yes	yes	yes	yes	yes	yes	(yes)	yes	(no)	(yes)	no	(no)	(no)
MathModelica / Mathematica	yes	yes	yes	yes	yes	yes	yes	(yes)	yes	(no)	(yes)	yes	yes	yes
Mosilab	yes	yes	yes	yes	(no)	yes	yes	(no)	(yes)	yes	yes	yes	no	(yes)
Open Modelica	yes	yes	yes	yes	yes	yes	(no)	(no)	yes	(no)	(yes)	no	no	no
SimulationX	yes	yes	yes	yes	yes	yes	yes	yes	yes	(no)	(yes)	no	yes	(yes)
AnyLogic	yes	yes	(yes)	(yes)	no	no	no	yes	no	yes	yes	yes	no	no
Model Vision	yes	yes	yes	yes	yes	yes	no	yes	no	yes	yes	yes	yes	no
Scilab	no	no	(yes)	(yes)	no	no	no	(yes)	no	no	no	yes	yes	yes
Scicos	yes	(yes)	yes	yes	(yes)	yes	yes	(yes)	(yes)	yes	(yes)	no	no	no
Scilab/ Scicos	yes	yes	yes	(yes)	yes	yes	yes	(yes)	(yes)	yes	(yes)	yes	yes	yes
(MapleSim)	yes	(yes)	(yes)	yes	yes	yes	yes	yes	yes	no	no	(yes)	(yes)	yes

Table 10: Availability of Structural Features in Simulators - DAEs, State Events, Modelica Notation, Structural Decomposition, and Related Features

Simulation in Industry, Business and Services

Use of Fuzzy Reasoning in the Simulation of Risk Events in Business Processes

Paul Taylor, Jesús Jimenez Godino, and Basim Majeed
Intelligent Systems Research Centre,
British Telecommunications plc,
pp. 12, 1st Floor, Orion Building, Adastral Park, Martlesham Heath, IPSWICH, UK, IP5 3RE
E-mail: {paul.n.taylor, jesus.jimenezgodino, basim.majeed}@bt.com

KEYWORDS

Simulation, Risk, Business Process, Fuzzy Logic, KPI, KRI.

ABSTRACT

The current drive towards Service Oriented Architecture (SOA) and Business Process Execution Language (BPEL) in enterprises will increase dependency on efficient businesses processes. In the current competitive environment, process efficiency gains are seen as a crucial factor for business success. However it is not sufficient to design a process that works well under normal conditions. Risk analysis and mitigation is an important activity that should be tackled systematically during process design and improvement. The process designer's job has thus become particularly complex, requiring tools that combine traditional business process management with operational risk analysis.

In this paper we introduce a simulation environment that has been developed within British Telecommunications plc to simulate business process performance. The simulator incorporates a facility to simulate arbitrary risk effects on the performance of the process. Since risk analysis typically deals with qualitative values such as "high probability risk" or "low impact risk", measuring key risk indicators (KRIs) can be difficult. The simulator allows the process designer to formulate a fuzzy system of rules to define how risk is measured; these allow the user to produce KRIs that utilise the qualitative risk knowledge in addition to the ability to derive quantitative risk measures should they be needed.

INTRODUCTION

The increasingly process driven nature of businesses and the increasing interest in Service Oriented Architecture (SOA) and Business Process Orchestration technologies such as Business Process Execution Language (BPEL) in IT Departments in many companies mean it is imperative that each business process be as efficient as possible, both in terms of the systems that are invoked in the course of process execution and in terms of the human tasks that are required to ensure a favourable result.

In this paper we showcase some of the innovations that are currently being developed within British Telecommunications plc (BT) to drive efficiencies within business processes and also showcase some of the tools that are being developed to aid process designers in designing and re-engineering efficient business processes.

Our existing work on Real-Time Business Intelligence (RTBI) (Azvine et al. 2006) has emphasised the need for businesses to develop and use efficient business processes that are able to respond appropriately to internal and external factors that can affect the outcome or execution of the process.

To this end we have developed an extensible process simulation tool that allows Process Designers to simulate the execution of their business processes and gain insights into the outcome of those process in terms of the estimated value of specified Key Performance Indicators (KPI) and in terms of the throughput of the process through the estimation and visualisation of the queues for each activity.

As is discussed in the literature (Lambert 2006; Neiger 2006) good business processes should take into account the risks involved in carrying out the process and choices at design time should take into account the level of risk for each activity so the overall level of risk for the holistic process is minimised.

Our simulation tool builds upon plain discrete event simulation of business processes and includes facilities to utilise uncertainty and risk within the analysis that is performed including the ability to use of fuzzy logic to model risk.

The tool is targeted at process designers and process managers in order to provide some predictive analytic capability. This would allow stakeholders to recalculate Key Risk Indicators (KRI) quickly when considering adjustments to the process and additionally gain insights into the performance of the process both with and without the possibility of risk occurring during the process execution.

In this paper we next introduce some information about business process management (BPM) as a discipline and its links with risk analysis. The third section introduces the different features of the simulator. The fourth section provides a concrete example of how the tool is used giving examples of the insights that can be gained. The final section concludes the paper.

BUSINESS PROCESS MANAGEMENT & RISK

Business processes are defined by Davenport (1993) as a set of structured, measured activities across time and place, with a beginning, an end, and clearly identified inputs and outputs which are designed to produce a specified output for a particular customer or market. It is easy to see from this definition that most businesses can be seen as a collection of business processes. An efficient set of business processes is therefore essential to the efficient running of such a 'process driven' business.

Producing efficient business processes is not an easy task. One of the most important tools in process design and improvement is a simulation environment that can provide insight into the bottlenecks that could be encountered in the process, and also will allow the designer to see the effect of differing resource levels upon the defined process.

A number of articles in the literature mention the link between business processes and risk analysis (Lambert et al. 2006; Neiger et al. 2006; Jallow et al. 2007). The importance of analysing process risk led to this work which extends a basic process simulator with the ability to simulate the effect of risk events upon the process execution.

According to Frost et al. (2001) a risk is an uncertain future event which may have an effect upon the ability of an organisation to reach its objectives. Furthermore a risk consists of two attributes: first, a probability of occurrence and second, an effect.

Before the simulation approach discussed in this paper can begin it would be necessary for the risks involved in the process to be defined. The best approach would be to leverage a standard risk analysis process such as COSO (2004), for more information see Jallow et al. (2007).

Once risks to a process have been identified and assigned a probability and an effect, a corresponding risk can be defined within the simulator and will be used in the simulation of the process model when executed. These risk probabilities and risk effects are typically of a qualitative nature. This presents a challenge for the simulation of business processes and because risk analysis of a process is of limited value without knowledge of the business performance under the specified risk, it is important for the simulator to be able to express the performance of the process in values which can be easily understood by business users, that is to say quantitative values.

In the paper by Intaver (2005) a strong case is made for a combined quantitative and qualitative approach to risk management to aid project managers in accurate estimation of risks and effects. Zadeh (1994) argues that humans will typically prefer to reason about risks in a qualitative way. However since this reasoning is typically subjective it can be compromised by a number of other factors that may effect the quality of the results that are obtained. In addition the performance of business processes is typically measured in a quantitative rather than qualitative way and a quantitative answer that illustrates business performance is usually desirable.

Fuzzy logic allows for reasoning about quantitative measures in a qualitative way, and as such seems to be an ideal solution to the issues that we have described.

As such, we have implemented a simulation environment that allows the users of the system to define fuzzy logic rules to express the qualitative nature of risks to the process under test, and have the simulator return the performance of the business process in terms of KPIs that allow for easy comparison across simulation executions.

In addition to the advantages of fuzzy logic for reasoning about systems, it also allows the system to suppress small changes in those input values and therefore provide a more holistic view of the process and its simulation.

PROCESS SIMULATOR

The process simulator forms only one part of the RTBI framework that is being developed within BT Research & Venturing. Specifically the simulator is built on top of the open-source JBPM stack and uses their JPDL workflow language to define processes. Both of these components are produced by JBoss, which is a division of Red Hat Middleware, LLC. (http://www.jboss.com).

The defined process is described to the system using the JBPM process designer, itself an extension to the Eclipse Framework (http://www.eclipse.org). Processes are represented as a series of opaque boxes (one per task) each with associated input and output attributes which can be defined by the user. In addition to allow the monitoring of task level attributes in the results of the simulation these output attributes can be mapped onto process output attributes.

Simulation Approach

The simulator is constructed of a number of high-level components (Azvine et al. 2007):

- Event-Scheduler – This is used to fire events at the appropriate time within the simulators abstract time scale.
- Workflow System – This is used to ensure the execution of the process follows the process model
- Process Sources – These are used to generate new process instances which are then used as start points for new simulated processes.
- Task Execution Estimators – These estimate the outputs for each task.
- Resource Management System – This controls the assignment of actors to tasks.

Monitoring of a simulated process can be performed in a number of complementing ways within the simulator.

First, a KPI can be defined. These are most useful for monitoring the estimated performance of the process

during the simulation itself. An example would be, the time taken between the placing of an order and the associated installation task being completed.

Second, a rule could be defined. Rules are used to trigger an action, or consequent, when a condition is met either at the end of a process execution or at specific time intervals (for globally applicable rules). For example, a process level rule could be triggered when an installation task passes the target date set by the Service Level Agreement covering that task. Alternatively a global rule could be triggered when the time spent waiting for an installation to occur exceeds the current average wait time. Another usage for rules would be to monitor regulatory compliance.

Third, a KRI could be defined. These are similar to KPIs in many ways however their purpose is different. Intuitively a KPI is intended to measure business performance however a KRI is intended to measure risk factors. The simulator allows KPIs and KRIs to be defined separately for simplicity.

The simulator uses a custom syntax to define KPIs, KRIs, rules, and risks. The syntax itself is reasonably intuitive and those extracts embedded in the text will be accompanied by a plain language explanation.

In a departure from traditional business intelligence products these monitoring criteria can be defined dynamically by business users without knowledge of the underlying semantics. This removes the time-lag that could occur if a new KPI, rule, or KRI is required.

Once these monitoring criteria are defined and the simulator initialised then the simulation can be executed. The simulation results are then stored in a database to allow analysis to be performed at any time between the end of the simulation and the time the simulator is reset.

The simulator also allows for dynamic adjustment of resource levels to allow the user to easily check the performance of the process with differing resource scenarios.

Simulation of Risk

The previous section describes the basic functionality of the simulator. If run as above, the simulator will not consider the effect that appropriate risks may have on the execution or outcome of a process or task. However it will have triggered the appropriate rules, and have available results for each of the KPI and KRIs that have been defined.

In order to consider risks within the analysis, each risk that is to be tested needs to be defined in a form the simulator can understand. This is accomplished using a graphical editor similar to that provided for KPI definition and rule definition. The user will need to provide the two components of a risk – the probability of occurrence, and the consequence of the risk occurring.

When the simulation is then executed, it will proceed as before until a task with an associated risk is to be executed, the simulation engine will then probabilistically decide if the risk would occur in that

task instance and then factor in the consequences accordingly. By varying the consequences and probability the simulator can estimate how this business process will react to risk, and evaluate the KRIs and KPIs under these conditions.

In order to provide maximum flexibility, the simulator allows for the risks to be enabled and disabled for each simulation execution. This will allow additional analysis to discover the most debilitating risk in terms of effect on business performance which may be important for risk analysis. This sensitivity analysis is particularly important to identify areas that would benefit from process improvement. According to Papamichail & Robertson (2004) an additional benefit, of sensitivity analysis can be to increase confidence in the results of risk analysis, and aid the identification of the best way to proceed when presented with a variety of options.

Fuzzy Logic

Fuzzy logic can be used in the definition of KPIs and KRIs. It allows us to reason with continuous criteria (e.g. temperature) in an intuitive way. This document cannot include a detailed introduction to fuzzy logic however interested readers are directed to the wide variety of literature on that topic for further information.

One key use of fuzzy logic, inside the simulator is through the keyword `is` in a statement like "`@temperature is hot`" this will obtain the value of temperature from the process outputs and consult the associated fuzzy class for the term hot. It will then compare these and return a truth value for temperature belonging to the class 'hot' between 0 and 1 with 0 being not contained at all and 1 being completely contained in the class 'hot'.

Inside the simulator fuzzy classes are created using a graphical editor in which the user is asked for the name of the set, the number of classes to be defined, the upper and lower bounds of values for this class, and the type (triangular, trapezoidal, bell-shaped, list) of function that models the transition between classes. The function can be adjusted by dragging the function curve into the desired position.

As well as testing membership with fuzzy logic, we can also use the simulator to derive values for KPI and KRIs using fuzzy systems.

For example a KRI could be defined with rules similar to those in Figure 1 to derive a result that represents a risk value based on the cost and duration of the job. Within the current implementation of the simulation environment a fuzzy system can contain as many or as few rules as are required to express the property. It is worth noting at this point that fuzzy systems can also be used to define KPIs in a similar way.

In addition to using fuzzy logic to derive compound values as above, the simulator can also derive and display the fuzzy distribution that results from the application of a fuzzy set. This can help to identify the

risk profile of the process instances and could be an aid in identifying trends.

```
fs (
  case ( @cost is HighCost &
  @EstimatedDuration is LongJob )
    HighRisk,
  case (...)
    ...
);
```

Figure 1: An example fuzzy definition

EXAMPLE – CLIENT-ORDER PROCESS

Now that the basics of the simulator have been presented this section will introduce an example of the simulator operation and of the results that can be expected.

The process that will be used is the following simple ordering process for computing services. When a client of the company places an order it is received by a member of the company's contact centre. The order is then classified to determine which type of work is required. One of the company's engineers then takes the job and completes it appropriately at the customer's premises.

A graphical depiction of the process is shown in Figure 2.
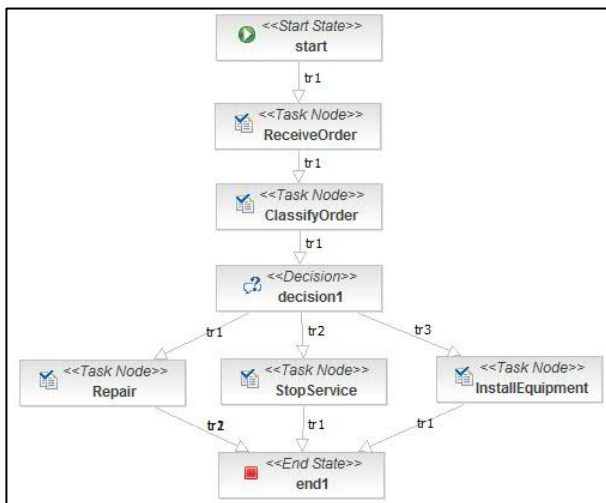
Figure 2: A Screenshot Showing the Process to Import

Initialising the Simulator

In order to simulate the process we first need to create the process with the editor. The process will need a single start node (which would be added ahead of ClassifyOrder) and a single end node (which is connected to all of the last 3 steps). Additionally an explicit decision node will need to be added between Classify Order and the lower activities to make the choice explicit. The process as it appears in the process designer is shown in Figure 2.

Once the simulator is loaded we can then import the process, set up the time parameters, and then complete the initialisation by setting a task assignment policy and process source. We assume from this point on that the source is configured to generate new processes at the rate to be tested, and an appropriate assignment strategy has been selected.

Once this is done we are free to define the KPIs, rules, KRIs, and risks that we would like to be simulated.

Key Performance Indicators

Consider that the business contains two types of employee who will take part in the process, OrderHandlers (initially 3) and Engineers (initially 3), and the company is interested in the overall end to end duration of the process which is typically called CycleTime. The definition of CycleTime in the monitor is very simple; we add a new KPI defined as: @duration; (which is the duration of the process) and after the simulation is run we can request a report of expected performance on this measure. We also select to monitor the length of the queue for each activity during the simulation and after the simulation is complete.

Suppose we then simulate the process under these assumptions and find the CycleTime is growing as the simulation continues (as in Figure 3). This is clearly a cause for concern to the business. We can now check the queues to try and find an indication of which activity(s) are causing the issue.
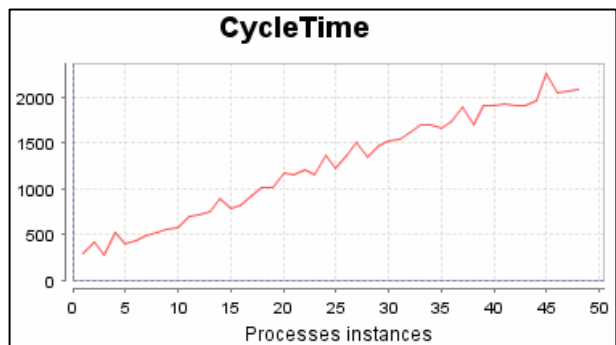
Figure 3: The CycleTime with 3 OrderHandlers

When checking the queues we find that both the ReceiveOrder and ClassifyOrder task queues grow in a similar way, but the queues for the engineer tasks are empty. From this we can conclude that changes need to be made to the order handling tasks to reduce the CycleTime. The process owner sees this data and would like to see how many additional staff in the OrderHandler role would be required to keep the CycleTime within target. Using the facilities in the simulator he or she can add additional resource and repeat the simulation and immediately see the effect of that change.

Rules

Rules are defined in a similar way to KPIs but are more suited to warning conditions or the monitoring of

regulatory compliance. For example, if the board of this business decide that any jobs with a CycleTime above 500 units should cause an alert condition. A rule can be defined as `@duration > 500;` and a consequent chosen to perform the desired alert. In this example case on screen alerts were used to present the information required for a process manager to drill-down into the processes and discover the reason for the excessively high CycleTime

Consequents in this scope are actions that the simulator can take when it detects a simulated process breeching the condition. A number of pre-built consequents are available within the simulator however a Java interface is available should a new consequent be required by the user.

Risks

Risks to a process can be as diverse as the processes themselves and accordingly we have defined a suitably fine-grained approach to the definition of risk within the simulation environment.

Each risk is given a name as an identifier and the probability of occurrence (expressed as a decimal). Each effect of the risk occurring can then be entered using the graphical editor and then saved. The editor itself can be seen in Figure 4.
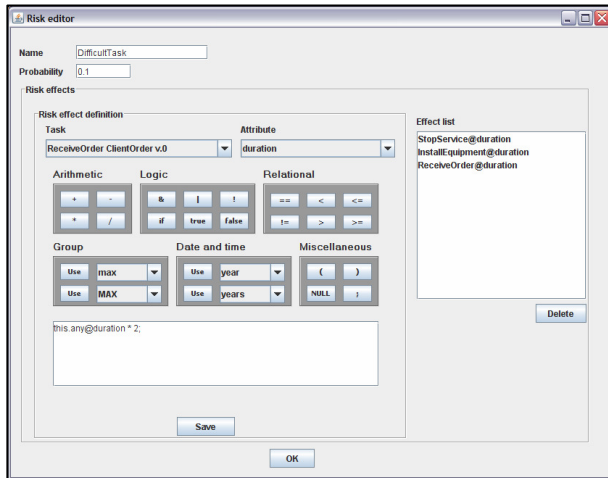
Figure 4: The Risk Editor for the Simulator

Once each risk to the process that needs to be simulated is defined, those to be considered in the current simulation execution can be activated. When activated the simulator will take into account the risk and apply to appropriate processes, allowing the business metrics to be calculated accordingly.

Key Risk Indicators

As discussed previously KRIs have many resemblances to KPIs but have very different purposes. Consider that this company has performed some risk identification and as a result has defined criteria to judge the risk level of each of the instances of this process (the criteria are given in Table 1). They wish to simulate this risk indicator to gain insight into the

current risk level and simulate the effect of changes in the execution of the process, and changes in the execution environment on the processes risk level.

Table 1: Risk Criteria

CycleTime	Cost	Risk Level
Short	Low	Low
Medium	Low	Low
High	Low	Medium
Short	High	Medium
Medium	High	High
High	High	High

Given the guidelines and rules used to define what is considered to be Low/Medium/High in each case we can use fuzzy logic in our analysis.

The first step is to use the graphical editor in the simulator to define the fuzzy sets for each of the 3 values (CycleTime, Cost, and Risk Level) using the appropriate functions and values. Once this is completed we assign the fuzzy sets to the attributes they represent (this could be either a task attribute or a process attribute – or both). We then define a fuzzy rule similar to that in Figure 1 as a KRI. The simulator will then use the fuzzy rules that are defined to generate a risk value for each process instance. A graph of these risk values can be seen in Figure 5. For the purposes of this example we have defined the Risk Value to sit within the range 0-100.
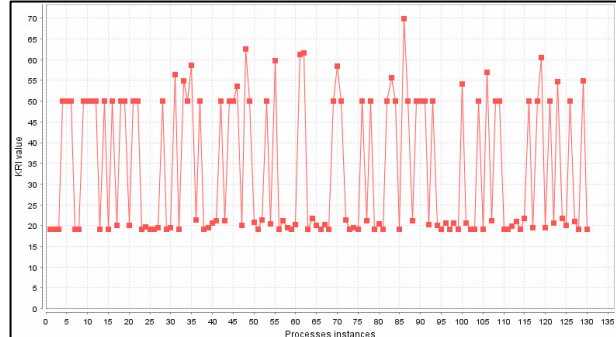
Figure 5: A Graph Showing the Application of the Fuzzy Rule Set,

Additionally the resulting fuzzy set for Risk Value can be displayed with its Centre of Gravity (Figure 6).
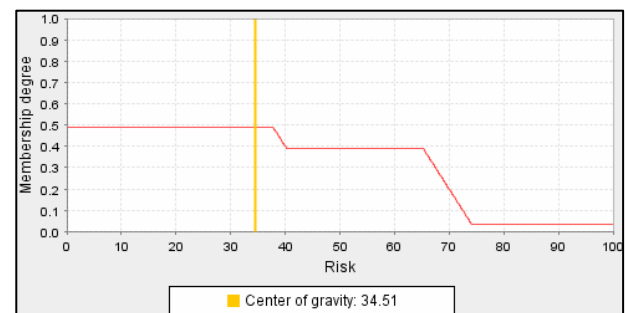
Figure 6: Fuzzy Set for Risk Value

This shows for each risk value, what membership value that risk would receive. Also the transition points between Low, Medium, and High can be deduced from the shape of the graph. The centre of gravity shows the mean risk.

CONCLUSIONS

The COSO framework defines Enterprise Risk Management as "… a process […], applied in strategy setting and across the enterprise, designed to identify potential events that may affect the entity, and manage risk to be within its risk appetite, to provide reasonable assurance regarding the achievement of entity objectives" (COSO, 2004).

This expounds the link between risk analysis and process performance. Therefore it seems clear that to maximize the business value of a process simulation environment, it should be possible (as shown in this paper) to integrate facilities for the performance analysis and monitoring of the process under test into the simulation environment.

Many experts are of the opinion that, for large businesses, qualitative analysis provides the most meaningful risk analysis, however we believe that quantitative risk analysis can provide clear business benefits when used in conjunction with a holistic performance framework such as our RTBI system (see Azvine et al. (2006) for more information).

In this paper we have shown how the business process simulator could be used in the simulation of business processes. We have also shown how risk events, and additionally how fuzzy logic can be utilised in this simulation framework to increase the business utility of the tool.

Furthermore, we have demonstrated how the use of fuzzy logic can help business users express qualitative measures that may change the outcome of the process and gain a quantitative insight into the estimated performance of the process, simplifying the sensitivity analysis of risks.

Since fuzzy logic provides a close analogue for human reasoning (Zadeh 1994) it allows natural expression of sometimes complex business conditions in an intuitive way, whilst preserving the ability to provide quantitative results thus lowering one of the boundaries of adoption for this tool in business.

In summary, it is important for the simulation of risk events in business processes to take place in an integrated framework that will allow performance analysis and risk analysis to be carried out side by side.

REFERENCES

Azvine, B.; Z. Cui; D.D. Nauck; B. Majeed, "Real Time Business Intelligence for the Adaptive Enterprise", In *E-Commerce Technology*. The 8th IEEE International Conference on Enterprise Computing, E-Commerce, and E-Services, pp.29-29, 2006

Azvine, B; Z. Cui; B. Majeed; M. Spott, "Operational Risk Management with Real-Time Business Intelligence", in BT Technology Journal, Vol. 25 No. 1, pp 154-167, 2007.

Bohn, C.; B. Kemp, "Enterprise Risk Management Quantification - An Opportunity". Presented at *Enterprise Risk Management Symposium*, Society of Actuaries, Chicago, IL, April 23–26, 2006.

COSO, "*Enterprise Risk Management — Integrated Framework*", 2004.

Davenport, T.H., "*Process Innovation: Reengineering Work through Information Technology*", Harvard Business School Press, Boston, 1993.

Ken McKinley, "Qualitative and Quantitative Risk Analysis", *Project Decision and Risk Analysis Journal*, 2005.

Jallow, A.K.; B. Majeed; K. Vergidis; A. Tiwari; R. Roy, "Operational risk analysis in business processes", in *BT Technology Journal*, Vol. 25, No 1, pp 168-177, 2007.

Lambert, J.H..; R.K. Jennings; N.N. Joshi, "Integration of Risk Identification with Business Process Models", In *Systems Engineering*, Vol. 9 No. 3, 2006.

Mestchian, P., Risk Intelligence – "From Compliance to Performance", In *Journal of Risk Intelligence*, Issue 1.

Neiger, D., L. Churilov, M. zur Muehlen, M. Rosemann, "Integrating Risks in Business Process Models with Value Focused Process Engineering", In *European Conference on Information Systems* (ECIS), 2006.

Zadeh, L.A., "Soft computing and fuzzy logic", *Software, IEEE*, Vol.11 No. 6, pp.48-56, 1994.

AUTHOR BIOGRAPHIES

Paul Taylor joined BT in October 2007 and currently works as a Researcher within the Intelligent Systems Research Centre. He holds a Degree in Computer Science (2006) and a Masters Degree in Computer Security (2007), both from the University of Birmingham. He has previously worked in areas such as Electronic Voting and Protocol Verification and is currently working on the Business Process Management project as part of the Real-Time Business Intelligence theme in BT.

Jesús Jimenez Godino joined BT in October 2005 and currently works as a Researcher within the Intelligent Systems Research Centre. He holds a Masters Degree in Computer Science from the University of Sevilla (2006). He is currently working on the Business Process Management project as part of the Real-Time Business Intelligence theme in BT.

Basim Majeed is a Principal Research Professional at the Intelligent Systems Research Centre within BT Research and Venturing. He holds a Masters degree (1987) and a PhD degree (1992) from the University of Manchester. He is a Member of the IET and IEEE, and a Chartered Engineer. He has published many papers in the areas of intelligent control, fault detection and isolation in dynamic systems, signal processing and intelligent data analysis. Since joining BT in 2003, he worked on a number of projects including the intelligent data analysis project within the Care in the Community Centre and the business process management project within the Real Time Business Intelligence (RTBI) theme.

DEVS MODELING OF RUN-TIME WORKFLOW SIMULATION AND ITS APPLICATION

Byoung K. Choi, Duckwoong Lee, and Dong H. Kang
Department of Industrial Engineering
KAIST
373-1 Guseong-dong, Yuseong-gu, Daejeon, 305-701, South Korea
E-mail: bkchoi@kaist.ac.kr

KEYWORDS

Run-time workflow simulation, Parallel simulation, Time synchronization, DEVS model, Workflow Management

ABSTRACT

Existing studies on workflow simulation focused mainly on simulating *process definition models* at *build time*. Recently, a novel workflow simulation method called run-time workflow simulation (RTWS) was proposed with which *process instances* at *run-time* can be simulated. This paper presents a formal DEVS model of the run-time workflow simulation module that can be embedded in an existing business process management system (BPMS). The presented DEVS model has been implemented and a *test-bed BPMS for RTWS experiment* was developed.

INTRODUCTION

Workflow management system (WfMS) is a software system that completely defines and automatically executes workflows in order to manage the actual flow of work in an organization (WfMC 1995). The software module in charge of managing the actual flow of work is called **workflow engine**, and the services it provides to execute the predefined workflows are referred to as enactment service. A portion of workflow engine responsible for enactment service is called **enactment server**. A predefined workflow model is called **process definition model** (**PDM**), while the one that is being executed is called a **process instance**. In recent years, a WfMS is often called a business process management system (BPMS) with an emphasis on orchestrating operational business processes (Delphi 2005, Smith and Fingar 2003). Along with this development comes an increased awareness of need for **workflow simulation** in evaluating and improving business processes (Smith and Fingar 2003).

As discussed in the next section, existing studies on workflow simulation are mainly focusing on simulating a PDM at its build time. Recently, the authors' group proposed a novel simulation method where the future (i.e., *enabled* and *inactive* activities) of process instances is simulated at run-time, which we call **run-time workflow simulation**. This paper represents a formal description of a run-time workflow simulation system (Hwang 2006) using the well-known **DEVS** model primitives (Hong 1997, Zeigler et al. 2000).

In the proposed run-time workflow simulation system, a local simulator is constructed for each **participant** and a separate *enactment server* is employed for the workflow simulation. As it is a parallel simulation, the **time synchronization** issues (Fujimoto 2000) have to be resolved.

BACKGROUND & RELATED WORKS

In order to make the paper self-contained, basics of enactment service mechanism are briefly described using Figure 1. For a given PDM, process instances (**PI**) are created. Shown in the figure is a PI having seven **activities** (including *Start* and *End*). Right after the completion of activity W1, the two activities W2 and W3 are *enabled*, and then the enactment server provides the following sequence of enactment services:

(1) Creates instances of the newly *enabled* activities W2 and W3
(2) Sends out W2 and W3 to respective **work-list handlers** (activity becomes a **new workitem** to be processed by the human participant).
(3) Receives the **completed workitem** W2 (In this scenario, W2 is completed first)
(4) Updates the state of the PI such that W2 becomes a *completed* activity.
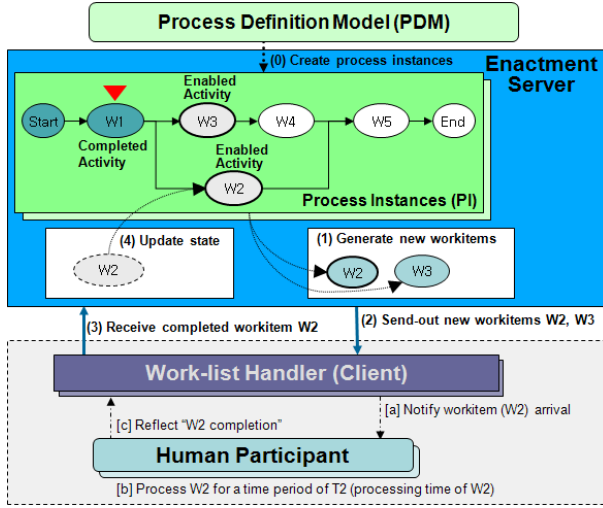
Figure 1: Enactment Service Mechanism

Upon receiving the new workitem W2, the work-list handler *notifies* its participant, and the participant works on W2 for a time period of T2 and *reflects* the results at the work-list handler so that the completed workitem is returned back to the enactment server.

The communications between the enactment server (workflow engine) and the work-list handlers are made based on the **interface standards** provided by Workflow Management Coalition (WfMC 1995). For this purpose, the workflow engine has a type of internal data called workflow **relevant data** that can be manipulated by work-list handlers as well as by the workflow engine.

Most of workflow simulation studies focused on validating and/or optimizing a given PDM via simulation. The basic approach here is that the PDM is converted into a formal model, such as Petri-net (Peterson 1981) or DEVS (Zeigler et al. 2000), and then simulation is performed using this converted model. In some commercial BPMS (Bizflow 2008, FileNet 2003), PDM is converted into a proprietary simulation language. There are quite a few studies reported in the literature (Aalst and Hofstede 2000, Bae et al. 1999, Chan 2004, Greasley 2003, Hong et al. 2003, Li et al. 2002, Li at el. 2005).

RUN-TIME WORKFLOW SIMULATION

Recently, a run-time workflow simulation (**RTWS**) method was proposed in which the future of the enactment service process of a BPMS is simulated at any point in time by a RTWS module embedded in the workflow engine (Hwang 2006). It was demonstrated that a BPMS equipped with a RTWS module could be used as a *simulation-based job shop scheduling* system (Hwang and Choi 2007). The RTWS module consists of an enactment server, a **synchronization manager**, and a number of **participant simulators** as depicted in Figure 2. In order to perform a run-time workflow simulation, a copy of each process instance (PI) being executed is obtained and the work-list handler ID of each activity in the PI is changed to the corresponding participant simulator ID. Now, the enactment server of the RTWS module has a set of modified process instances (PI*).
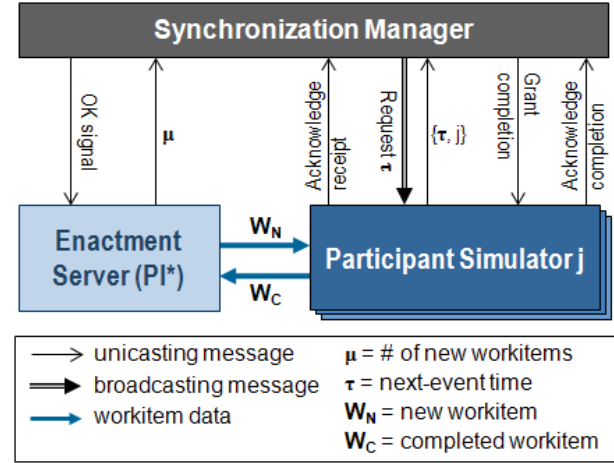
Figure 2: Run-time Workflow Simulation Module

The participant simulators in the RTWS module are **logical processors** in a parallel simulation system in which **centralized barriers** are often used for time synchronization (Fujimoto 2000). Starting from the point when an Enactment **Server** receives a completed workitem W_C from a Participant **Simulator**, the RTWS module of Figure 2 performs the following operations: a) Server waits until OK signal comes from Synchronization **Manager** and then updates PI* and generates new workitems, b) Server sends μ (number of new workitems) to Manager and then sends newly generated workitems W_N to Simulator, c) Simulator acknowledges the receipt of workitem to Manager, d) Manager waits until the receipt count equals to μ and then requests Simulators to send their *next-event time* τ, e) each Simulator **j** sends its τ to Manager, f) Manager waits until it receives τ from all Simulators and then grants the Simulator that has minimum τ value to complete its workitem, g) the granted Simulator acknowledges its workitem completion to Manager and sends the completed workitem W_C to Server. The wait-until points in Steps a, d, and f are *centralized barriers*.

DEVS MODEL OF RUN-TIME WORKFLOW SIMULATION

The internal structure of a parallel system is not easy to describe as a sequential algorithm. In this section, we describe the RTWS module (Figure 2) as DEVS coupled model (Ziegler et al. 2000) using the well known **DEVS model primitives** (Hong 1997) as shown in Table 1. For example, '**?**' symbol is used for *external input*, '**!**' symbol for *message output*, '**∼**' symbol for *condition*, etc.

Shown in Figure 3 is our RTWS module with the synchronization manager and enactment server represented as DEVS atomic models. The structure of the **Enactment Server** is quite simple: Normally, it stays in the initial state 'Ready'. When it receives both a completed workitem W_C and 'OK signal', it 1) updates the process instances PI^*, 2) generates new workitems $\{W_N\}$, 3) sends out μ (number of newly generated workitems) to the synchronization manager, and 4) sends out new workitems $\{W_N\}$ to participant simulators.

Table 1: DEVS Model Primitives
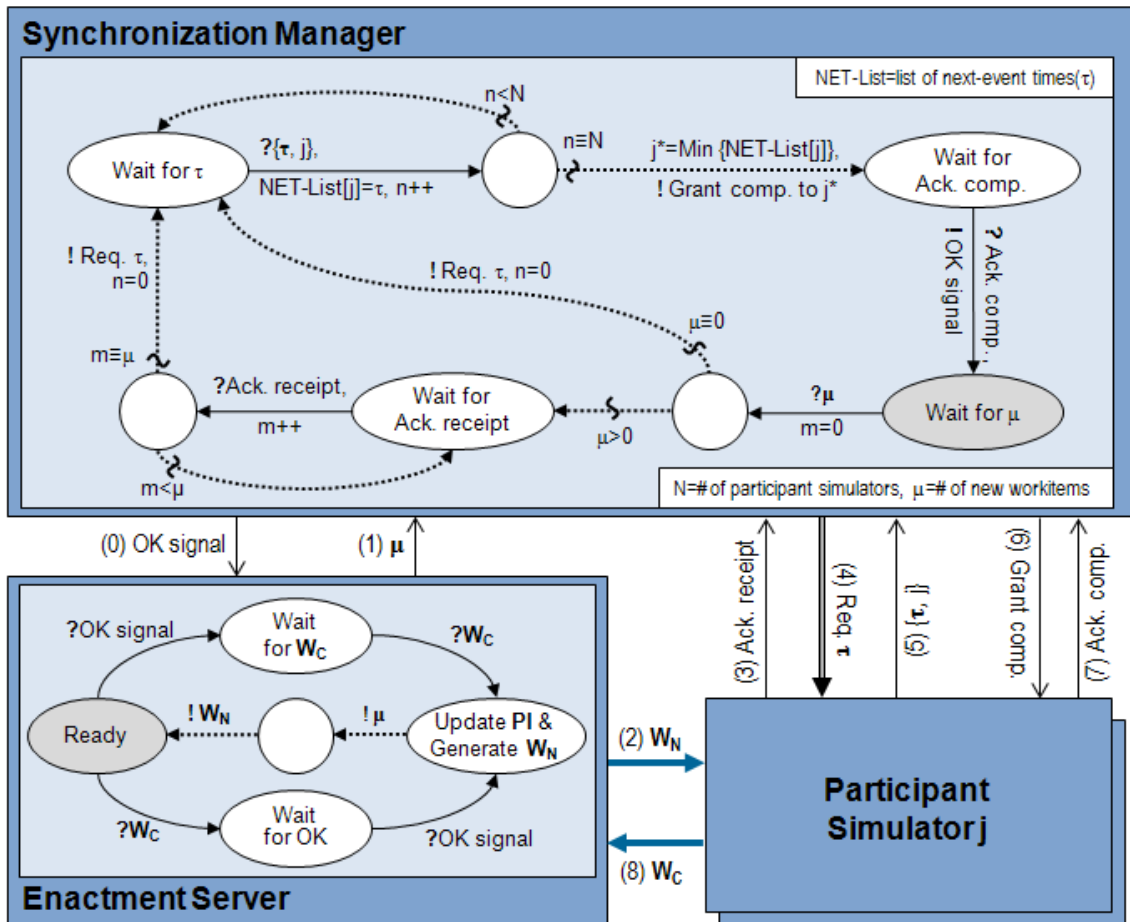
Figure 3: DEVS Model of Synchronization Manager

The DEVS model of Synchronized Manager (**SM**) has four 'Wait for' states: 1) Wait for μ, 2) Wait for acknowledge receipt, 3) Wait for τ and 4) Wait for acknowledge completion. Once initialized, SM stays in the 'Wait for μ' state. Upon receiving μ, SM sits in the 'Wait for Ack. receipt' state until the 'receipt' count becomes μ and then moves to the 'Wait for τ' state while sending out 'Request τ' message to all participant simulators (if $\mu \equiv 0$, it directly moves to the 'Wait for τ' state). And SM waits there while storing the next-event time (NET) τ of simulator j in NET-List[j]. If it receives $\{\tau, j\}$ from all simulators, 1) the simulator j^* that has the

smallest τ is selected (i.e., j*=Min NET-List[j]), 2) 'Grant comp.' message is sent to the selected simulator, and 3) its state is changed to 'Wait for Ack. comp.' Finally, when SM receives the 'Ack. comp.' message from the selected simulator, it sends out 'OK signal' to the enactment server and moves to the initial state 'Wait for μ' to initiate the next cycle.

Shown in Figure 4 is a *coupled DEVS model* of a 'single machine' participant simulator. A typical *human participant* of a BPMS may be regarded as a **single machine system** in which arriving jobs (i.e., new workitems) are put into a queue and a job is selected for processing based on a dispatching rule. The coupled model consists of Coordinator, Queue and Processor each of which is an atomic DEVS model. It is a typical single machine system model.

The **Coordinator** model has four 'Wait for' states: 1) Wait for new workitems W_N, 2) Wait for τ 3) Wait for Grant completion and 4) Wait for completed workitem W_C. Normally, it stays in 'Wait for W_N' while passing received W_N to Queue and sending 'Ack. receipt' message back to Enactment Server. If 'Req. τ' message is received, it moves to 'Wait for τ' after sending out 'ask τ' message to Processor. Then upon receiving **return τ**, it moves to 'Wait for Grant comp.' after passing received τ to Synchronization Manager. At this point, its state may be changed back to 'Wait for τ' (if 'Req. τ' is received) or to the next state 'Wait for W_C' (if 'Grant comp.' message is received). Finally, it moves to the initial state if it receives W_C from Processor.
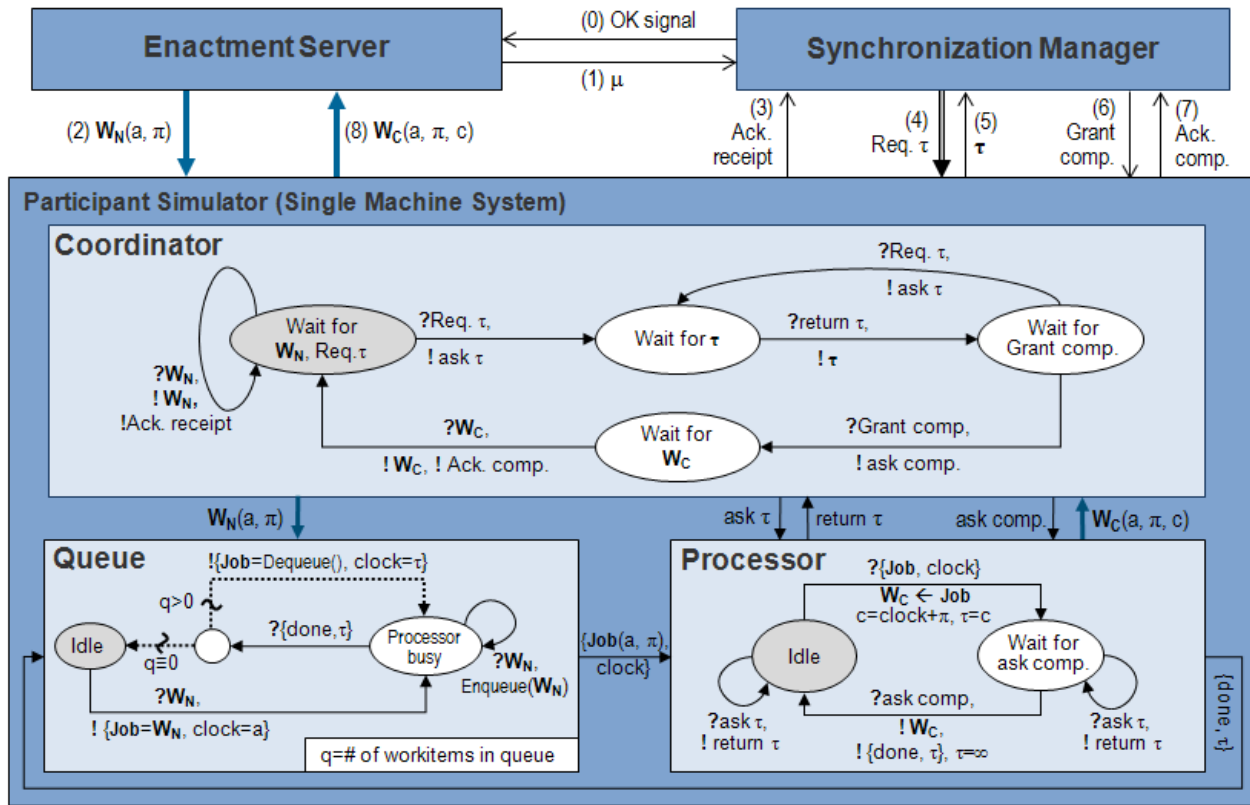
Figure 4: DEVS Model of a Participant Simulator

The **Queue** model receives a new workitem W_N (with *arrival time-stamp* 'a' and *processing time* 'π') from the Coordinator. The received W_N is either sent to idle Processor or stored in a queue (i.e., Enqueue) when Processor is busy. Upon receiving a 'done' signal, it retrieves a job (W_N) from the queue (Dequeue) and sends the job (together with the job-start time '**clock**') to the Processor.

The reader is advised to examine the **Processor** model shown in Figure 4 which should be self-explanatory (Note: c = completion time-stamp).

IMPLEMENTATION AND APPLICATION

The DEVS model of a run-time workflow simulation (RTWS) module has been implemented and embedded in a test-bed BPMS named Harmony® (Choi and Hwang 2005). As it is not convenient to perform RTWS

experiments under a real life environment, we have developed a test-bed BPMS for run-time workflow simulation as shown in Figure 5.
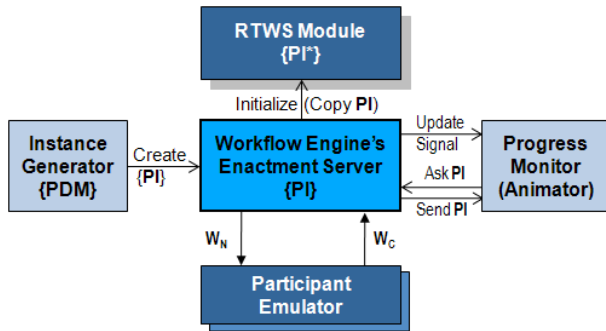
Figure 5: A Test-bed BPMS for RTWS Experiments

The test-bed BPMS for the RTWS experiment consists of the original enactment server, the RTWS module, an instance generator, a number of participant emulators, and a progress monitor. Each of the work-list handler is replaced by a **participant emulator** which is a *scaled real-time simulator* representing a human participant. The **instance generator** creates a sequence of process instances {PI} in scaled real time according to a predefined schedule. With the two components added to the original enactment server, we can create a virtual enactment service scenario. Figure 6 shows a GUI for *process instance generation* (left-middle) and *participant emulation* (right).
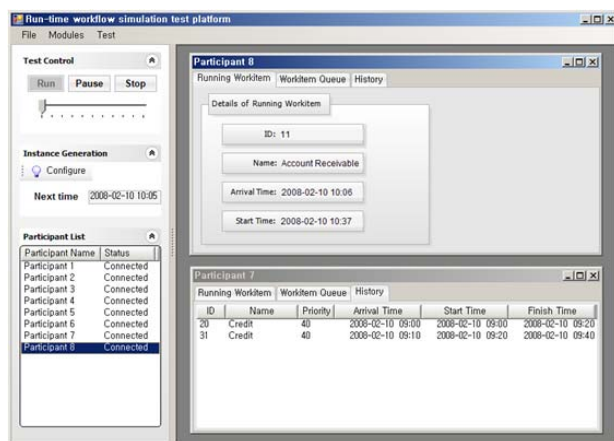
Figure 6: GUI for Process Instance Generation and Participant Emulation

The **progress monitor** component displays the state changes of the selected process instances (i.e., it animates the progresses of process instances). Also, it is responsible for visualizing the overall progress of the business process. Shown in Figure 7 is a display screen for the progress monitor (C = completed activity, A = active activity, I = inactive activity, X = excluded activity).
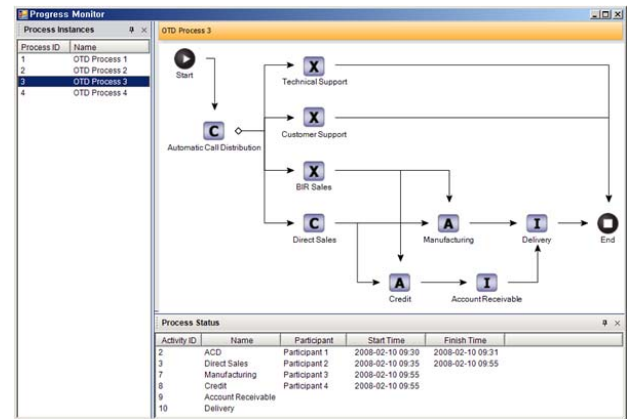
Figure 7: Progress Monitor's GUI

At any point in time, the user may trigger the **RTWS module** to start a workflow simulation. As discussed earlier, the RTWS module (See Figure 2) gets a copy (PI*) of each process instance PI that is stored in a DB and performs simulation in an as-fast-as-possible manner. In the current implementation, every participant is modeled as a 'single machine system' consisting of a queue and a processor (See Figure 4).

CONCLUSION

Presented in this paper is a formal DEVS model of a run-time workflow simulation (RTWS) module that can be embedded in an existing BPMS (business process management system). The proposed RTWS module is a parallel simulation system having a number of *logical processors* (participant simulators and enactment server) and a *synchronization manager*, and centralized barriers are used for time synchronization. Also presented is a test-bed BPMS for RTWS experiment consisting of the original enactment server, a RTWS module, a process instance generator, the participant emulators, and a progress monitor. The main purpose of the research is to demonstrate the validity of the presented DEVS model using a test-bed BPMS. Further research efforts might be need in order to apply it to commercial BPMSs.

REFERENCES

Aalst, W.M.P. and Hofstede, A.H.M., 2000, Verification of Workflow Task Structures: A Petri-Net-Based Approach, *Information Systems*, v.25, no.1, pp.43-69.

Bae, J.S., Jeong, S.C., Seo, Y.H., Kim, Y.H. and Kang, S.H., 1999, Integration of Workflow Management and Simulation, *Computers & Industrial Engineering*, v.37, pp.203-206.

BizFlow®, 2008, http://www.handysoft.com

Chan, D.Y.K., 2004, Design An Effective Workflow in Simulation, *The 8th International Conference on Computer Supported Cooperative Work in Design Proceedings*, v.2, pp.324-328.

Choi, B.K. and Hwang, H.C., 2005, Architecture of A 3-layered Closed-loop BPMS Harmony®, *KAIST VMS Lab Technical Report*, http://vms.kaist.ac.kr/publications/TechnicalReport/3layer.pdf (in Korean)

Delphi, 2005, *BPM 2005 Market Milestone Report*, http://www.delphigroup.com/research/whitepaper_request_download.htm

FileNet, Bruce Silver Associates, 2003, Event-driven Business Process Management. *Industry Trend Reports*, http://whitepapers.silicon.com

Fujimoto, R.M., 2000, *Parallel and Distributed Simulation Systems*, John Wiley & Sons.

Greasley, A., 2003, A Simulation of A Workflow Management System, *Work study*, v.52, no.5, pp.256-261.

Hong, G.P., 1997, DEVS-Based Framework for Logical and Performance Analysis of Discrete Event Systems, *Doctoral Thesis*, *Department of Electronic Engineering, KAIST*.

Hong, K.J., Lee, J.K., Kim, D.H. and Kim, T.G., 2003, DEVS Framework Instrumented with Database for Web-Based Workflow Modeling Simulation, *Society for Computer Simulation (Simulation Councils, Inc.)*, v.31, no.3, pp.113-118.

Hwang, H.C. and Choi, B.K., 2007, Workflow-based Dynamic Scheduling of Job Shop Operations, *International Journal of Computer Integrated Manufacturing*, v.20, no.6, pp.557-566.

Hwang, H.C., 2006, Development of a Workflow Engine based Simulation System, *Doctoral Thesis, Department of Industrial Engineering, KAIST*.

Li, J.J., Casati, F. and Shan, M.C., 2002, Business Process Simulation with HP Process Manager, *Simulation series*, v.34, no.1, pp.291-295.

Li, R., Zhu, Z., Wang, X., Liu, L. and Jiang, X., 2005, Workflow Simulation and Its System Development, *Proceedings of SPIE-the International Society for Optical Engineering*, v.6041, pp.604104.

Peterson, J.L., 1981, *Petri Net Theory and the Modeling of Systems*, Prentice Hall.

Smith, H. and Fingar, P., 2003, *Business Process Management: The Third Wave*, Meghan-Kiffer Press, 2003.

Workflow Management Coalition (WfMC), WfMC-TC-00-1003, 1995, *The Workflow Reference Model*, http://www.wfmc.org

Zeigler, B., Praehofer, H. and Kim, T., 2000, *Theory of Modeling and Simulation*, Academic Press: Boston.

AUTHOR BIOGRAPHICS
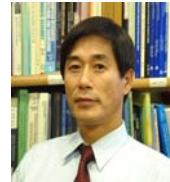
Byoung K. Choi is a professor of the Department of Industrial Engineering at KAIST since 1983. He received a BS from Seoul National University in 1973, a MS from KAIST in 1975, and a Ph.D. from Purdue University in 1982, all in Industrial Engineering. His current research interests are system modeling and simulation, BPMS, simulation-based scheduling, and virtual manufacturing.

Duckwoong Lee is a PhD candidate student in the Department of Industrial Engineering at KAIST. He received as BS from Ajou University in 2002, a MS from KAIST in 2004, all in Industrial Engineering. His research interests are in the area of BPMS, system modeling and simulation, and parallel and distributed simulation. He can be reached at ldw721@vmslab.kaist.ac.kr
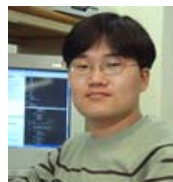
Dong H. Kang is a PhD candidate student in the Department of Industrial Engineering at KAIST. He received as BS from KAIST in 2003 in Computer Science, a MS from KAIST in 2005 in Industrial Engineering. His research interests are in the area of BPMS, system modeling and simulation and RFID application. He can be reached at donghun.kang@vmslab.kaist.ac.kr

MANAGING RESOURCES AT AN INTERNATIONAL AIRPORT

Javier Otamendi[1], Pablo García Ansola[2], Miguel Poyatos[3], José Manuel Pastor[3], Andrés García Higuera[2]

[1]Universidad Rey Juan Carlos
Facultad CC. Jurídicas y Sociales
28032 Madrid, Spain

[2]University of Castilla-La
Mancha
School of Industrial Engineering
13071 Ciudad Real, Spain

[3]University of Castilla-La
Mancha
Polytechnic School
16000 Cuenca, Spain

KEYWORDS

Optimization, Scheduling, Simulation.

ABSTRACT

An application of a combination of scheduling theory, simulation modelling, visualization and control is presented. The system under study deals with human resources and assets management to perform the daily operations at an international airport. The paper describes the simulation-based tool that is being developed and pinpoints the cost analysis that is being performed to decide on the assignment and the level of the resources. Following the modelling approach, a fully detailed explanation of the data requirements, the decision variables and the output variables is given, highlighting the structure of the cost objective function.

INTRODUCTION

Spain is suffering important changes in the air service industries. Several new private airports are being built all over the country to attract low cost companies. Ciudad Real is a city located 200 kilometres south of Madrid and its airport is to start operations in July of 2008. For that reason, the company is starting to develop tools that will help in the management operations with the aim of optimally allocating resources.

The first objective of the research was the graphical definition and representation of the operations that are going to be carried at the parking positions: embarking and disembarking of the passengers and the crew and all of the associated activities. The tool has been developed in JAVA to favour visualization and on-line control (Pastor et al 2007).

The interface (Figure 1) shows on the right panel four parking positions with their corresponding assigned resources and on the left panel the pool of idle resources.

The JAVA application that has been developed is executed after the spreadsheet simulation, which, using the VisualBasic routines, generates a JAVA-based Gantt chart with the operations to be performed per resource. It gives the user several visualization possibilities that range from the static representation at a particular time to the dynamic representation of part

or the totality of the time horizon reflected in the Gantt chart.
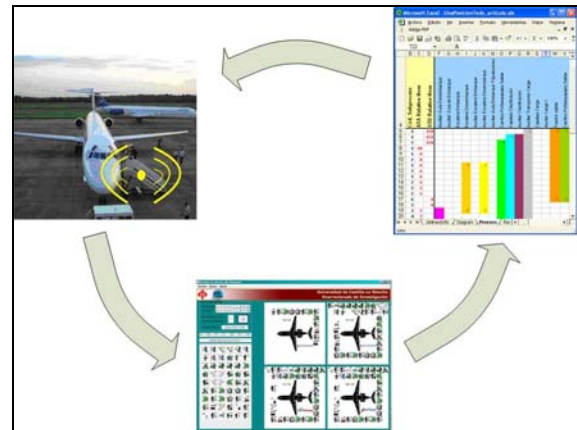
Fig. 1: Description of the Decision Making System

The tool interacts both with MsExcel spreadsheets and with the actual resources via RFID, so data is automatically and constantly being updated. Therefore, all the resources are continuosly located and a forward simulation and visualization of the near future activities might be performed and scheduling decisions might accordingly be taken.

The management of the facilities is feeling more confortable with the tool as days go by so to ask for minor refinements of the tool for easier use and decision making.

The addressing of the second objective immediately followed: the staffing and the scheduling of the resources. As of today, the management is still not engaged in the development and implementation of an automatic decision system for staffing and scheduling the resources since the airport is still young and too small as to perform the scheduling directly from the visualization information included in the interface. Therefore, the focus right now is to correctly define the information to include in the platform and properly show it on the screen, although the software should be ready for including a more sophisticated search engine to perform the optimization.

The structure of the optimization problem to be solved is the focus of this article. The stress is on the calculation of the cost objective function, that includes

not only direct costs but also a penalty term for not fulfilling timeliness requirements is also included.

THE MODELLING APPROACH

Lawrence Boland proposes in his book "The Foundations of Economic Method" (Boland 1982) the utilization of a methodology dependent on the problem in hand (Problem-Dependent Methodology). In fact, he proposes modelling as a means to represent reality, each model representing a different problem. This methodology, or method of analysis, might be renamed as the modelling approach.

Three stages might be distinguished when following this approach (Figure 2):

1. Definition of the problem

2. Modelling of the system

3. Experimenting with the model and decision making

Problem Definition

The first stage consists of the conceptualization of the real system under study. A system might be defined as "process … or any combination of machines, methods, personnel or other resources which transform a given input in an output that present one or more observable responses" (Montgomery 1991). It is clear that the decision variables **x,** the data **z**, or phenomena that interrelate within the system, and the output criteria **y** that want to be quantified as a result of the interactions within the system must be specified.
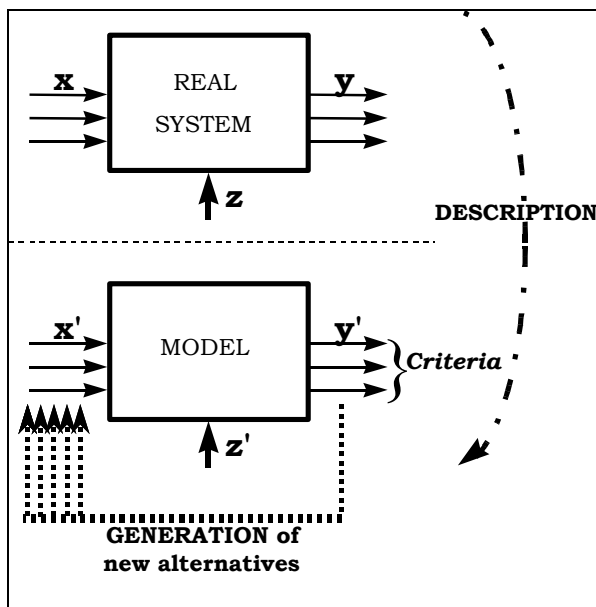
Figure 2: Description of the Real System into a Model

It seems essential that the problem be delimited, clearly specifying not only the variables and the data but also their interdependencies within the system. In very complex situations, it is also necessary to define the limits of the system and the appropriate level of detail according to the objectives of the study.

System Modelling

The study of a system is rarely performed directly on it. Either it is not possible to alter normal operations or it is not feasible to repetitively experiment with it so it is necessary to develop a model which acts as an abstraction of the real system and that is liable and ready for experimentation. The success of the analysis process, and consequently, of the implementation of the decision, clearly depends on this abstraction.

The model will be characterized by a set of input variables **x'** and input data **z'**, which are a representation of the corresponding **x** and **y** of the real system, and another set of output variables **y'** that are a function of the input variables and the existing relationships between them.

Computer simulation has received a lot of attention in the last decades to abstract and model complex systems under uncertainty, in many areas but specifically in scheduling and sequencing (Fisher and Ittner 1999, Hershauer and Ebert 1975, Hollier 1968, Macaskill 1973, Moccellin and Nagano 1998, Wein and Chevalier 1992). The simulation models jointly consider efficacy, or degree of attainment of the optimal solution, and efficiency, or time to achieve it.

Experimentation With The Model And Decision Making

If the model is valid and credible, it is ready for experimentation and for decision making, that is, ready to search for the adequate values of the input variables.

The first stage of this third phase is the characterization of the criteria that allow for the evaluation of each alternative. The different criteria are computed out of one or several of the output variables **y'** of the model and out of the subjective information introduced by the decision maker. If there exist several criteria simultaneously, they should be aggregated so that the alternatives might be compared against each other.

In the second stage, the aim is to obtain information on the values of the input variables that improve the values of the output criteria. Any method of analysis must generate, first, different alternatives or combinations of values for the parameters **x'**; then, it must evaluate the alternatives individually with respect to each of the criteria that characterize the system, and, finally, the algorithm must select the alternative that seems superior to the rest of alternatives.

THE SYSTEM

Operations

The system under study includes all the operations that will be carried at the airplane parking positions at the airport. The loading of passengers includes the escorting of the passengers from the gate to the stairs of the airplane, while the unloading includes the walk from the airplane to the arriving lounge. The resources that are available are operators that escort either passengers or crew and the stairs to enter or exit the aircraft.

Decision variables

The human decision maker might have an influence in the result of the assignment process of tasks to resources. The decision implies that the resources just assigned are freezed and cannot be assigned to any other task. So, the first decision variable is the actual ASSIGNMENT, or ORDER, in which the tasks are performed. This is a tactical decision.

The rest of the variables are mainly strategic. The first, medium-range, set of variables in this group is the QUANTITY OF RESOURCES of each type:

1. Human Resources

2. Motorized and Unmotorized Equipment

The QUANTITY of resources is to be parameterized so that the user might alter the configuration according to the real value before the execution of the application.

The values for the variable ASSIGNMENT are not specified before the execution but are obtained as a result of the execution of the decision making routine.

Output variables

With regard to the output variables, results or criteria, two types might be mentioned. The first type relates to the input- output variable duality. The assigned value to the input variable ASSIGNMENT is an output value of the optimization process, and the decision maker treats it as such, since it is an outcome of the application.

The second type, on the other hand, has to do with the search of the proper value of ASSIGNMENT. For each one of the possible ASSIGNMENT values, a value must be calculated for the measurable criteria, three in this case:

1. TOTAL COST of the ASSIGNMENT, which is to be minimized. The cost is composed of direct labour costs and direct equipment cost

2. PERCENT SATISFACTION or SERVICE LEVEL, which is to be maximized. Out of the total tasks, the percentage of tasks that are completed within the allotted time is calculated.

3. OCCUPATION LEVEL, which might be close to a given target, usually around 70-80%.

Additionally, it is necessary to aggregate the two first competing criteria in a single measure, which is liable to be optimized. The decision is for the conversion of SERVICE LEVEL into a PENALTY cost, which accounts for the lateness in completing the task. This PENALTY might then be easily included into the TOTAL COSTS as an additional cost.

Data

First, the input data related to the flights is to be included: company, flight number, arrival time/date, departure time/date and passengers.

Then, the data of the human resources is to be specified: code, name, number of available resources, cost for renting a resource and type.

Finally, the data for the equipment should also be incorporated.

THE INTERFACE FOR CALCULATING THE COST FUNCTION

Interface

Figure 3 shows the main screen of the application. In the "Archivo" menu, there exists the possibility of accessing to the Excel files and update the input values that might be manually changed to go with the data that is automatically updated.
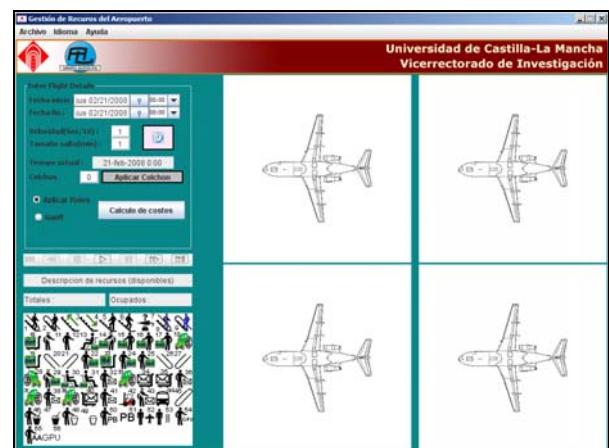
Figure 3. Main Interface

The cost calculations are accessible by pressing the "Calculo de costes" button. The Gantt chart is automatically generated and shown on an emerging window.

Input data: flights

Figure 4 shows the scheduled flights, with times and passengers.

Airlines Name	Flight Number	Arrival Date_Time	Departure Date_Time	Passengers
AIR NOSTRUM				
	YW 100	20080201 17:30	20080201 18:30	100
	YW 101	20080201 17:30	20080201 17:30	120
AIR EUROPA				
	UX 100	20080201 18:30	20080201 20:30	40
	UX 101	20080201 18:45	20080201 19:45	80
IBERIA	IB 100	20080201 19:10	20080201 20:30	76
	IB 101	20080201 19:30	20080201 20:30	49

Fig. 4: Data for Flights

Input data: Human Resources

Figure 5 includes the information about labor:

- Code
- Resource Name
- Quantity available
- Cost per hour per unit
- Group code

Code	Resource	Quantity	Cost/hr	Group
0	Auxiliar Guía Desembarque	2	7.39	Auxiliar
1	Auxiliar Guía de Embarque	2	7.39	Auxiliar
2	Escalera Embarque	2	40.58	Escalera
3	Escalera Desembarque	2	40.58	Escalera
4	Auxiliar Escalera	2	7.39	Auxiliar
5	Auxiliar Escalera	2	7.39	Auxiliar
6	Atención Lost&Found	2	9.51	Atención
7	Auxiliar Guía Desembarque	2	7.39	Auxiliar
8	Auxiliar Guía Embarque	2	7.39	Auxiliar
9	Carritos Portaequipajes	2	9.51	Carritos
10	Capataz Clasificacion	2	11.93	Capataz

Fig. 5: Data for Human Resources

Input data: Equipment

Figure 6 depicts the screen that is used to input the necessary data for the equipment. For scheduling purposes, the quantity of the resources and the unit cost are the critical information.

Motorised Equipment					60,026
GSE Family	Brand	Type	Qty	Unit Cost	Total Cost
Tractor	TemG	TG-5000	3	1,389	4,167
Passenger stairs mot	TemG	TG-2244	1	1,420	1,420
Passenger stairs mot	TemG	TG-2258	1	1,863	1,863
Conveyorbelt	TemG	TG-6000	3	1,360	4,080
Electrical Passenger stair	FMC	Universal (~TG2258)	1	1,958	1,958
Electrical tractor	Charlatte	T135	4	798	2,394
Electrical beltloaders	TLD	NBL E	3	1,125	3,375
Pusback small	TemG	TG-5016	2	3,039	6,078
Toilet truck	TemG	TG-0217	1	1,828	1,828
Water truck	TemG	TG-0218	1	1,762	1,762
Ambulift	TemG	TG-0601	1	5,454	5,454
GPU	Houchin	C690 100 + 28vdc	1	1,464	1,464
ASU (Air start unit)	TLD		1	3,517	3,517
Pushback Large (40T)	Schopf	F300	1	5,881	5,881
Loader 7T	Trepel	CHAMP70W	2	4,769	9,539
Passenger bus	Noge	Aertour	1	5,246	5,246

Unmotorised Equipment					3,169
GSE Family	Brand	Type	Qty	Unit Cost	Total Cost
Baggage cart	TemG	TG-0104	10	81	808
Pallet dollies	TemG	TG-0018	10	199	1,986
Passenger stairs towable	TemG	TG-0022	1	359	359
Passenger stairs towable	TemG	TG-0023	1	388	388
Passenger stairs towable	TemG	TG-0025	1	435	435

Fig. 6: Data for Equipment

Output data: Staffing Requirements

The first output screen is shown in Figure 7. It shows the number of calls that have simultaneously been placed for each of the resource time, if unlimited resources were available. It also shows the peak interval for that maximum level of resources.

	A	B	C	D	E
178	Recurso	Fecha inicial	Fecha final	Accesos simultaneos	
179	Auxiliar Guía Desembarque	2007-10-17 17:32:00	2007-10-17 17:43:00	3	
180	Auxiliar Guía de Embarque	2007-10-17 17:43:00	2007-10-17 18:01:00	3	
181	Escalera Embarque	2007-10-17 17:55:00	2007-10-17 18:01:00	3	
182	Escalera Desembarque	2007-10-17 17:24:00	2007-10-17 17:32:00	3	
183	Auxiliar Escalera Embarque	2007-10-17 17:55:00	2007-10-17 18:01:00	3	
184	Auxiliar Escalera Desembarque	2007-10-17 17:24:00	2007-10-17 17:32:00	3	
185	Atención Lost&Found	2007-10-17 17:45:00	2007-10-17 18:10:00	3	
186	Auxiliar Guía Desembarque Trip	2007-10-17 17:40:00	2007-10-17 17:47:00	3	
187	Auxiliar Guía Embarque Tripula	2007-10-17 17:47:00	2007-10-17 17:50:00	3	
188	Carritos Portaequipajes Salida	2007-10-17 16:00:00	2007-10-17 17:30:00	3	
189	Capataz Clasificacion	2007-10-17 15:55:00	2007-10-17 17:30:00	3	
190	Auxiliar Clasificacion	2007-10-17 15:55:00	2007-10-17 17:30:00	3	
191	Auciliar Transporte Carga	2007-10-17 15:52:00	2007-10-17 17:35:00	3	
192	Capataz Carga	2007-10-17 17:35:00	2007-10-17 18:00:00	3	
193	Auxilar Carga 1	2007-10-17 17:42:00	2007-10-17 18:00:00	3	
194	Auxiliar Carga 2	2007-10-17 17:43:00	2007-10-17 18:00:00	3	
195	Auxiliar Carga 3	2007-10-17 17:43:00	2007-10-17 18:00:00	3	
196	Tractor salida	2007-10-17 15:52:00	2007-10-17 18:00:00	3	
197	Carritos Portaequipajes Salida	2007-10-17 15:52:00	2007-10-17 18:00:00	3	
198	Cinta Transportadora Delantera	2007-10-17 17:55:00	2007-10-17 18:00:00	3	
199	Cinta Transportadora Trasera C	2007-10-17 17:55:00	2007-10-17 18:00:00	3	
200	Auxiliar Descarga 1	2007-10-17 17:21:00	2007-10-17 17:38:00	3	
201	Carritos Portaequipajes Llegada	2007-10-17 17:21:00	2007-10-17 18:03:00	3	
202	Auxiliar Descarga 2	2007-10-17 17:21:00	2007-10-17 17:38:00	3	
203	Auxiliar Descarga 3	2007-10-17 17:21:00	2007-10-17 17:43:00	3	

Fig. 7: Gross Requirements

For the resources, a Gantt chart (Figure 8) is also displayed with the color coding indicating type of operation: preparation, operation and post-operation.

Fig. 8: Gantt Chart

By moving the mouse over any bar, information is shown regarding the type of resource and the current time.

Output data: Costs per passenger

The first cost screen (Figure 9) includes the information about the cost per flight and per individual passenger.

Flight	Total Cost	Pax	Cost/Pax
YW 100	440.6429	100.0	4.41
YW 101	440.6429	120.0	3.67
UX 100	440.9259	40.0	11.02
UX 101	440.3284	80.0	5.50
IB 100	440.9259	76.0	5.80
IB 101	440.3284	49.0	8.99

Fig. 9: Cost per passenger

Output data: Direct labour

Figure 10 depicts the costs per resource and also their utilization ratio.

Resource Code	Name	Use (min.)	Direct Cost (€)	% usage
1	Auxiliar Guía Desembarque	247	30.41	74.62
35	Auxiliar Guía Desembarque	248	30.53	74.92
37	Auxiliar Guía de Embarque	244	30.04	73.72
15	Auxiliar Guía de Embarque	224	27.58	67.67
2	Escalera Embarque	64	43.28	19.34
2	Escalera Embarque	20	13.53	6.04

Fig. 10: Costs per Resource

Output data: Timeliness (Delays)

Figure 11 shows the output screen that summarizes the calculation of the delays per resource, including the flight number and also the time at which the delay happened.

Fig. 11: Delays

Output data: Total Costs

Finally, the total costs are calculated by adding the penalty costs (50 euros/min) associated with delays to the traditional direct costs (Figure 12).

TOTAL COSTS	110,238.10
DIRECT COSTS	**65,838.10**
DIRECT LABOUR	2,643.79
MOTORISED EQUIPMENT	60,025.80
UNMOTORISED EQUIPMENT	3,168.50
PENALTY DELAYS	**44,400.00**
DELAYS	888.00
PENALTY FACTOR	50.00

Fig. 12: Summary of Costs

CONCLUSION

A platform to manage the resources at a young international airport is being developed with care. State-of-the-art tools and techniques are being incorporated as the management of the facilities start to demand their use.

To the visualization capabilities, a complete cost structure has been incorporated so that, given a set level of resources and a plan, the cost of operations is calculated.

The future calls for incorporating a search or optimization engine to optimize the assignment decisions. With the engine already implemented, it will be easier to start incorporating other online or automatic decision support tool, like RFID.

REFERENCES

Boland, L. 1982. *The Foundations of the Economic Method*. George Allen and Unwin, London.

Fisher M.L. and C.D. Ittner. 1999. "The Impact of Product Variety on Automobile Assembly Operations: Empirical Evidence and Simulation Analysis." *Management Science*, vol. 45, no. 6, 771-786.

Hershauer J.C. and R.J. Ebert. 1975. "Search and Simulation Selection of a Job-Shop Sequencing Rule." *Management Science*, vol. 21, no. 7, Theory Series, 833-843.

Hollier R.H. 1968. "A Simulation Study of Sequencing in Batch Production." *Operations Research*, vol. 19, no. 4, 389-407.

Macaskill J.L.C. 1973. "Computer Simulation for Mixed-Model Production Lines." *Management Science*, vol. 20, no. 3, Theory Series, 341-348.

Moccellin J.V. and M.S. Nagano. 1998. "Evaluating the Performance of Tabu Search Procedures for Flow Shop Sequencing." *The Journal of the Operational Research Society*, vol. 49, no. 12, 1296-1302.

Montgomery, D.C., 1991. *Diseño y Análisis de Experimentos*, Grupo Editorial Iberoamérica, S.A. de C.V., México, D.F., México.

Pastor, J.M., J. Otamendi and A. García Higuera. 2007. Graphical Visualization for the Management of Spreadsheet Simulations, in *Proceedings of EUROSIM 2007*.

Wein L.M. and P.B. Chevalier. 1992. "A Broader View of the Job-Shop Sequencing Problem." *Management Science*, vol. 38, no. 7, 1018-1033.

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 at the Rey Juan Carlos University in Madrid. E-mail: franciscojavier.otamendi@urjc.es.

PABLO GARCÍA ANSOLA received degrees in Computer Science Engineering, MBA and Computer Science Master at Castilla La Mancha University. Now, he is developing multi-agent systems for control and production, applying solutions in differents environments like airports, factories, etc. while obtaining the Phd degree. E-mail: pablo.garcia@uclm.es

MIGUEL POYATOS was born in Cuenca, Spain. He received the Telecommunications Technical Engineer degree from the University of Castilla-La Mancha (UCLM) in 2008. He works in Autolog group since 2007, where he developed his interests in J2EE applications orientated to analysis of resource allocation for incoming flight and RFID luggage tracking for Central Airport at Ciudad Real. His e-mail address is: miguel.poyatos@uclm.es and the Web-page of the Autolog association is: http://autolog.uclm.es.

JOSE M. PASTOR received the Electronics and Automatic Control Engineering degree from the Polytechnic University of Madrid (UPM) in 1991 and a Ph.D. in Robotics and Artificial Intelligence in 1997. In 2005 Dr. Pastor moves to the Informatics Systems Department of the Castilla La Mancha University where he is currently Associate Professor. His research interests include intelligent manufacturing systems, and systems simulation and optimization. His e-mail address is: josemanuel.pastor@uclm.es

ANDRÉS GARCÍA HIGUERA received degrees in Industrial Engineering for Automation and Electronics at the Polytechnical University of Madrid and in Mechanical Engineering for the University of Córdoba (Spain). He obtained his PhD in Robotics and Automation at the Polytechnical University of Madrid. He then joined the Automation and Control Group, Institute for Manufacturing, Department of Engineering, University of Cambridge as Senior Research Associate. In 2003 Andrés qualified as Senior Lecturer in Spain (Profesor Titular de Universidad) and went back to Castilla La Mancha Univ. Nowadays he supervises several nationally funded research projects with strong industrial participation and is director of the AutoLog research group. His e-mail address is: andres.garcia@uclm.es

AN ALTERNATIVE TO DEAD RECKONING FOR MODEL STATE QUANTISATION WHEN MIGRATING TO A QUANTISED DISCRETE EVENT ARCHITECTURE

Arno Duvenhage and Bernardt Duvenhage
The Council for Scientific and Industrial Research
Pretoria
South-Africa
Email: aduvenhage@csir.co.za, bduvenhage@csir.co.za

KEYWORDS

Quantised discrete event, state quantisation, state estimation, dead reckoning, pure pursuit, alpha-beta filter

ABSTRACT

Some progress has recently been made on migrating an existing distributed parallel discrete time simulator to a quantised discrete event architecture. The migration is done to increase the scale of the real-time simulations supported by the simulator. This however requires that the existing discrete time models be modified to work within the quantised discrete event environment. To this end the use of model state quantiser and quantised integrator pairs are required. An alternative to dead reckoning is suggested for the quantized integrator algorithm: a state estimation algorithm that has successfully been used to inject live aircraft into a discrete time simulator.

INTRODUCTION

The South African National Defence Force (SANDF) currently requires a system of systems simulation capability for supporting the different phases of a Ground Based Air Defence System (GBADS) acquisition program as discussed by Baird and Nel(Baird and Nel, 2005) and others(Oosthuizen, 2005)(Naidoo and Nel, 2006). A non-distributed, fast-as-possible discrete time *synthetic environment* simulator was developed by the Council for Scientific and Industrial Research (CSIR) in support of the simulation capability during the *concept and definition* phases of the acquisition life cycle as detailed by le Roux(le Roux, 2006). The acquisition life cycle is part of the system life cycle shown in Figure 1. Real-time simulation execution has, however, now become a prioritised requirement to support the development phase of the acquisition life cycle.

In support of the real-time requirement a distributed parallel simulator was implemented. The logical discrete time management and modelling approach was kept without modification to economically reuse the existing models. Duvenhage and Kourie (Duvenhage and Kourie, 2007) have since shown that the distributed discrete time architecture adequately supports the currently required parallelisation speed-up, but that a parallelisation speed-up ceiling exists just beyond the current use. Progress has
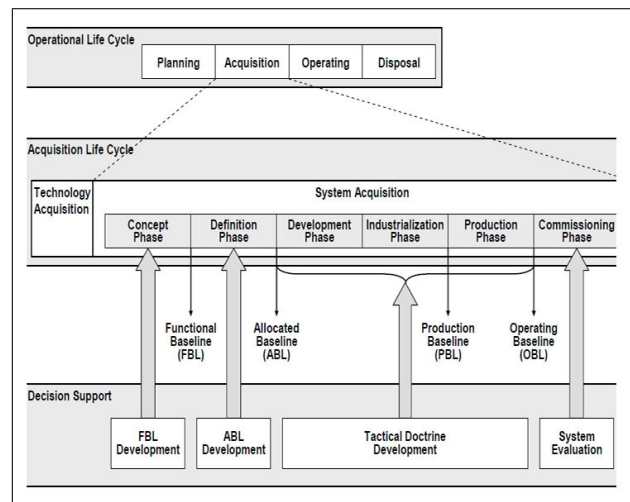
Figure 1: The System Life Cycle(Naidoo and Nel, 2006)

been made on the migration of the discrete time simulator to a quantised discrete event architecture as discussed by Duvenhage(Duvenhage, 2008). It is well known from the literature, (Zeigler et al., 2000) among others, that a discrete event approach to modelling a dynamical system is more efficient than a discrete time approach, both in terms of the communication bandwidth and the model execution time. According to Duvenhage(Duvenhage, 2008), selective application of quantiser and quantised integrator pairs (QQIPs) have shown promise in creating quantised DEVS envelopes around the existing discrete time models that significantly improve upon the simulator scalability.

Live aircraft injection refers to modelling an aircraft through state estimation in real-time when only plot and track data from sensors are available through the relevant command and control (C2) links. This has successfully been used to engage live aircraft with simulated air defence batteries as detailed by Duvenhage (Duvenhage, 2007) within the SANDF GBADS environment. Aircraft injection can also be used to facilitate interoperability or collaboration between different C2 systems and simulators.

This article will briefly discuss the proposed quantised discrete event approach. The focus of the article is however the use of the model state estimator, as described in
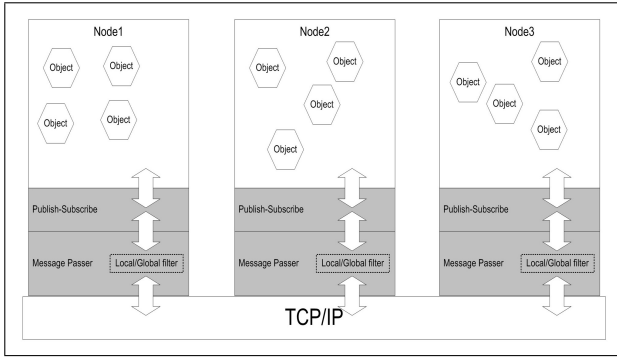
Figure 2: The Layered Discrete Time Architecture.(Duvenhage, 2008)

Figure 3: The Discrete Time Architecture's Parallelisation Speed-Up Against Number of Processing Nodes.(Duvenhage, 2008)

Figure 4: DEVS Envelopes are Wrapped Around Groups of Discrete Time Models.(Duvenhage, 2008)

(Duvenhage, 2007), as a quantised integrator of a QQIP.

MIGRATING TO A QUANTISED DISCRETE EVENT SIMULATOR ARCHITECTURE

The distributed parallel discrete time architecture is a peer-to-peer message passing architecture discussed by Duvenhage and le Roux(Duvenhage and le Roux, 2007) with a publish-subscribe simulation model layered on top of it as shown in Figure 2. Many of the existing models have evolved from high fidelity engineering models often based on numerical method solutions. This resulted in a modelling dependence on a high 100Hz discrete time simulation frame rate and logical time management. In case of real-time execution the simulation execution is throttled to not exceed the wall clock time, but nothing can be done if the simulation consistently runs slower than real-time.

Duvenhage and Kourie (Duvenhage and Kourie, 2007) have analysed the performance of the discrete time architecture. It was found that the architecture has a *real-time* parallelisation speed-up ceiling of around 4 while its current use often requires a parallelisation speed-up of 3.4. The parallelisation speed-up (also referred to as computational load) performance of the discrete time architecture is shown in Figure 3. The performance of an ideal distributed simulator is also shown. An ideal distributed simulator's distribution overhead has no impact on the performance.

Notice that in the case of the discrete time simulator, its maximum parallelisation speed-up is reached on 8 processing nodes. Adding more processing nodes to the simulator would not increase the overall performance, but actually decrease it due to the added distribution overhead. The authors feel that the architecture's current usage is too close to the speed-up ceiling and a quantised discrete event simulator is proposed by Duvenhage(Duvenhage, 2008) as a future simulator migration step.

The proposed discrete event architecture aggregates the existing discrete time models into groups and then wraps the groups within quantised discrete event envelopes. This concept as applied within the GBAD system of systems model is shown in Figure 4.
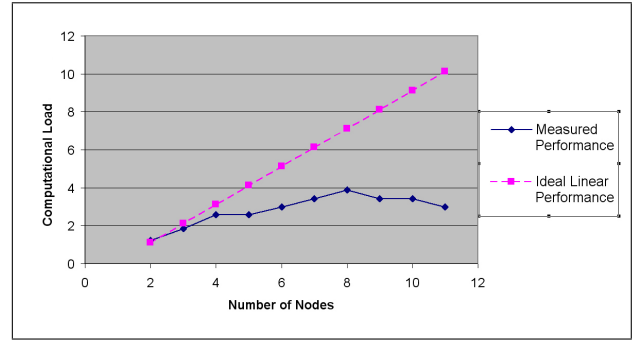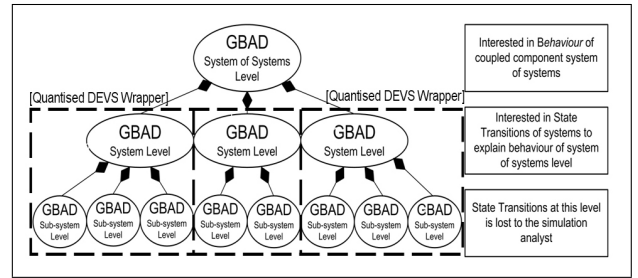
The parallelisation speed-up behaviour, shown in Figure 5, of an experimental implementation of the proposed architecture resembles that of an ideal distributed simulator. This is due to the fact that the architecture is now processor limited and not communication bandwidth limited as was the case in the discrete time simulator. It should be noted that the experimental implementation enveloped the existing discrete time models and the same bench mark scenarios were used for both simulator architectures.

Being bandwidth limited, as in the case of the discrete time simulator, implies that the simulator's performance may be improved by upgrading the communication infrastructure. This is unfortunately not always feasible. On the other hand, being processor limited, implies that the available distributed processing power of the architecture is well utilised. It also has the additional advantage that the simulator's scalability may continually be improved by adding processing nodes or upgrading the existing processing nodes.

MODEL STATE QUANTISATION

State quantization can be divided into two parts as shown in Figure 6. The dead reckoning (DR) algorithm can be used to do model state quantization. This section will however discuss the use of the state estimation algorithm described in (Duvenhage, 2007) as an alternative to dead reckoning for quantized state integration. An alternative
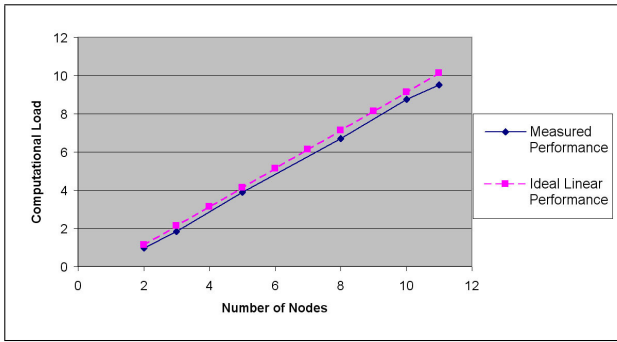
Figure 5: The Discrete Event Architecture's Parallelisation Speed-Up Against Number of Processing Nodes.(Duvenhage, 2008)
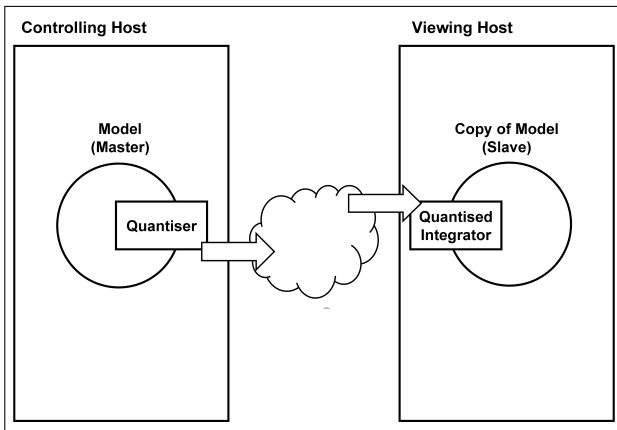
Figure 6: State quantiser and quantised integrator deployment

state quantiser is used in this case.

Dead Reckoning

Dead reckoning is a technique used by simulation technologies like the Distributed Interactive Simulation (DIS) and High Level Architecture (HLA) to reduce the frequency and consequently the bandwidth requirements of distributing state updates and deals with network issues. This is done by predicting a model's state on all hosts until the model's controlling host deems a state update necessary. The model's controlling host will send out a state update if the prediction error becomes too large. These error correcting updates will correct the model state and cause a state update jump on all hosts viewing the model. Larger errors are tolerated if bandwidth requirements are more strict. DR can be setup with several prediction algorithms depending on the model's expected behaviour (Duvenhage, 2007).

The dead reckoning algorithm can be used to do both the quantization and integration of the model state. The dead reckoning state update *jump* in the reconstructed or integrated model state can also be smoothed by using time stamped updates and *phasing in* position updates. It is important to note that the quantiser and integrator should be setup with the same prediction algorithm (Fu-
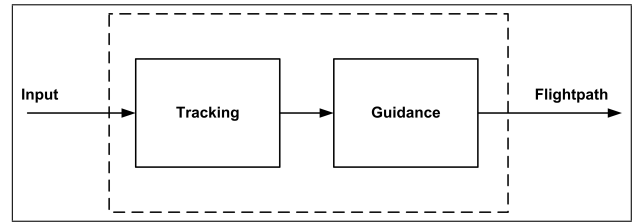
Figure 7: The Tracking and Guidance Algorithm (Duvenhage, 2007)

jimoto, 2000).

The Aircraft Injection Algorithm

The aircraft injection (state estimation) algorithm shown in Figure 7 includes both state estimation (tracking) and guidance.

The basic principle is that the guidance component guides the simulated aircraft from its current position on an ad hoc basis (in real-time) to position predictions calculated by the tracking component. The position predictions are made for the next expected update time each time a new state update is received.

The tracking component is responsible for:

- Estimating the position of the aircraft from the state updates,

- estimating the time of the next state update,

- compensating for temporal variance and lag in state updates, and

- mitigating possible lag caused by the guidance algorithm by being able to predict guidance input into the future.

The tracking component can be a kinematic tracking filter like an $\alpha - \beta$ or $\alpha - \beta - \gamma$ type tracking filter: These are 2nd and 3rd order kinematic filters that are easy to implement compared to the more complex Kalman, particle and Interacting Multiple Model (IMM) type tracking filters that may also be used (Blackman, 1986).

The guidance component is responsible for:

- Guiding the aircraft to the predicted aircraft positions on an ad hoc basis (in real-time as new predictions are made),

- calculating the aircraft orientation if required, and

- limiting aircraft movement to stay within the dynamics of the relevant aircraft type.

Guidance laws that can be used for the guidance component are pure pursuit and proportional navigation. Pure pursuit guidance can also be called a tail chase algorithm since the pursuer always tries for a trajectory directly toward the target. Proportional navigation on the other hand over steers in an attempt to keep the line of sight angle between the pursuer and target constant. The line
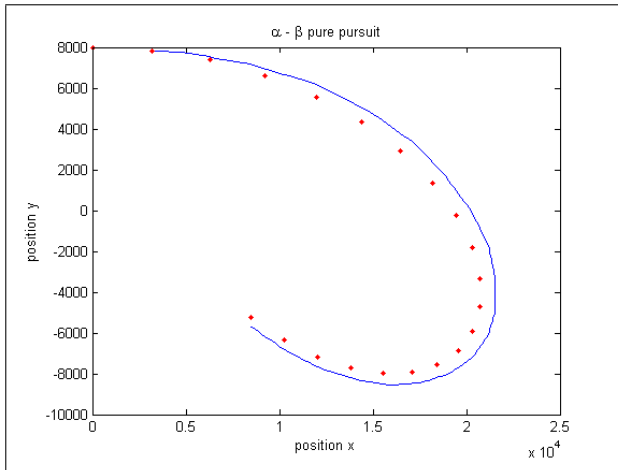
Figure 8: Target (dotted line) and output of alpha-beta-pure-pursuit algorithm (solid line) (Duvenhage, 2007)

of sight angle can be defined as the angle between *the line of sight from the pursuer to the target* and the pursuer trajectory (Duvenhage, 2007).

Alpha-beta-pure-pursuit Quantized Integrator

The aircraft injection algorithm can be called an *alpha-beta-pure-pursuit* algorithm since it uses an $\alpha - \beta$ tracker and a pure pursuit guidance algorithm (Duvenhage, 2007). Figure 8 shows the output of this algorithm for a specific set of state updates. The algorithm accepts the state updates in real-time and always outputs an up to date estimation of the model state.

The $\alpha - \beta$ filter is a 2nd order kinematic filter that assumes the model moves at a constant velocity between measurement updates. This is not the case for maneuvering or accelerating models, although the assumption may hold for sections of the model's path. This specific implementation allows for variable update rates and delayed observations (also using time stamped updates) (Duvenhage, 2007).

The pure pursuit guidance moves the pursuer (or model in this case) directly toward its target. Pure pursuit guidance will cause the pursuer to trail the target and intercept it from behind. The model's orientation can be calculated based on the model's acceleration and velocity (Duvenhage, 2007).

Alpha-beta-pure-pursuit Quantiser

The $\alpha - \beta$ filter requires updates to be able to stabilize for the integrator to reconstruct the model state with the minimum amount or error. One option is to simply sample the model state at regular intervals and use that as the quantized states. This will allow the integrator to stabilize as well as minimize the communications bandwidth requirements of the distributed simulation.

CONCLUSION

Progress has recently been made, as mentioned, on migrating an existing distributed parallel discrete time simulator to a quantised discrete event architecture. The migration is investigated to increase the scale of the real-time simulations supported by the simulator. Such a migration is however economically possible only if the existing discrete time models are reused. This implies that the discrete time models be enveloped in quantised discrete event wrappers.

From the discussion it is clear that the aircraft injection algorithm could be used as a quantised integrator algorithm. This offers a good alternative to dead-reckoning with the major advantages being that there already exists a proven implementation of the aircraft injection algorithm within the SANDF GBADS environment. The algorithm also doesn't require matching of prediction algorithms between the different simulation hosts since the tracking algorithm can estimate the model state based on arbitrary updates. One drawback however is that the quantiser will have to send more updates, compared to the active quantization done by DR, for the integrator to stabilise fast enough.

FUTURE WORK

Future work on using more advanced tracking and guidance algorithms will contribute to the success of the proposed simulator migration step towards a quantised discrete event architecture. Included in such a study would be the comparative analysis of the performance and error behaviour of all of these algorithms.

The error behaviour is of particular importance due to the fact that the GBAD System of systems model has become reliant on the 100Hz logical and discrete time management. Future effort is also required in re-validating, firstly, the individual enveloped models (models updated with QQIPs) and, secondly, the composed GBAD system of systems model.

REFERENCES

Baird, J. and Nel, J. J. (2005). The evolution of M&S as part of smart acquisition using the SANDF GBADS programme as an example. In *Proceedings of the 12th European Air Defence Symposium*, volume 3694, pages 173–182.

Blackman, S. (1986). *Multiple Target Tracking with Radar Applications*. Artech House.

Duvenhage, A. (2007). A state estimation approach for live aircraft engagement in a C2 simulation environment. In *Proceedings of the 2007 Fall Simulation Interoperability Workshop*.

Duvenhage, B. (2008). Migrating to a real-time distributed parallel simulator architecture. Master's thesis, University of Pretoria.

Duvenhage, B. and Kourie, D. G. (2007). Migrating to a real-time distributed parallel simulator architecture. In *Proceedings of the 2007 Summer Computer Simulation Conference*, page 46.

Duvenhage, B. and le Roux, W. H. (2007). Peer-to-peer simulation architecture. In *Proceedings of the 2007, High Performance Computing & Simulation (HPCS'07) Conference*, pages 684–690.

Fujimoto, R. M. (2000). *Parallel and Distributed Simulation Systems*. Wiley-Interscience, New York, USA.

le Roux, W. H. (2006). Implementing a low cost distributed architecture for real-time behavioural modelling and simulation. In *Proceedings of the 2006 European Simulation Interoperability Workshop*, pages 81–95.

Naidoo, S. and Nel, J. J. (2006). Modelling and simulation of a ground based air defence system and associated tactical doctrine as part of acquisition support. In *Proceedings of the 2006 Fall Simulation Interoperability Workshop*, pages 551–558.

Oosthuizen, R. (2005). Doctrine development during systems acquisition and the importance of modelling and simulation. In *Proceedings of the 2005 European Air Defence Symposium*.

Zeigler, B. P., Kim, T. G., and Praehofer, H. (2000). *Theory of Modelling and Simulation, second edition*. Academic Press, San Diego, California, USA.

AUTHOR BIOGRAPHIES

ARNO DUVENHAGE is a Researcher for the Council for Scientific and Industrial Research (CSIR), South Africa. He joined the CSIR's Mathematical and Computational Modelling Research Group in January 2005 as a Software Engineer. Arno's current work involves modelling and simulation for acquisition decision support, focusing on air defense, specializing in distributed and networked systems. Arno has a BEng Degree in Computer Engineering from the University of Pretoria, South Africa, and is currently specializing in software engineering. His email address is `aduvenhage@csir.co.za`

BERNARDT DUVENHAGE is a Researcher at the Council for Scientific and Industrial Research (CSIR), South Africa. He joined the CSIR's Mathematical and Computational Modelling research group in 2004 as a Computer Scientist. He has since moved to the CSIR's Optronics Sensor Systems Simulation Research group and is currently involved in the modelling and development of the simulator architecture for a first principles physics based infrared and visual band synthetic scene simulator. Bernardt has a Bachelors of Science degree and Honours degree in Computer Science from the University of Pretoria. He is currently finishing off a Masters degree in Modelling and Simulation at the University of Pretoria's Computer Science Department. His email is `bduvenhage@csir.co.za`.

MODELS AND AN ALGORITHM FOR MULTI-CRITERIA SYNTHESIS OF CONTROL TECHNOLOGIES MANAGING INFORMATION SYSTEMS OF VIRTUAL ENTERPRISES

Dmitry A. Ivanov
Chemnitz University of Technology
Department of Economics and Business Administration
Chair of Production and Industrial Management
D-09107 Chemnitz, Germany
E-Mail: idm@hrz.tu-chemnitz.de

Boris V. Sokolov, Dmitry N. Verzilin, Evgeniy M. Zaychik,
Russian Academy of Science,
Saint Petersburg Institute of Informatics and Automation
39, 14 Linia, VO
St.Petersburg, 199178, Russia
E-mail: sokol@iias.spb.su, verzilin@SV10100.spb.edu, EZaychik@beeline.ru,

KEYWORDS

Optimization, Planning, Dynamic Models, Simulation, Information Systems, Virtual Enterprises.

ABSTRACT

Integrated information systems (IIS) of virtual enterprises (VE) were considered as objects of control. Two interrelated problems were jointly stated. The first problem lied in program management of IIS and ranking its structural states. The second one implied that control functions regulating business processes should be reallocated among elements and subsystems of IIS in a real time mode. A formal multi-criteria description of these problems was made and a combined (simulation-based) algorithm of a solution was worked out.

INTRODUCTION

Virtual enterprises unite independent multi-business partners (real enterprises) within a temporal task-oriented technical-organizational structure through information technologies and telecommunications (Camarihna-Matos et al. 2004, Wang and Norrie 2001, Ivanov 2003). Virtual enterprises are highly adaptive to consumer needs and benefit juridical and physical persons providing them with dynamic use of common resources during remote collaboration within a business project.

A virtual enterprise is a typical example of a modern integrated transport, production and trading network performing intensive structural dynamics. This issue makes the structural synthesis of a VE more complicated. In particular, the structure dynamics have a complicative influence upon the following tasks: partner selection (producers and suppliers of components,

retailers, etc); end products configuring; placement of orders; configuring transport network and information resources (Camarihna-Matos et al. 2004, Wang and Norrie 2001).

An integrated information system is one of the main subsystems of VE. It should be constructed through a real-time configuring (structure-functional synthesis) and interconnection of individual information systems belonging to participants (real enterprises).

It is obvious that IIS run under conditions of structure dynamics same as VE does (Okhtilev et al 2006). Possible variants of structure dynamics involving modern information systems are illustrated in fig. 1. Our previous investigations confirmed that incrementation (stabilization) of IIS potentialities and capacity for work necessitates structures control (including the control of IIS structures reconfiguration). There are many possible variants of structure-dynamics control suitable for IIS. The following variants are the most typical: alteration of IIS functioning means and objectives; controlled motion of CTS elements and subsystems; alteration of the order of IIS tasks; redistribution of functions, tasks, control algorithms and information flows among IIS levels; reconfiguring of degraded structures; flexible use of reduced technologies of IIS control.

As applied to IIS, the structure-dynamics control belongs to the general discipline of structure-functional synthesis and program construction, provided for IIS development (Okhtilev et al 2006, Kalinin and Sokolov 1995, Zvirkun and Akinfiev 1993, Zvirkun et al. 1985, Zimin and Ivanilov 1971).

Here we consider an important problem of structure-dynamics control including interrelated business processes (BP) planning and planning of IIS operations aimed at BP improvements.
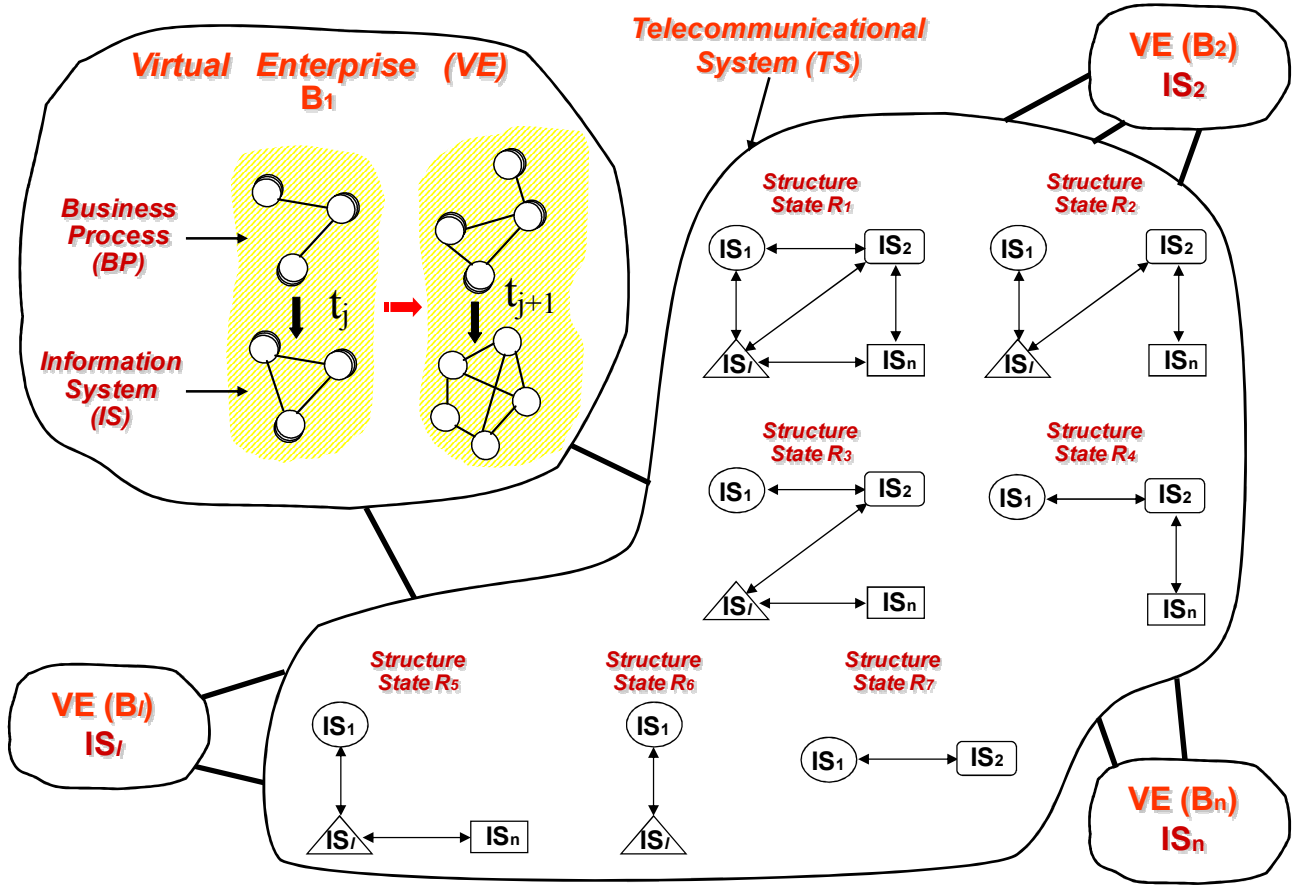
Figures 1: Variants of structure dynamics

PROBLEM STATEMENT

Modern IIS ought to reconfigure and adjust information processes in order to agree with changeable business projects and conditions of their execution.

Efficient functioning of IIS necessitates flexible redistribution of tasks, functions and algorithms among elements, subsystems and levels of the system. Therefore following main tasks have to be redistributed: receiving, transmission and processing of information, planning and control of IIS and VE operation. Moreover, different control technologies involve different variants of IIS structures and different information flows in control loops of IIS.

From a formal point of view a selection (synthesis) of BP structure as well as a real-time structure-functional synthesis of IIS implies joint multi-criteria optimization of VE and IIS operation, selection of control functions for business processes and redistribution of control functions among nodes of IIS. Due to the real-time mode of the IIS, the problem to be solved is more complicated than the ones described in the works of (Zvirkun and Akinfiev 1993, Zvirkun et al. 1985).

The problem of real-time distribution of control functions can be solved at different stages of BP life cycle. In our approach we consider the period of IIS operation planning with simultaneous preliminary distribution of control functions among main elements of IIS and with construction of control programs for these elements. The control programs can be corrected at the stage of real-time control (implementation of the plan). Program corrections can be accompanied by reallocation of resources and by reconfiguration of IIS structures.

Let us introduce some notation for problem definition. Let $A = \{A_i,\ i \in N=\{1,...,n\}\}$ be a set of business processes (and corresponding control functions) to be implemented at some node of IIS at a given time interval $T = [t_0,\ t_f]$. To achieve the VE goals during the interval T, the BPs have to be fulfilled. We distinguish between the functions of goal definition, planning (long term and operational planning), real-time control, VE states analysis, external situation analysis and coordination. The set $A=\{A_i,\ i \in N\}$ is related to sets of informational-technological operations
$$D^{(i)} = \{D^{(i)}_{\text{æ}},\ \text{æ} \in K = \{1,...,s_i\}\},\qquad \text{that are}$$
necessary for implementation of BP $A_i,\ i = 1,...,n$. Let $B=\{B_j, j \in M=\{1,...,m\}\}$ be a set VE main elements and subsystems. Each element B_j can include technical facilities $\quad C^{(j)} = \{C^{(j)}_{\lambda},\ \lambda \in L = \{1,...,l\}\}\quad$ with

appropriate computer equipment and software. Technical facilities are used for implementation of control functions.

Let $E(t) = \|e_{i\,j}(t)\|$ be a known matrix function, with $e_{ij}(t)=1$ in case of the subsystem B_j is carrying out the function A_i at time t in accordance with time-spatial, technical and technological constraints, $e_{ij}(t)=0$ otherwise.

Fig.1 presents an example of seven possible structural states of IIS. The arrows show variants of communication within the system. These variants correspond to different control technologies (methods of IIS application) and different spatial structures of VE.

Now the verbal description of a functions-distribution problem can be presented as follows. It is necessary to select the best variants of functions distribution among the nodes of IIS for each structural state $R_1,R_2,...,R_k$ of IIS (under known time spatial, technical and technological constraints) and to find the best variants of functions implementation. The structural states of IIS should be sorted according to their preference. The preference relation can be expressed through quality functions characterizing efficiency of IIS and its structural and technologic characteristics.

The described problem belongs to the class of multi-criteria choice problems with finite sets of alternatives (structural states of IIS).

ALGORITHM OF MULTI-CRITERIA PLANNING OPERATIONS IN IIS

The general algorithm for the problem includes the following steps.

Step 1. Models (analytical, simulation and combined models) describing structural states $R_1,R_2,...,R_k$ are used for optimal distribution of BP and control functions among subsystems of IIS, for technological operations planning and for evaluation of IIS efficiency. The following characteristics of IIS efficiency can be used: the total number of functions implemented in subsystems during the interval T, the total number of BP in given macro-states, the total number of technological operations executed over the time interval T, the total time of operations over the time period T. The above-mentioned characteristics can have stochastic or fuzzy interpretation if uncertainty factors are present (Okhtilev et al 2006, Orlovski. 1981).

The following dynamic model of functions distribution can be used for evaluation of IIS efficiency (Okhtilev et al 2006, Kalinin and Sokolov 1995, Zimin and Ivanilov 1971).

$$\dot{x}_i^{(\phi)} = \sum_{j=1}^{m} \varepsilon_{i\,j}(t)u_{i\,j}^{(\phi)}; \quad \dot{x}_{i\text{æ}j}^{(0)} = \sum_{\lambda=1}^{l} b_{i\text{æ}j\lambda}u_{i\text{æ}j\lambda}^{(0)};$$

$$\dot{y}_{i\,j}^{(\phi)} = v_{i\,j}^{(\phi)}; \tag{1}$$

$$\sum_{j=1}^{m} u_{i\,j}^{(\phi)}\left[\sum_{\alpha\in\Gamma_{i1}}\left(a_{\alpha}^{(\phi)} - x_{\alpha}^{(\phi)}\right) + \prod_{\beta\in\Gamma_{i2}}\left(a_{\gamma}^{(\phi)} - x_{\gamma}^{(\phi)}\right)\right] = 0; \tag{2}$$

$$\sum_{\lambda=1}^{l} u_{i\text{æ}j\lambda}^{(0)}\left[\sum_{v\in\Gamma_{i\text{æ}1}}\left(a_{i\,v\,j}^{(0)} - x_{i\,v\,j}^{(0)}\right) + \prod_{\mu\in\Gamma_{i\text{æ}2}}\left(a_{i\,\mu\,j}^{(0)} - x_{i\,\mu\,j}^{(0)}\right)\right] = 0; \tag{3}$$

$$\sum_{i=1}^{n} u_{i\,j}^{(\phi)}(t) \le 1; \ \forall j; \ \sum_{j=1}^{m} u_{i\,j}^{(\phi)}(t) \le 1; \ \forall i;$$

$$u_{i\,j}^{(\phi)}(t) \in \{0,1\}; \tag{4}$$

$$\sum_{j=1}^{m}\sum_{\lambda=1}^{l} u_{i\text{æ}j\lambda}^{(0)}(t) \le 1, \ \forall i, \forall \text{æ}; \ \sum_{i=1}^{n}\sum_{\text{æ}=1}^{s} u_{i\text{æ}j\lambda}^{(0)}(t) \le 1,$$

$$\forall i, \forall \text{æ}; \ u_{i\text{æ}j\lambda}^{(0)}(t) \in \{0,u_{i\,j}^{(\phi)}\}; \tag{5}$$

$$v_{i\,j}^{(\phi)}\left(a_{i\,s\,j}^{(0)} - x_{i\,s\,j}^{(0)}\right) = 0; \ v_{i\,j}^{(\phi)}(t) \in \{0,1\}; \tag{6}$$

$$x_i^{(\phi)}(t_0) = x_{i\text{æ}j}^{(0)}(t_0) = y_{i\,j}^{(\phi)}(t_0) = 0; \tag{7}$$

$$x_i^{(\phi)}(t_f) = a_i^{(\phi)}; \ \left(a_{i\text{æ}j}^{(0)} - x_{i\text{æ}j}(t_f)\right)y_{i\,j}^{(\phi)}(t_f) = 0;$$

$$x_i^{(\phi)}(t_f) = a_i^{(\phi)}; \ \left(a_{i\text{æ}j}^{(0)} - x_{i\text{æ}j}(t_f)\right)y_{i\,j}^{(\phi)}(t_f) = 0; \tag{8}$$

$$J_0 = \sum_{i=1}^{n}\sum_{j=1}^{m} v_{i\,j}^{(\phi)}(t_f); \ J_1^{(n)} = \sum_{i=1}^{m} v_{n\,j}^{(\phi)}(t_f);$$

$$J_0 = \sum_{i=1}^{n}\sum_{j=1}^{m} v_{i\,j}^{(\phi)}(t_f); \ J_1^{(n)} = \sum_{i=1}^{m} v_{n\,j}^{(\phi)}(t_f); \tag{9}$$

where $x_i^{(\phi)}(t)$ is equal to total duration of the business process Ai fulfillment in subsystem Bj as $u_{i\,j}^{(\phi)}(t)=1$; the variable $x_{i\text{æ}j}^{(0)}$ express the current state of the technological operation $D_{\text{æ}}^{(i)}$; $y_{i\,j}^{(\phi)}$ is equal to the time passed after A_i completion in B_j until the time $t = t_f$; $a_{\alpha}^{(\phi)}, a_{\alpha}^{(0)}, a_{\gamma}^{(0)}, a_{i\,v\,j}^{(0)}, a_{i\,\mu\,j}^{(0)}$ are given values setting end conditions for $x_i^{(\phi)}(t), x_{\alpha}^{(\phi)}(t), x_{\gamma}^{(\phi)}(t), x_{i\,v\,j}^{(0)}(t), x_{i\,\mu\,j}^{(0)}(t)$ at time $t = t_f$; $u_{i\,j}^{(\phi)}, u_{i\text{æ}j\lambda}^{(0)}, v_{i\,j}^{(\phi)}$ are control inputs. Here $u_{i\,j}^{(\phi)}$ (t)=1 if BP A_i is being executed in the subsystem B_j at time t, $u_{i\,j}^{(\phi)}$ (t)=0 otherwise; $u_{i\text{æ}j\lambda}^{(0)}$ (t)=1 if the technological operation $D_{\text{æ}}^{(i)}$ is executed in the technical facility $C_{\lambda}^{(j)}$, $u_{i\text{æ}j\lambda}^{(0)}$ (t)=0 otherwise; $v_{i\,j}^{(\phi)}$ =1 if BP A_i was implemented in the subsystem B_j, $v_{i\,j}^{(\phi)}$ =0 otherwise. Here the sets Γ_{i1}, Γ_{i2} include the numbers of functions that are direct predecessors of the control function Ai. The set Γ_{i1} indicates predecessors connected by logical "and", the set Γ_{i2} indicates

predecessors connected by logical "or". The sets $\Gamma_{i\ae1}$, $\Gamma_{i\ae2}$ include the numbers of technological operations $D_\nu^{(i)}$ and $D_\mu^{(i)}$ that are direct predecessors of the operation $D_\ae^{(i)}$. The subscripts 1 and 2 express the type of logical connection as stated above.

Therefore, constraints (2) and (3) define allowable sequences of control functions and technological operations. Constraints (4) and (5) specify that each BP at each time can be carried out only in one subsystem B_j ($i=1,...,n$; $j=1,...,m$) and conversely, each subsystem B_j can carry out only one BP A_i at the same time. Similar constraints are used for technological operations $D_\ae^{(i)}$ that are executed at the technical facility $C_\lambda^{(j)}$.

Expression (6) states switching-on conditions for the auxiliary control input $\nu_{ij}^{(\phi)}$ (t). Expressions (7) and (8) specify end conditions for the state variables at the time $t = t_0$, $t = t_f$, R^1 is a set of positive real numbers. The functionals J_0, J_1, J_2 are quality measures for distribution of BP in IIS. Here J_0 is equal to total number of functions by the time $t = t_f$, J_1 is equal to the number of subsystems the function A_i is implemented in, J_2 expresses the elapsed time for implementation of all necessary functions.

A simulation model of real-time control can be used together with expressions (1)-(9) for taking into account uncertainty factors. In this case special procedures of inter-model coordination can be used (Okhtilev et al 2006, Kalinin and Sokolov 1995).

Extreme values of functionals characterizing IIS efficiency can be determined via solution of optimal control problem for finite-dimensional differential system with mixed conditions. The solution algorithms and different aspects of their programming are considered in (Okhtilev et al 2006, Kalinin and Sokolov 1995).

Step 2. Structure-topological characteristics of IIS are being evaluated (Zimin and Ivanilov 1971) including: the coefficient of attainability J_4, different measures of structure compactness (radius J_5 of the structure, diameter J_6 of the structure, integral measure J_7 of structural compactness), measures J_8 of structure centralization (decentralization).

The formulas for computation of measures are proposed in (Zimin and Ivanilov 1971).

Step 3. The pairwise-comparison matrix Kc is completed for measuring the IIS efficiency. Expert appraisal is used for completion of the matrix.

Step 4. The weights of measures (significance coefficients) are evaluated according to the matrix Kc. The algorithm proposed in (Orlovski 1981) is used here. The vector of coefficients is equal to the normalized eigenvector $\vec{\omega}_c$ corresponding to the maximal eigenvalue $Lmax$ of the matrix Kc. Thus the following equation has to be solved:

$$(Kc - Lmax*I)\,\vec{\omega}_c = 0, \qquad (10)$$

where I is a unitary matrix.

Then a weight of each structural state $(R_1, R_2,..., R_k)$ of IIS for each measure taken separately is evaluated. These weights complete the matrix Kr. Each column of the matrix Kr includes relative weights of the states in respect of some measure. A weighted sum of measures is received for each alternative $R_1, R_2,..., R_k$. In other words, total sets of weights are determined for each structural state via the formula:

$$Kr\,\vec{\omega}_c = \vec{\omega}^*. \qquad (11)$$

Step 5. The structural states are sorted according to their preference. The best one is characterized with the maximal element of the vector $\vec{\omega}^*$. Each element of the vector $\vec{\omega}^*$ can be interpreted as a total weight of some structural state.

CONCLUSIONS

Dynamic interpretation of operation planning in IIS let thoroughly describe and investigate interrelation and interaction of business processes and the processes of information processing, storing and interchange.

The framework of integrated multi-criteria operations planning in the context of IIS structure-dynamics control results in the following advantages. The goals of IIS planning can be directly interrelated with the goals of business processes. Structure-dynamics operations (IIS control technology) can be reasonably selected and substantiated. Efficient compromise solutions can be found for allocation of control functions among the elements of IIS and for general programs (plans) of IIS operation. The preliminary ordering of IIS structural states let rapidly reconfigure it in case of failures (Okhtilev et al 2006).

Several prototype versions of software were produced for structure-dynamics control of IIS in different application domains (cosmonautics, power industry, management, etc, see http://www.spiiras-grom.ru). The experiments with software confirmed efficiency of models applied.

REFERENCES

Camarihna-Matos, L., Kluwer (editors) et al 2004. *Virtual Enterprises and Collaborative Networks*, Academic Publishers, 610 p.

Wang L. and Norrie D. H. 2001. "Process Planning and Control in a Holonic Manufacturing Environment." *Journal of Applied Systems Studies*, 2(1), 106–126.

Ivanov, D.A. 2003 *Virtual Enterprises and Logistics Chains: Integrated Approach to Organization and Control in New Forms of Production Cooperation* SPbSUEF, 120 p. (In Russian)

Ohtilev, M.Yu., Sokolov, B.V., Yusupov, R.M. 2006. *Intellectual Technologies for Monitoring and Control of Structure-Dynamics of Complex Technical Objects.* Moscow, Nauka, 410 p. (in Russian)

Kalinin, V.N. and Sokolov, B.V. 1995. "Multiple-Model Approach to Description of Control Processes in Space Systems." *Control Theory and Systems*, No 1, 149-156. (In Russian).

Zvirkun, A.D. and Akinfiev, V.K. 1993. *Structure of the Multi-Level Systems (Synthesis and Development).* Moscow, Nauka. (in Russian).

Zvirkun, A.D. and Akinfiev, V.K., Filippov, V.A. 1985. *Simulation Modeling in the Problems of Complex Systems Structure Synthesis.* Moscow, Nauka, 172 p. (in Russian).

Zimin, I.N. and Ivanilov Yu.,P. 1971. "Solving of Network Planning Problems via a Reduction to Optimal Control Problems." *Journal of Calculus Mathematics and Mathematical Physics.* 11, No 3, 632-641 (in Russian).

Ore, O. *Theory of Graphs.* 1962. AMS Colloquium Publications Vol. 38, AMS, Providence, RI, 270 p.

Orlovski, S.A. 1981. *Decision Making under Fuzzy Information.* Moscow, Nauka, 208 p. (in Russian).

AUTHOR BIOGRAPHIES

DMITRY IVANOV is a researcher at the Chemnitz University of Technology and Chair of the German-Russian Coordination Office for Logistic. He studied production management and engineering (2000). In 2002, he graduated in Saint Petersburg as a Ph.D. in Economics on the topic of operative supply chain planning and control in virtual enterprises. In 2006, he received the Dr.rer.pol. degree at the Chemnitz University of Technology. He is an author of six scientific books and more than 70 papers published in international and national journals, books and conference proceedings. Since 2001, he has been involved in research and industry projects on supply chain management and virtual enterprises. Dr. Ivanov received a German Chancellor Scholarship Award in 2005. His e-mail address is: idm@hrz.tu-chemnitz.de.

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 deputy director of St.-Petersburg Institute for Informatics and Automation. His e-mail address is: sokol@iias.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 leading researcher of St.-Petersburg Institute for Informatics and Automation. Professional interests lay in operations research, simulation, and statistical analysis. His e-mail address is verzilin@SV10100.spb.edu.

EUGENIY M. ZAYCHIK was born in 1962 in Rostov-on-Don, Russia. He graduated from Military Academy of Communication in 1992. 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 St.-Petersburg Institute for Informatics and Automation. His e-mail address is EZaychik@beeline.ru.

THE QUAY CRANE DEPLOYMENT PROBLEM AT A MARITIME CONTAINER TERMINAL

Pasquale Legato
Dipartimento di Elettronica, Informatica e Sistemistica
University of Calabria
Via P. Bucci, Cubo 41C, 87036, Rende (CS), Italy
E-mail: legato@deis.unical.it

Daniel Gullì, Roberto Trunfio
Centro di Supercalcolo per l'Ingegneria Computazionale
CESIC – NEC Italia S.r.l.
Via P. Bucci, Cubo 22B, 87036, Rende (CS), Italy
E-mail: {daniel.gulli, roberto.trunfio}@eu.nec.com

KEYWORDS
Quay crane deployment problem, scheduling problem, maritime container terminals, discrete-event simulation, optimization.

ABSTRACT

Container unloading/loading at marine container terminals (MCTs) is a key logistic process, to which some research efforts have been addressed by using mathematical programming models formulated in a deterministic-static environment. Vice versa, DES models in a stochastic-dynamic environment are well capable of representing the entire process. Hence, simulation results to be an effective planning and control tool for decision making at all decisional levels. Here we remark that optimal decisions in MCTs may be practically pursued by modelling the whole MCT and focusing attention on the core logistic processes, while representing in a simplified manner the remainder. We focus on the operational management of the cranes deployed along the quay, during the container unloading/loading process at a given number of vessels according to a previously planned berth-schedule. We suggest a two-phase approach to the quay crane deployment problem: in the first phase an IP model is used to decide when and how many cranes must be assigned to each vessel; afterwards, we propose a heuristics to determine which specific crane should be assigned to a vessel. We indicate how this approach can be successfully integrated in a DES model, already available, to support dynamic assignment of cranes to berthed vessels.

INTRODUCTION

Freight transportation plays a key role in modern economies as it allows goods exchange between far-off countries. The most notable and steady technology for transporting freight, especially on long maritime routes, is *containerization*. A considerable growth in worldwide containerised freight transportation has been registered in recent years – approximately 90 percent of the world's cargo traffic moves by container (UNCTAD 2007).

Competitiveness within the growth can be achieved by enforcing the introduction of decision support systems in the organization and management of core logistic processes involved in transportation. An efficient and effective management of logistic activities in a container terminal can decrease the operating costs and service times and increase the quality of services.

A maritime container terminal is a complex facility organised around a set of logistic processes. The logistic activities at a container terminal often belong to more complex logistic processes. This fact is critical for a good management of the system and the choice of the system modelling approach. A firm classification of the decision problems in a maritime container terminal concerns to the following logistic processes (Vis and De Koster 2003; Steenken et al. 2004): *i) arrival of the ship*, *ii) unloading and loading of the ship*, *iii) transport of containers from ship to stack and vice versa*, *iv) stacking of containers*, and *v) inter-terminal transport and other modes of transportation*.

Several interesting papers focusing on the previous logistic processes have been proposed (Legato and Mazza 2001; Park and Kim 2003; Legato and Monaco 2004; Cordeau et al. 2005; Cordeau et al. 2007; Canonaco et al. 2008). Among these works, only the ones that are based on a simulation modelling approach, to capture the dynamic and non deterministic framework, are able to evaluate large instances in a reasonable time, to conduct scenario analysis and overall performance evaluation. Besides, simulation offers the opportunity of highlighting congestion phenomena occurring at those (shared) resources resulting as bottleneck within a logistic process (e.g. the quay cranes).

In this paper we propose a two-phase approach to the *quay crane deployment problem* (QCDP). The QCDP is a complex scheduling problem that arises when multiple vessels berths to a quay and a limited set of quay cranes must be assigned to the berthed vessels in order to respect vessels committed due-time of departure. The first phase consists of an integer programming model that produces the optimal number of cranes that must be assigned to each berthed vessels on the basis of a one-hour time-slot. In the second phase, this output data are used to assign the cranes to the vessels according to an ad hoc heuristics.

We are currently integrating the proposed approach in a discrete-event simulator to support runtime crane assignment using a deterministic berth schedule. The approach has been validated using the Park and Kim

(op cit) schema. The comparison study showed that our approach minimizes the number of utilized cranes and maximizes the cranes utilization. It also guarantees the vessel overall completion time and the respect of non-crossing constraints for rail-mounted quay cranes.

The remainder of this paper is organized as follows. For sake of completeness, in the next section we briefly describe the logistic processes in which our approach is arranged in. Afterwards we focus our attention to the QCDP and then we define the mathematical model and the heuristics applied within the two-phase approach. At last, we show some computational results to illustrate the potentiality of our approach.

PROBLEM DESCRIPTION: LOGISTIC PROCESSES AND MODEL OBJECTS IN A MARINE CONTAINER TERMINAL

Modelling maritime container terminal is a complex activity. Simulation is the most powerful tool for the study of this complex and large system, but specific algorithm must be used to model complex processes and to solve assignment and scheduling problems: on that way, mathematical models and heuristics helps in the best modelling of the system. In the following, we first identify the set of model objects involved into a container terminal and then we briefly describe the logistic processes that are involved into the QCDP.

A marine container terminal is a large facility composed by: an harbor entrance wide and water-depth enough followed by an adequate roadstead where incoming vessels may stop and wait; a small number of pilot boats; one or more quay of an adequate length (generally more than 1,000 m) with along a channel of multiple water depth ranging; for each quay, a limited set of rail-mounted gantry cranes (RMGs) and rarely less rubber-tyre gantry cranes; a fleet of straddle carriers (SCs) or AGV shuttle vehicles (AGVs) used to transfer containers between the quay cranes and the yard (SVs are able to handle containers if there are no yard cranes); a set of RMGs for the yard operations; and a large yard surface suitably organized in sectors by a regular layout (yard from large container terminals are able to store nearly thousands twenty equivalent units – better known as TEUs, an ISO standardized containers unit measure).

The logistic processes at a container terminal are described in the following.

When calling at the port, a vessel makes a sort of "advanced reservation" based on its ETA - Expected Time of Arrival. Ship entrance into the port depends on the following requirements: *i) formal conditions* (e.g. contractual agreements between the ship's shipping line and the port of call for the use of port facilities), that means a priority policy for the port entrance queue; *ii) operational settings* (i.e. pilot boat availability, berth spaces assignment). If requirements are met, the ship is maneuvered into its berth slot by one or two pilot boats; otherwise it must wait in the roadstead. This is the so called arrival of the ship process. Vessels are of two types: mother vessels and feeders. The first one is a large container ship (up to 3,000 TEUs and no more than 14,000 TEUs) that covers transoceanic lines (hub-to-hub connections). Feeders are smaller ships that cover short and middle routes. They are widely used to connect the spokes to the transshipment hub (and vice versa).

Once a vessel is berthed, container discharge/loading can be initiated only if mechanical (and human) resources are allocated; if not, the ship waits in its berth position until resource assignment. Discharge/loading operations are performed by RMGs placed along the berth: one or multiple cranes move containers between the ship and the quay area. The maximum number of quay cranes that may be assigned to each vessel is restricted by *i)* the total number of cranes in the quay and *ii)* the maximum number of allowed cranes for each vessel, due to physical (i.e. the length of the vessel) and logical constraints (i.e. interference between cranes operations). Considering the span of the cranes (approximately 30 m) and the horizontal space necessary to stack and to transfer away the incoming/outgoing containers of a vessel, the maximum number of cranes allowed for the longest vessel is usually 5 (this number is proportionally decreased for shortest vessel). When multiple cranes are assigned to the same ship, crane interference has to be avoided and a complex scheduling problem arises to manage the relationships (precedence and mutual exclusion) existing among the holds of the same ship. Considering that the service rate of the RMGs is *30 TEUs/hour*, the performance of the discharge/loading process highly depends on the availability of this type of cranes and their turnover speed. Therefore, the best deployment of these resources affects the overall completion time of each vessel. Finally, a fleet of vehicles, SCs (or AGVs), take in charge containers and move forth and back between the berth area and the yard area.

The Quay Crane Deployment Problem

In the unloading and loading of the ship process, a key decisional problem is the Quay Crane Deployment Problem (QCDP). This problem has been successfully discussed by Park and Kim (2003) together the berth scheduling problem. The QCDP is described in the following.

Planners of the terminal operating company weekly construct a "berth schedule", which shows the berthing position and time windows for the incoming vessels (solution to the so called *berth scheduling problem*). A time window shows the expected time of berthing and un-berthing for each vessel; time windows are constructed using the ETA (Expected Time of Arrival) and PTD (Promised Time of Departure) of each vessel (a penalty cost is incurred if the departure of a vessel is later than the previously committed PTD). In Figure 1 is shown an example of berth schedule, where the

berth-time and space have been partitioned into 22 x 24 grid squares (24 time-slots of one hour).
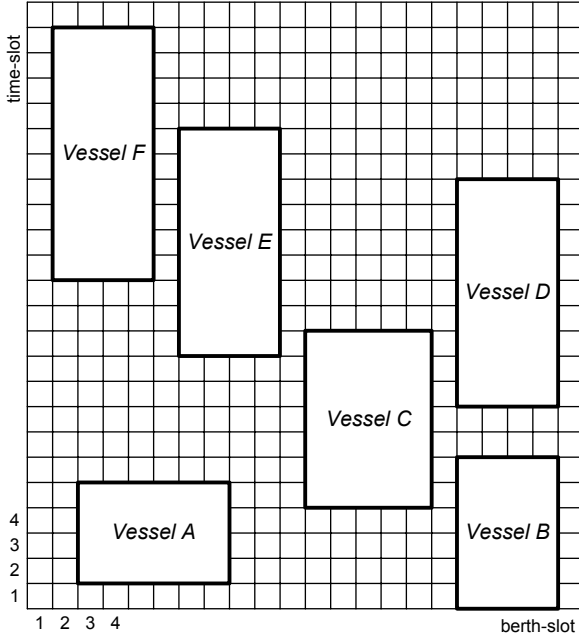
Figure 1: An Example of Berth Schedule Presented by Park and Kim

In the QCDP, the berth schedule is used to assign the RMG cranes to the incoming vessels on a weekly basis. The double goal is *i)* minimize the number of used quay cranes and to *ii)* maximize their utilization, under the constraint of completing the loading/unloading operations, for each vessel, within their expected time of un-berthing,

The QCDP is solved under the following assumptions:

1. Each vessel has a time window; the start time of the time window is the expected time of berthing of the vessel, while its end is the vessel expected time of un-berthing.

2. Each vessel has a total number of TEUs to be handled in its time window: this number is put in correspondence with the moves required by the vessel

3. Each vessel has a maximum and a minimum number of cranes that can and must be assigned when the operations starts. The maximum number of cranes that can be simultaneously assigned to a vessel is limited by vessel length. Vice versa, the minimum number of cranes to be assigned is due (usually for mother vessels) to contract terms between terminal operating company and the vessel shipping company. By default, when the operations start on a vessel, until the operations are completed all the time-slots related to the vessel have almost one assigned crane.

4. Quay cranes are of the RMG type, so there are no-crossing constraints that must be guaranteed. Furthermore, cranes are never un-available.

The solution approach we propose here to the QCDP is decomposed in two phases, as shown in Figure 2.
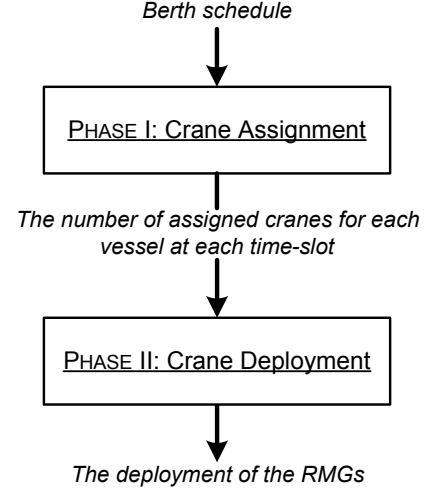
Figure 2: The Schema of the Two-Phase Approach to the QCDP

In the first phase, we solve a mathematical model using CPLEX solver (ILOG 1999) to identify the optimal number of cranes that must be assigned to each vessel for each time-slot. Thus, the model is able to identify exactly when the discharging/loading operations starts and ends within the vessel time-window.

In the second phase, an ad hoc built heuristics is used to assign the cranes to the vessels in order to respect non-crossing constraints and to minimize the number of crane shifts from a vessel to another.

An IP Model for The Crane Assignment Phase

The following notations will be used for the formulation of the IP model used in the crane assignment phase (i.e, the *Quay Crane Assignment Problem – QCAP*):

T	:	The set of time-slots, with $|T| = N$
Ω	:	The set of vessels
C	:	The set of quay cranes, with $|C| = M$
s_c	:	The service rate for the *c-th* crane, expressed in TEUs per time-slot
m_i	:	The number of moves for the *i-th* vessel
etb_i	:	The berthing time for the *i-th* vessel, i.e. the time-slot starting from which the vessel *i-th* is ready, at the berth, for the first lift, where:

$$1 \leq etb_i \leq N - etu_i + 1$$

etu_i	:	The un-berthing time of the *i-th* vessel; thus it is the latest time-slot until

which the vessel *i-th* is available for the operations, where:

$$etb_i \le etu_i \le N$$

min_i : The minimum number of cranes that must be assigned to the *i-th* vessel, when the operation starts.

max_i : The maximum number of cranes that can be assigned to the *i-th*. vessel

We introduce the following decisional variables:

V : The maximum number of cranes used to performs vessels operations.

θ_{it}^c : 1, if crane *c* works on the vessel *i* at the time-slot *t*, 0 otherwise.

ϕ_{it} : 1, if vessel *i* is processed at time-slot *t*, 0 otherwise.

γ_{it} : 1, if operations for the vessel *i* were started at time-slot *t*, 0 otherwise.

η_{it} : 1, if operations for vessel *i* have not been completed at time-slot *t*, 0 otherwise.

The QCAP can be formulated as follows:

$$min \quad N \cdot V + \sum_{i \in \Omega} \sum_{t=etb_i}^{etu_i} \gamma_{it} + \sum_{i \in \Omega} \sum_{t=etb_i}^{etu_i} \eta_{it} \quad (1)$$

s.to

$$\gamma_{it} \le \gamma_{i(t+1)} \qquad \forall i \in \Omega, t = etb_i,\ldots,etu_i \quad (2)$$

$$\eta_{it} \ge \eta_{i(t+1)} \qquad \forall i \in \Omega, t = etb_i,\ldots,etu_i \quad (3)$$

$$\gamma_{it} + \eta_{it} = \phi_{it} + 1 \qquad \forall i \in \Omega, t = etb_i,\ldots,etu_i \quad (4)$$

$$\sum_{c \in C} \sum_{t=etb_i}^{etu_i-1} s_c \theta_{it}^c = m_i \qquad \forall i \in \Omega \quad (5)$$

$$\sum_{c \in C} \sum_{t=1}^{etb_i-1} \theta_{it}^c = 0 \qquad \forall i \in \Omega \quad (6)$$

$$\sum_{c \in C} \sum_{t=etu_i+1}^{n} \theta_{it}^c = 0 \qquad \forall i \in \Omega \quad (7)$$

$$\sum_{i \in \Omega} \theta_{it}^c \le 1 \qquad \forall c \in C, t \in T \quad (8)$$

$$\sum_{c \in C} \theta_{it}^c \le max_i \phi_{it} \qquad \forall i \in \Omega, t = etb_i,\ldots,etu_i \quad (9)$$

$$\sum_{c \in C} \theta_{it}^c \ge min_i \phi_{it} \qquad \forall i \in \Omega, t = etb_i,\ldots,etu_i \quad (10)$$

$$\sum_{c \in C} \sum_{i \in \Omega} \theta_{it}^c \le V \qquad \forall t \in T \quad (11)$$

$$V \ge 0, int \quad (12)$$

$$\theta_{it}^c \in \{0,1\} \qquad \forall i \in \Omega, t \in T, c \in C \quad (13)$$

$$\phi_{it}, \gamma_{it}, \eta_{it} \in \{0,1\} \qquad \forall i \in \Omega, t \in T \quad (14)$$

A Heuristic Approach for the Crane Deployment Phase

As stated before, the solution of the mathematical model provides the number of cranes that must be assigned in order to complete the operations in time. This data are used in the following to assign, for each time-slot, the RMGs.

The step-by-step description of our heuristics is given in the following.

Step 0. *Initialization phase*

Select V contiguous quay cranes from the set C, thus let Q be the set of selected quay cranes ordered by non decreasing position along the quay. Let S be the number of berth-slots and M be the number of the first half of berth-slots $(M = \lceil L/2 \rceil)$; besides, let $from_i$ the berth-slot starting from which the vessel i is berthed and c_{it} be the number of quay cranes assigned to the vessel i at the time-slot t.

Step 1. *Crane assignment phase*
FOR EACH $t \in T$
  Let B be the set of the berthed vessels;
  Compute c_t as the sum of all the c_{it};
  Let $k = |Q| - c_t$ be the number of cranes that will be idle during the time-slot t;
  IF "$B \ne \emptyset$" THEN
    Order B by non-decreasing order of berth position;
    Extract the vessel j from the top of B;
    IF "$from_i < M$" THEN
      WHILE "$B \ne \emptyset$" DO
        i. Extract the vessel i from the top of B;
        ii. IF operations for the vessel i were started at time-slot $t-1$ $(\phi_{i(t-1)} = 1)$ THEN
            IF $k > 0$ THEN
              Extract no more than k cranes from the top of Q if none of this crane was already assigned to the vessel i;
              Update k;
            END IF
        END IF
        iii. Extract c_{it} cranes from the top of Q and assign them to the vessel i;

END WHILE
   ELSE
      WHILE "$B \neq \varnothing$" *DO*
         i. Extract the vessel i from the bottom of B;
         ii. IF operations for the vessel i were started at time-slot $t-1$ $\left(\phi_{i(t-1)} = 1\right)$ THEN
             IF $k > 0$ THEN
                Extract no more than k cranes from the bottom of Q if none of this crane was already assigned to the vessel I
                Update k;
             END IF
         END IF
         iii. Extract c_{it} cranes from the bottom of Q and assign them to the vessel i;
      END WHILE
     END IF
    END IF
   END FOR

A Numerical Experiment

At this point, we compare the results obtained using our approach against the results showed by Park and Kim (2003). In their work, a method for constructing a berth schedule and then to deploy cranes was also developed. Therefore, we compare their results with ours to demonstrate the validity of our method.

In Figure 3 we repeat the optimal assignment reported into the paper by Park and Kim based on the berth schedule previously shown in Figure 1. For each vessel, ringed numbers depict the quay cranes assigned to each vessel. As it is possible to see, the QCDP solved by Park and Kim use 9 cranes to complete all the operations in time.

In this case study, the minimum number of cranes that must be assigned to the vessels during the operations is one, while the maximum number of cranes that is possible to assign to a specific vessel is equal to the number of occupied berth-slots.

The first step of our approach produces the assignment depicted in Figure 4. In this berth schedule, for the *i-th* vessel and the *t-th* time-slot the optimal c_{it} is shown.

As it is easy to recognise, our mathematical model fills a berth schedule minimizing the overall number of cranes that must be used to process all the vessels. In fact, the first phase of our approach produced an assignment of 7 quay cranes against the 9 quay cranes found by Park and Kim. Moreover, in our solution, when the operations start there is no operations

discontinuity for each vessel: this is a primary contractual agreement requested to the terminal operating company by shipping companies as well as the minimum number of assigned cranes and the respect of the committed operations completion time.
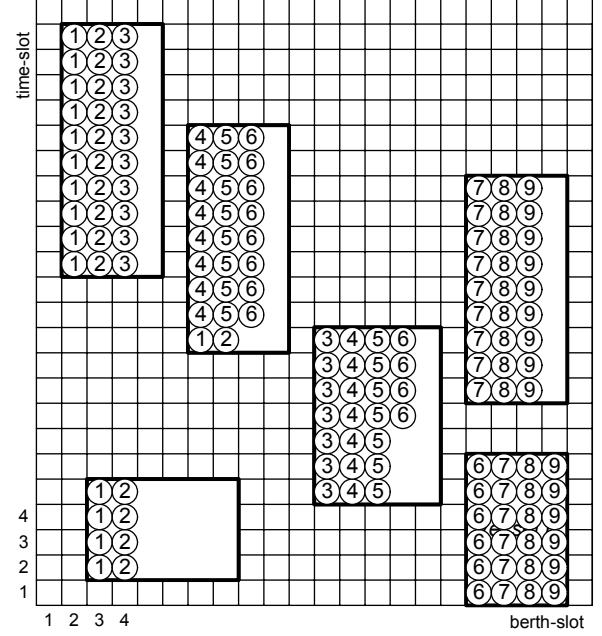
Figure 3: The Optimal Deployment Found by the Park and Kim Methodology
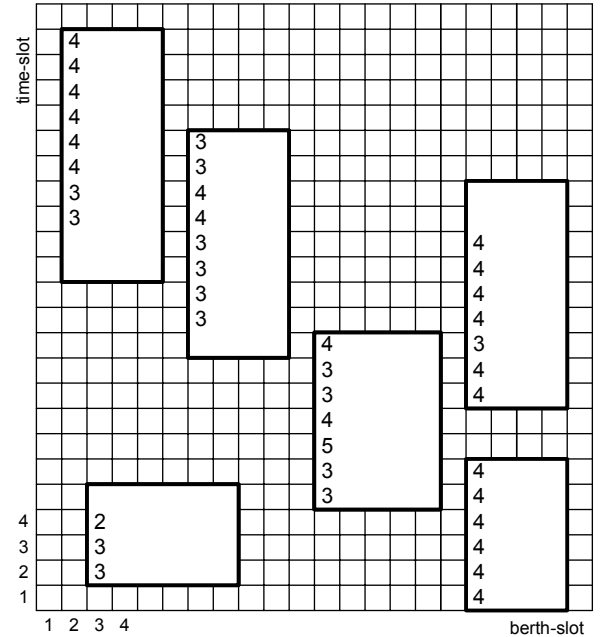
Figure 4: The First Phase Output of our Methodology

As results of the second phase of our algorithm, we propose the quay cranes deployment shown in figure 5.

The improvement obtained with the approach proposed in this paper is not only due to the minimization of the activated cranes. In fact, *i)* in two cases we obtained a reduction of the vessel overall

completion time and *ii)* we improved the average crane utilization (*0.69 vs. 0.87*).
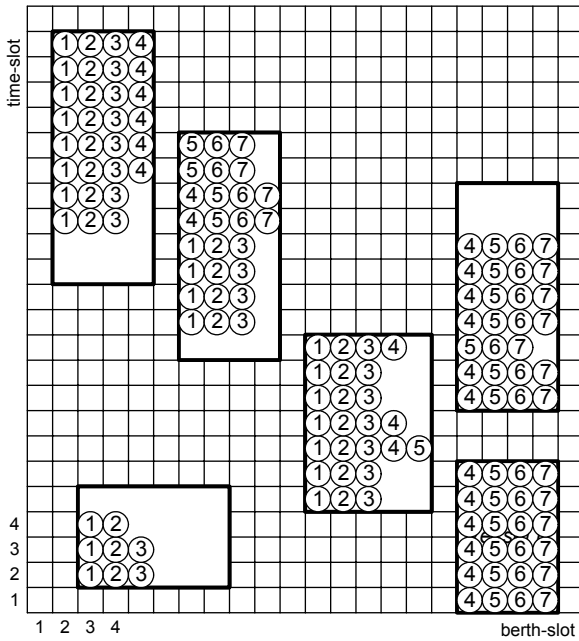
Figure 5: The Final Output by our Methodology

THE NEED FOR SIMULATION INTEGRATION

As stated in the introduction, our belief is that deterministic approaches must be integrated into simulation platforms to conduct scenario analysis through statistical estimates of some system performance measures (Nakayama 2002).

Currently, we are integrating this methodology into a discrete-event simulator for the berth planning and discharge/loading processes at a real maritime container terminal. As a matter of fact, the starting point is the simulator validated and tuned for one of the operating company of the Gioia Tauro terminal, located in the southern Italy (Canonaco et al. 2008). The terminal is the most important transhipment terminal in the Mediterranean Sea and our integrated model will allow the planners of the terminal to identify the best cranes deployment day-per-day. The role of the simulator is central due its capability of verifying the practical possibility of guaranteeing the respect of the planned time windows (berthing–unberting), in a more realistic modelling environment where operation delays and unpredictable events may occur,

As guideline for the integrated simulator, observe that the programmed time-instants of berthing and unberthing, for each vessel, are considered as fixed events within the events list of the simulator. Vice versa, all the operations involved in the discharging-charging process of whatever vessel will be represented with their natural randomness. Particular attention will be devoted to capture the effect of the time required by the process of shifting cranes from one berth slot to another. This time is at the basis of

possible delays upon the programmed time instants of unberthing. Whenever crane shift operations should reveal to be cumbersome and/or time consuming, the terminal manager is asked to evaluate the possibility of increasing the number of available cranes suggested by our two step methodology.

Finally, we are also addressing the problem of integrating a second integer programming model aimed to schedule the work of those cranes assigned to a specific vessel with respect to the stows to be handled. Stow processing requires some precedence constraints and non simultaneous processing of adjacent stows by two different cranes. Thus an adequate crane scheduling policy should be embedded into the simulator, besides the crane deployment one.

CONCLUSION

We have proposed a methodology for the Quay Crane Deployment Problem. The methodology consists of a mathematical programming model and a heuristics. The mathematical model is used to identify the optimal number of quay cranes that must be assigned to the vessel time by time. The heuristics is used to identify which cranes must be assigned to the vessel time by time. We have compared our method with the optimal results obtained by another promising work. The comparison proved that our approach is able to outperform the previous near-optimal solution. Thus, we propose to embed our method into a simulation model to perform scenario analysis at the operational level in order to verify whether or not the time requirements of each vessel can be guaranteed by a suitably fixed number of quay cranes. Finally, we are developing alternative simulation based approaches to supply at the restriction implied by integer programming. Simulation-based Optimization looks as the most promising practical technique to support decisions for the Quay Crane Deployment Problem.

REFERENCES

Canonaco, P.; P. Legato; R.M. Mazza; and R. Musmanno. 2008. "A Queuing Network Model for Management of Berth Crane Operations". *Computers & Operations Research* 35, 8, 2432-2446.

Cordeau, J.-F.; M. Gaudioso; G. Laporte; and L. Moccia. 2007. "The Service Allocation Problem at the Gioia Tauro Maritime Terminal", *European Journal of Operational Research* 176, 1167-1184.

Cordeau, J.F.; G. Laporte; P. Legato; and L. Moccia. 2005. "Models and Tabu Search Heuristics for the Berth Allocation Problem". *Transportation Science* 39, 4, 526-538.

ILOG. 1999. CPLEX 6.5. Mountain View, CA.

Legato, P. and R.M. Mazza. 2001. "Berth Planning and Resources Optimisation at a Container Terminal via Discrete Event Simulation". *European Journal of Operational Research* 133, 537-547.

Legato, P. and M.F. Monaco. 2004. "Human resources management at a marine container terminal". *European Journal of Operational Research* 156(3), 769-181.

Legato, P. and R. Trunfio, 2007. "A Simulation Modelling Paradigm for the Optimal Management of Logistics in Container Terminals". In *Proceedings of the 21th European Conference on Modelling and Simulation.* (Prague, Czech Republic). 479-488.

Nakayama, M.K. 2002. "Simulation Output Analysis". In *Proceedings of the 2002 Winter Simulation Conference.* (San Diego, CA, Dec. 8-11). 23-34.

Park, Y.-M. and K.H. Kim. 2003. "A Scheduling Method for Berth and Quay Cranes". *OR Spectrum* 25, 1-23.

Pidd, M. and R.B. Castro. 1998. "Hierarchical Modular Modelling in Discrete Simulation". In *Proceedings of the 1998 Winter Simulation Conference.* (Washington, DC, USA). 383-389.

Steenken, D.; S. Voss; and R. Stahlbock. 2004. "Container Terminal Operation and Operations Research - a Classification and Literature Review". *OR Spectrum* 26, 3-49.

Vis, I.F.A. and R. De Koster. 2003. "Transshipment of Containers at a Container Terminal: an Overview". *European Journal of Operational Research.* 147(1), 1-16.

UNCTAD. 2007. United Nations Conference on Trade and Development. Review of Maritime Transport 2007. (United Nations, New York and Geneva).

AUTHOR BIOGRAPHIES

PASQUALE LEGATO is an Assistant Professor of Operations Research at the Faculty of Engineering (University of Calabria), where he teaches courses on simulation for system performance evaluation. He has published on queuing network models for job shop and logistic systems, as well as on integer programming models. He has been involved in several national and international applied research projects and is serving as reviewer for some international journals. His current research activities are on the development and analysis of queuing network models for logistic systems, discrete-event simulation and the integration of simulation output analysis techniques with combinatorial optimization algorithms for real life applications in Transportation and Logistics.

His home-page is http://www.deis.unical.it/legato.

DANIEL GULLÌ received a Laurea degree (*cum laude*) in Automotive Engineering from the University of Calabria, Rende, Italy, in 2005. He is devoted to research in numerical simulation at NEC Italy Center for High-Performance Computing and Computational Engineering (CESIC). His current research interests include discrete-event simulation models for logistic systems and parallel simulation.

ROBERTO TRUNFIO received a Laurea degree (*cum laude*) in Management Engineering from the University of Calabria, Rende, Italy, in 2005, and is currently pursuing the Ph.D. degree in Operations Research from the same university. Moreover, he is a logistics engineer at NEC Italy Center for High-Performance Computing and Computational Engineering (CESIC). His current research interests include modelling languages for discrete-event simulation, simulation-based optimisation of logistics systems, and parallel simulation.

MODELLING WITH MAPLE AND MAPLESIM

Jiří Hřebíček
Institute of Biostatistics and Analyses
Masaryk University
Kamenice 126/3, 625 00 Brno, Czech Republic
E-mail: hrebicek@iba.muni.cz

Martin Řezáč
Department of Statistics and Operational Analysis
Mendel University of Agriculture and Forestry
Zemědělská 3, 613 00 Brno, Czech Republic
E-mail: rezac@mendelu.cz

KEYWORDS

Maple, MapleSim, scientific computing, mathematical modelling, computer simulation, Maple rich technical document, Maple document mode, financial modelling, calibration.

ABSTRACT

The paper deals with the system Maple in mathematical modelling and computer simulation of technical or financial problems. It presents also possibilities of Maple for students in learning of mathematical modelling. There are presented some solved problems using Maple and introduced shortly developed modelling and simulation tool - MapleSim.

INTRODUCTION

Mathematical models used in scientific computing [1] are becoming large and the number of computations needed for simulation is increasing. Simultaneously there are growing requirements on model performance and accuracy. When simulation models are designed we encounter several problems that must be solved in order to achieve these goals. The system Maple is appropriate computer environment which enable to solve these goals. The main domains, where is possible to use Maple, are (see Fig. 1):

- o Monte Carlo simulations
- o Data regressions
- o Probabilistic modelling and analysis
- o Economic research and modelling
- o Quantitative pricing and hedging models
- o Calibration
- o Stochastic modelling
- o Optimizations
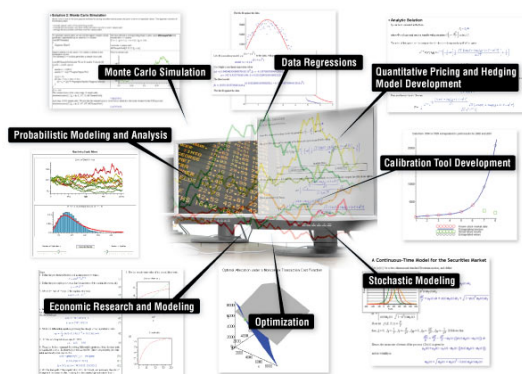
Figure 1: Modelling in system Maple

Firstly, there are presented new possibilities of the Maple computer environment for mathematics, mathematical modelling and simulation in scientific computing. Users can access the power of the Maple computation engine through a variety of user interfaces:

- o *the Standard Worksheet* (contains the full-featured interface),
- o *the Command-Line version* (it does not include graphical user interfaces features, but this method is recommended when solving very large complex problems or using scripts for batch processing),
- o *the Classic Worksheet* (on Windows and UNIX platforms provides a basic worksheet environment for older computers with limited memory),
- o *custom-built Maplet applications* (it allows to create windows, dialogs, and other visual interfaces that interact with a user to provide the power of Maple. Users can perform calculations or plot functions without using the Standard Worksheet interface. A Maplet application is a collection of elements including, but not limited to, windows and their associated layouts, dialogs, and actions, and
- o *the Maplesoft(TM) Graphing Calculator* (is a convenient calculator using Maple technology. It is available as part of Maple installation (Windows(R) only).

Maple provides users with two worksheet interfaces. Both have access to the full mathematical engine of Maple and take advantage of the new functionality in Maple. By default, worksheets open in the enhanced and more modern *Standard Worksheet*. The *Classic Worksheet*, available on Windows and UNIX platforms, has the traditional Maple worksheet look and uses less memory. In Windows, user can change the Maple worksheet file association using the Worksheet File Association Selector application from the Tools folder of the *Maple Start* menu. If computer system has less than the recommended amount of physical memory, it is suggested that users use the *Classic Worksheet* version of Maple. For system requirements, refer to the Maple Web site http://www.maplesoft.com and search on current version of Maple (now Maple 11) and System Requirements.

Every user can create powerful interactive documents with Maple. The Maple environment allows a user to start solving problems right away by entering expressions in *2-D math* and using simple point-and-click interfaces to solve these expressions without having to know any commands. The users can combine text and mathematical expression in the same line, add tables to organize the content of their work, or insert images, sketch regions, and spreadsheets. They can visualize and animate problems in two and three dimensions, format text for academic papers, or books, and insert hyperlinks to other Maple files, Web sites, or email addresses. They can embed and program graphical user interface (GUI) components, as well as devise custom solutions using the Maple programming language.

WORKING WITH MAPLE DOCUMENT MODE

The first Maple session opens with the **Startup** dialog explaining the difference between *Document mode* and *Worksheet mode*. Using either mode, users can create high quality interactive mathematical documents. Each mode offers the same features and functionality, the only difference is the input region of each mode.

- o *Document Mode* - The default input region is a *Document Block*. Document Blocks hide Maple syntax allowing you to focus on the problem instead of the commands used to solve the problem. When using context menus in Document Mode, input and output are connected using an arrow or equal sign with self-documenting text indicating the calculation that had taken place.
- o *Worksheet Mode* - The default input region is a *Maple input prompt* **>**. Using the Maple prompt, all commands are displayed.

Regardless of which mode users are working in, they have the opportunity to show or hide their calculations. Worksheet Mode and Document Mode are identical in the functionality that they both offer; therefore in each mode context menus and palettes are available when entering expressions; tables, spreadsheets, and images can be inserted; and access to the complete Maple library is available. Users can even insert a document block while in Worksheet mode or insert a Maple prompt while in Document Mode. The only difference between the two modes is the default input format.

Table 1. Default input region in worksheet and document mode

Worksheet Mode	Document Mode
Default input region is a Maple input prompt. > *l*	Default input region is a document block. ⊠ *l*

To start a Maple session in Document Mode, it is necessary to select from the **File menu → New** (showing Worksheet mode, document Mode, Templates…) → **Document Mode**. A blank document is displayed, see Fig. 2.
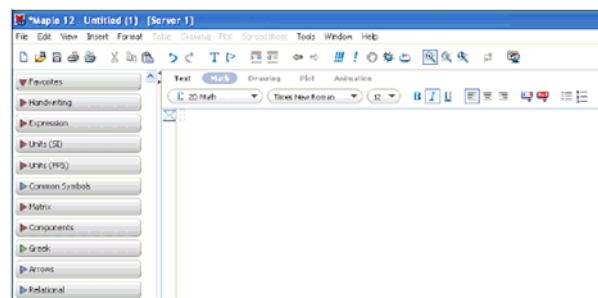
Figure 2: Blank document in Maple Document Mode

In subsequent sessions, is described how to us Maple Document Mode. Maple starts in Document mode. Every time you open a document, Maple displays a Quick Help pop-up list of important shortcut keys. To invoke Quick Help at any time, press the F1 key.

In order to see a Document Block region, you need to display the Markers bar. The Markers bar is a vertical bar along the left pane, see Fig. 1. To display **Markers**, select **View→ Markers**.

Maple users can even insert a document block in Document Mode, toggling between Text and Math modes switches between entering text and entering 2-D mathematical expressions. It is used the **F5** key to switch between Math and Text modes. Math mode is characterized by a slanted, italic prompt (/) whereas the Text mode is characterized by a regular prompt (|).

In Document mode, any Maple statement to evaluate must be entered in Math mode. In Document mode is used keys **Ctrl**+= to evaluate given mathematical expression or Maple command.

In subsequent sessions, are described menu of *Assistants, Tutors, Tasks* and *Pallets*, which improve efficiency using Maple environment and simplify user work with using Maple.

Assistants

Maple offers a set of *Assistants* in the form of graphical user interfaces for easily interacting with Maple without the need of syntax of Maple. Using the menu, user can access by selecting **Tools→Assistants** to help you accomplish these various tasks, see Fig. 3. In some cases, users can launch the assistants by entering an expression and selecting the assistants from the context menu that displays.
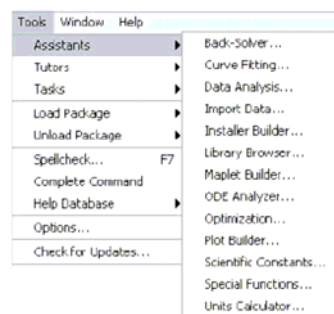
Figure 3: Assistants selection

Tutors

Over 40 interactive tutors that help beginner users gain insight and understanding of topics in *precalculus, calculus, multivariate calculus, vector calculus*, and *linear algebra* courses. Some tutors help users to work through a problem step-by-step. These tutors are easily accessible by selecting **Tools → Tutors→** a subject of your choice → any tutor of your choice, see Fig. 4.
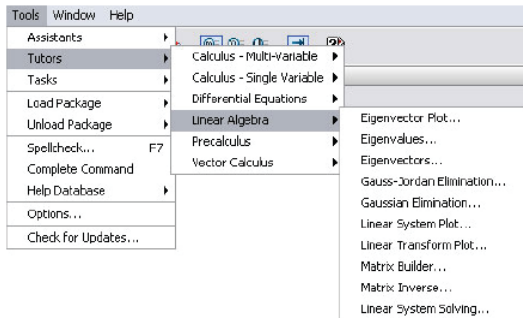
Figure 4: Tutors selection

Figures 5-7 show some examples of these tutors, concretely for *Cross Sections*, *Curve Analysis* and *Approximate Integration*.
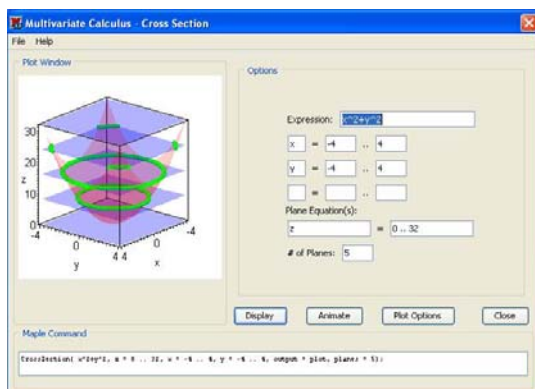
Figure 5: Tutor for Cross Section*s*
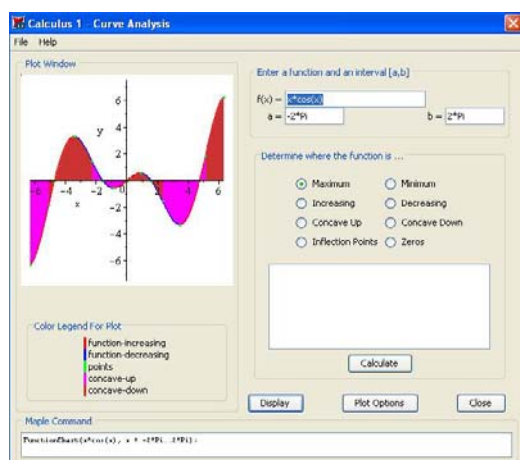
Figure 6: Tutor for Curve Analysis
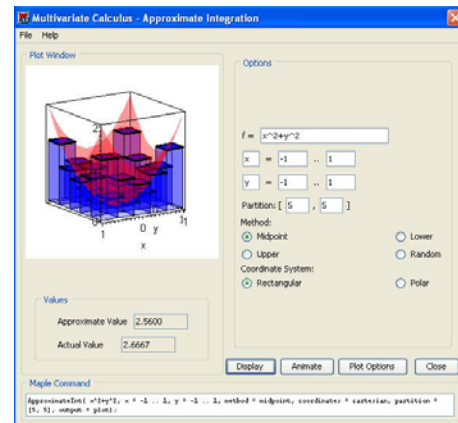
Figure 7: Tutor for Approximate Integration

Context menus

A *context-sensitive menu* is a dynamically generated Maple menu of actions that are applicable for the region upon which it is invoked. Context menus allow users to perform calculations and manipulations on expressions without using Maple syntax. *To display a context menu, users right-click (Control-click for) an object, expression, or region,* see Fig. 8.

Context menus are available for many input regions of document including:

- o *expressions*, to perform calculations, manipulations, or plotting,
- o *plot regions*, to apply plot options and manipulate the plot,
- o *tables*, to modify the table properties,
- o *palette regions*, to add or remove palettes and palette regions ,
- o *text regions*, to add annotations and format text,
- o *spreadsheets*, to manipulate the spreadsheet.
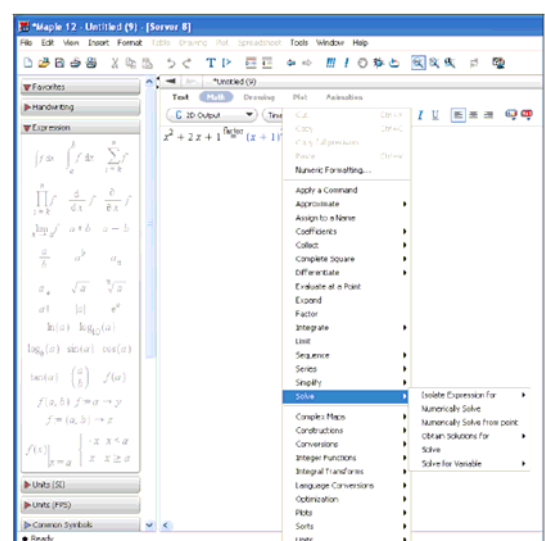
Figure 8: Context menu in Maple 12 Document Mode

When performing calculations or manipulations on an expression, an arrow or equal sign connects the input and output, with text above the arrow or equal sign, indicating the action took place.

New feature in Maple 12 in context menu is shown in Fig. 9 and 10. It is called *Explore* and allows to users to explore any Maple expression – here is explored 3D-plot of function $\sin(x^n + y^m)$, where *n,m* are parameters which could be set up by build-in sliders.
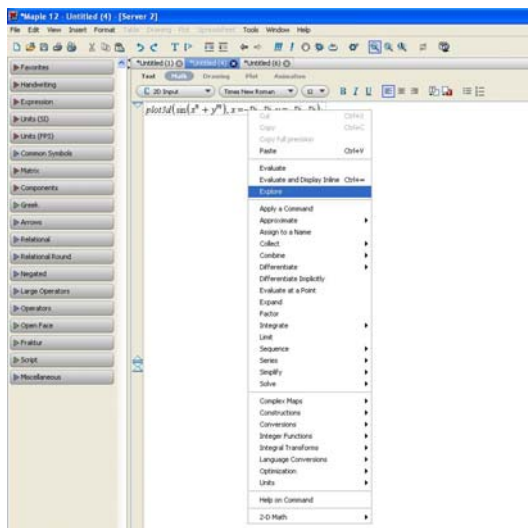
Figure 9: Explore - context menu in Maple 12
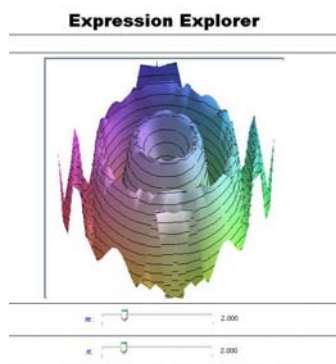
Figure 10: Exploring 3D-plot of $\sin(x^n + y^m)$

Tasks

Maple help users to perform *specific tasks* in Maple, such as:
- o *Performing a mathematical computation*; for example, solving an equation symbolically or numerically, or determining the Taylor approximation of a function of one variable,
- o *Constructing a Maple object*; for example, a function,
- o *Creating a document*; for example, an application.

The *tasks* in Maple environment are organized in the Maple help system by subject, to help users find the appropriate task quickly. Each task is a collection of

content, such as 2-D mathematics, commands, embedded components (for example, buttons) and plots that users can directly insert into their document. They specify the parameters of their problem, and then run the document.
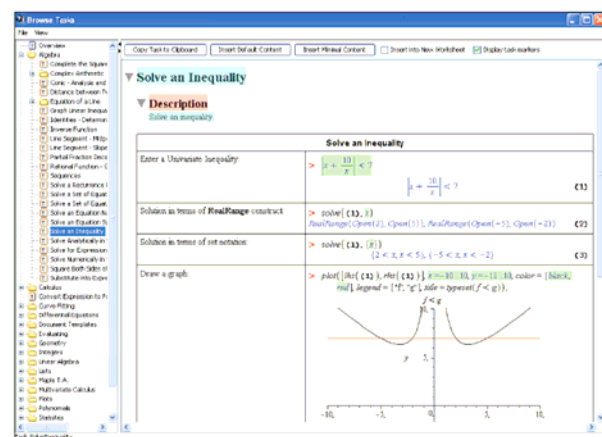
Figure 11: Task "Solve inequality" selection

The Maple document environment provides access to over 20 palettes. Palettes are collections of related items that users can insert by clicking or drag-and-dropping into a document.

Commands of Maple

Maple is powerful, flexible, and customizable. In addition to the point-and-click features, Maple contains a set of commands and a programming language. Commands are contained in the Maple library, which is divided into two groups: the *main library* and *packages*.
- o The *main library* contains the most frequently used Maple commands.
- o *Packages* contain related commands for performing tasks from disciplines such as Student Calculus, Statistics, or General Relativity Theory. For example, the *Optimization Package*

(Fig. 12) contains commands for numerically solving optimization problems.
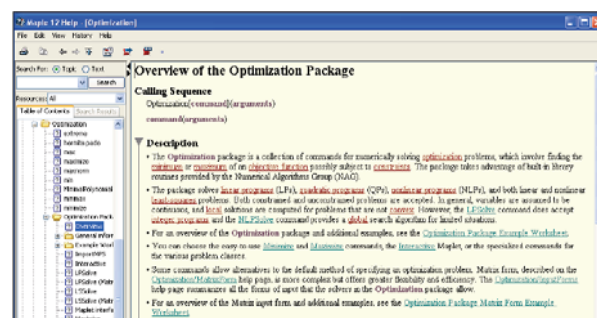
Figure 12: Help page of Optimization Package

If users are familiar with Maple commands or they want to interact with Maple using commands, simply enter

the command using 2-D math at an input prompt. The command names must display in italics. Maple commands are constructed in a format similar to *command (arguments)* based on the command they are using, see Fig. 13.
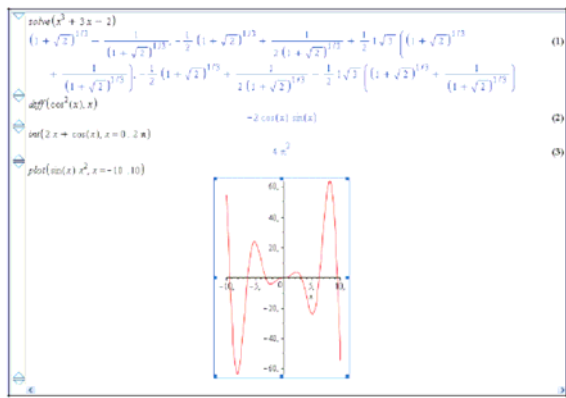
Figure 13: Entering commands

MAPLE TOOLBOXES

Maple Professional Toolboxes greatly extend the scope and functionality of Maple for specialized application areas. Designed to work seamlessly with the core Maple product suite, these high-performance products deploy best of breed technology to deliver maximum benefit from the world's most powerful technical software platform. Toolboxes target key application areas in engineering and science, and technical application development. The list of toolboxes is following:

- o Maple Financial Modelling Toolbox
- o Global Optimization Toolbox for Maple
- o Maple NAG Connector
- o Database Integration Toolbox for Maple
- o Maple Toolbox for Matlab
- o Grid Computing Toolbox for Maple
- o Maple Professional Math Toolbox for LabVIEW

Financial modelling

The Maple Financial Modelling Toolbox complements Maple's multi-disciplinary environment with over 100 new commands, designed specifically for quantitative financial modelling and analysis. These can be combined with the existing Maple tools - including ODE and PDE solvers, statistical data analysis and optimization, to produce analytical applications and product prototypes in the Maple interactive document interface. Common application fields for financial modelling are:

- o Risk analysis
- o Portfolio management
- o Quantitative analysis
- o Model validation/vetting

Two examples are presented on next figures. First figure, it is part of Black-Scholes model (with local volatility) for some market data, see Fig. 14.

Figure 14: Example of financial model (Black-Scholes)

Second figure (Fig 15) shows a model of stock prices with estimation of their distribution in fixed time point.
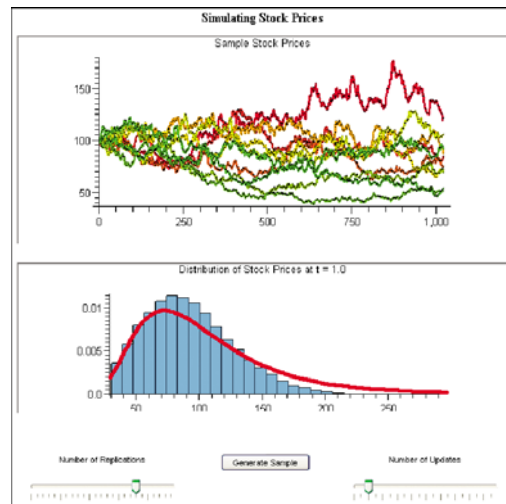
Figure 15: Example of financial model of stock prices

THE FUTURE OF MAPLE MODELLING ENVIRONMENT - MAPLESIM

There is introducing a new product that Maplesoft is working on called MapleSim, This product will address many problems that engineers are struggling with when using traditional mathematical modelling and simulation tools. By using extensive math solvers and simplification technologies, MapleSim will be able to produce highly concise models for efficient numeric computation. This will deliver much faster simulations than can be produced with traditional simulation software that needs to run a numeric iteration at each time step. MapleSim simulations will run faster because the equations for the overall model are generated once and then solved using a very efficient numeric DAE solver.

MapleSim is fully integrated with Maple so users can switch between the two environments, depending on the task they need to perform. Users can use MapleSim to

define and simulate the model, and then use Maple to visualize and validate the mathematical representation. Since the model in this form is fully parameterized, you can use it to perform detailed analysis of the system such as optimization, sensitivity analysis, validation, linearization and simplification, and approximations. MapleSim key features will be, see Fig. 16:

- o *Click, drag, collapse, expand and connect* operations that support hierarchical top-down and bottom-up development to maximize productivity.
- o Seamlessly mix mechanical with, for example, electrical, thermal, hydraulic and chemical components in one environment.
- o Provides both *causal* and *acausal* modelling as well as math and code entry in a combined single environment to reduce your software costs.
- o Unique symbolic technology delivers system equations for efficient numeric computation and advanced analysis.
- o "As-fast-as-possible" one-shot execution and real-time simulation for interactive use.
- o Support for Modelica libraries, and includes tools for creating your own component blocks.
- o Remove conversion errors.
- o Export to real-time platforms for HIL testing applications.
- o Attach live design documents to your system models to minimize rework.
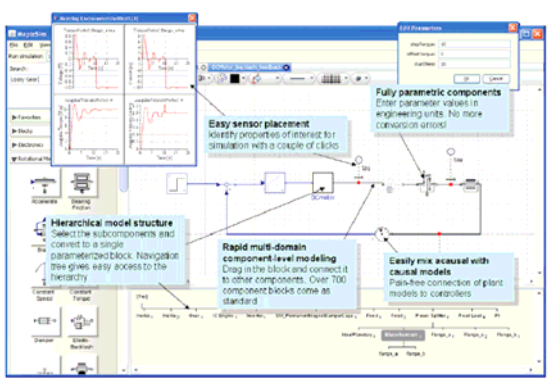
Figure 16: MapleSim GUI environment

CALIBRATION MODELS

Whenever a need to work with inexact measurements of graphical objects occurs, the calibration can be used. It is applicable for example in medicine (for location of cancer genes), in fotogrammetry (measurements of object from airplanes) and in many other fields.

Let us assume two random measurement X, Y of one object. Then calibration problem is aiming to find a function f so, that $Y=f(X)$. The function f is called calibration function. We assume a linear form of function f, i.e. $Y=a+BX$. Matrices a and B are called parameters of calibration problem. For more details see (Moll and Myšková 2007).

Free calibration

We do not assume any further constraint. The only assumption for the calibration model is linearity of calibration function. The model is given by

$$
\begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_n \\ Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{pmatrix} \approx N\left(\begin{pmatrix} M_1 \\ M_2 \\ \vdots \\ M_n \\ a+BM_1 \\ a+BM_2 \\ \vdots \\ a+BM_n \end{pmatrix}, \begin{pmatrix} \Sigma_X & O_k & \dots & O_k & O_k & O_k & \dots & O_k \\ O_k & \Sigma_X & \dots & O_k & O_k & O_k & \dots & O_k \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & & \vdots \\ O_k & O_k & \dots & \Sigma_X & O_k & O_k & \dots & O_k \\ O_k & O_k & \dots & O_k & \Sigma_Y & O_k & \dots & O_k \\ O_k & O_k & \dots & O_k & O_k & \Sigma_Y & \dots & O_k \\ \vdots & \vdots & & \vdots & \vdots & \vdots & \ddots & \vdots \\ O_k & O_k & \dots & O_k & O_k & O_k & \dots & \Sigma_Y \end{pmatrix} \right)
$$

The estimations of parameters of the model are given by

$$
\begin{aligned}
vec\,\dot{M} &= vec\,X + (P_{(1_n, M'_0)} \otimes \Sigma_X B'_0 C^{-1})(vec\,Y - vec\,B_0 X), \\
vec\,\dot{N} &= vec\,Y - (P_{(1_n, M'_0)} \otimes \Sigma_Y C^{-1})(vec\,Y - vec\,B_0 X), \\
\begin{pmatrix} \dot{a} \\ vec\,\dot{B} \end{pmatrix} &= \left(\begin{pmatrix} n & 1'_n M'_0 \\ M_0 1_n & M_0 M'_0 \end{pmatrix}^{-1} \begin{pmatrix} 1'_n \\ M_0 \end{pmatrix} \otimes I_3 \right) \left[vec\,Y - vec\,B_0 X \right] + \begin{pmatrix} 0 \\ vec\,B_0 \end{pmatrix}.
\end{aligned}
\tag{1}
$$

where P is operator of projection.

Calibration of identical non-specified objects

A natural request on calibration function is to retain the shape of objects. This leads to request on matrix B to be orthonormal. It is used Gauss-Markov model:

$$
\begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_n \\ Y_1 + B(\frac{\pi}{2}+\beta_0)M_1^0\beta_0 \\ Y_2 + B(\frac{\pi}{2}+\beta_0)M_2^0\beta_0 \\ \vdots \\ Y_n + B(\frac{\pi}{2}+\beta_0)M_n^0\beta_0 \end{pmatrix} \approx N(A \begin{pmatrix} \beta \\ a \\ M_1 \\ M_2 \\ \vdots \\ M_n \end{pmatrix}, \begin{pmatrix} I_n \otimes \Sigma_X & O_{2n} \\ O_{2n} & I_n \otimes \Sigma_Y \end{pmatrix})), \tag{2}
$$

where

$$
A = \begin{pmatrix} O_2 & O_2 & I_2 & O_2 & \dots & O_2 \\ O_2 & O_2 & O_2 & I_2 & \dots & O_2 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ O_2 & O_2 & O_2 & O_2 & \dots & I_2 \\ B(\frac{\pi}{2}+\beta_0)M_1^0 & I_2 & B(\beta_0) & O_2 & \dots & O_2 \\ B(\frac{\pi}{2}+\beta_0)M_2^0 & I_2 & O_2 & B(\beta_0) & \dots & O_2 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ B(\frac{\pi}{2}+\beta_0)M_n^0 & I_2 & O_2 & O_2 & \dots & B(\beta_0) \end{pmatrix},
$$

and matrix $B(\alpha)$ is an orthonormal matrix written in some special form. Explicit expressions, for estimation of parameters of the model, similar to (1), can be found.

Calibration of identical specified objects - circles

In this case, in addition, we assume that measured points lie on circles. Using Gauss-Markov model again, very similar to the model (2), it is possible to find explicit expressions for estimation of parameters of the model. They are similar to (1), but more complex. Further details can be found in (Moll and Myšková 2007).

APPLICATION

Let us consider following two measurements of 2-D objects:

$$X := \begin{bmatrix} 0 & 1.3 & 0 & -1.8 \\ -1 & 0.7 & 1.4 & 0 \end{bmatrix}$$

$$Y := \begin{bmatrix} -1.8 & -1.7 & -3 & -3.5 \\ 1 & 2 & 3 & 2 \end{bmatrix}$$

Appropriate Maple file (in document mode) for the free calibration is shown on Fig. 17.
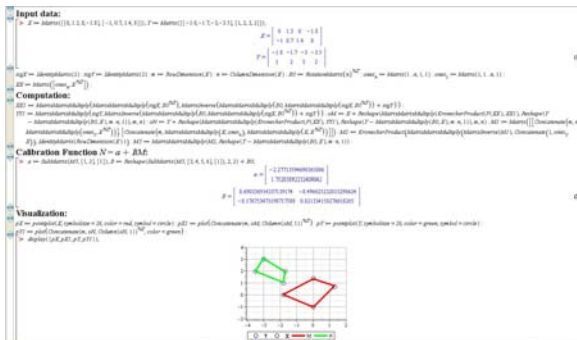
Figure 17: Maple document with free calibration

The whole procedure from setting of inputs to visualization of results is evidently very simple. Similar procedure for the other two types of described calibration models can be done.

Calibration function for free calibration for presented data is given by following matrices.

$$a := \begin{bmatrix} -2.27713594690365006 \\ 1.75203892232409042 \end{bmatrix}$$

$$B := \begin{bmatrix} 0.690336914337139174 & -0.496625232015298624 \\ -0.176753473198717508 & 0.821334158276618265 \end{bmatrix}$$

Visualization of measurements *X* and *Y* and appropriate calibrated values are shown on Fig. 18.
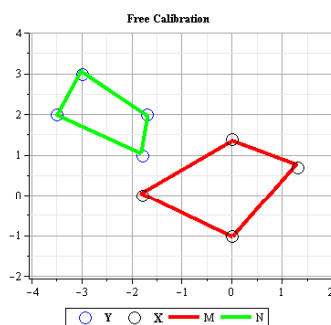
Figure 18: Free calibration

Visualizations of data and appropriate results for calibration of identical objects (non-specified and circles) are shown on Fig. 19 and Fig. 20.
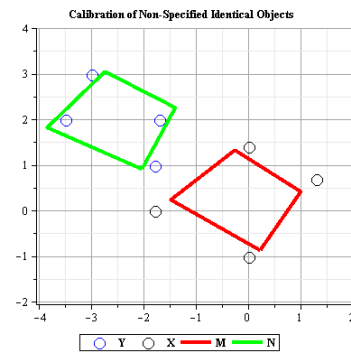
Figure 19: Calibration of identical non-specified objects
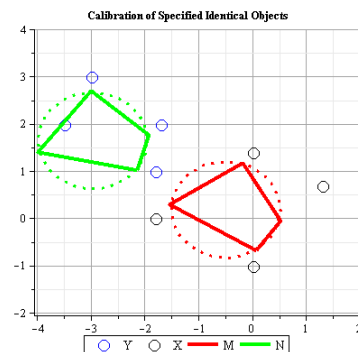
Figure 20: Calibration of identical objects - circles

CONCLUSION

http://www.maplesoft.com/applications/app_center_browse.aspx?CID=4&SCID=55 is Maple Web site resource for free applications related to mathematical modelling and simulations, which obtains 46 applications. There are many interested applications and some of them are available in translation (French, Spanish, and German). The new possibilities offering by the integrated systems Maple and MapleSim in modelling and simulation technical and scientific problems will enlarge the power of Maple technology. A further benefit offers Maple toolboxes, which was presented in case of financial modelling with two short examples. Three calibration models were introduced, including their application using the Maple.

Both Maple and MapleSim will be very useful also for students in learning of mathematical modelling. The MapleSim technology represents the next generation in the modelling and simulations.

REFERENCES

Gander, W. and Hřebíček, J. 2004. *Solving Problems in Scientific Computing Using Maple and Matlab.* Heidelberg Springer, Berlin.

Moll, I. and Myšková, K. 2007. Calibration of identical objects. Summer School, DATASTAT 06, Proceedings, Masaryk University, 195-201.

http://www.maplesoft.com

http://www.maplesoft.cz

COLLABORATIVE SUPPORT SERVICES
FOR CIVIL SERVANTS

Nikolaos V. Karadimas
Vassili Loumos
National Technical University of Athens,
Multimedia Technology Laboratory
9 Heroon Polytechneiou, Zografou Campus,
157 80 Athens, Greece
E-mail: nkaradimas@medialab.ntua.gr,
loumos@cs.ntua.gr

Nikolaos P. Papastamatiou

Omega Technology
4 El. Venizelou, Kallithea
17676 Athens, Greece
E-mail: nikos@omegatech.gr

KEYWORDS

Collaboration Platforms, Web Services, Social Services, Grid Architecture.

ABSTRACT

Civil servants, and especially those engaged in social services, have similar needs all over Europe to access and elaborate information on training and employment offers in order to deliver the best possible service to those who have the right. This work extends the Web Services that collect and offer up-to-date information on training and job offers from different web sources matched to the needs of each case (Karadimas and Papastamatiou 2008) to address the Grid architecture. Grid architecture can be defined as a layer of secure networked services that allow users single sign-on access to a distributed collection of data resources, offering up-to-date information on training and job offers from different web resources. The proposed architecture, as it will include personal data that must be secured from unauthorized access, will be based on the Open Grid Services Architecture (OGSA) that defines standard mechanisms for creating, naming, and discovering transient Grid service instances.

INTRODUCTION

The current work proposes the development of an integrated platform based on the Grid/Web Services architecture that will collect and offer up-to-date information on training and job offers from different web sources matched to the needs of beneficiaries (people of disandantaged citizens' groups searching for work and reinforcement). The system will help civil servants to connect the needs of those who have the right.to employment requests and training opportunities combining different web resources on an individual basis, minimizing at the same time communication expenses, data replication and expiration.

This communication will take place through a Grid/Web service installed in each Social Service Agency that will partially access its database elaborating data about beneficiaries (anonymously) in order to be able to suggest the best possible training and job offers. The web service will map its matching metadata scheme to each Agency's database fields. Several other Grid/Web services will offer information on jobs and training programs directly from the organizations which offer these programs. In this way, the civil servants will only be concerned with their consulting services while information on vocational training and job offers will come directly from the system.

The Grid can be defined as a layer of secure networked services that allow users single sign-on access to a distributed collection of data resources. Each user interested in connecting to a Grid/web service must have an issued certificate and all data transferred through the Grid can be encrypted. This architecture, as it will deal with personal data that must be secured from unauthorized access will be based on the Open Grid Services Architecture (OGSA) that defines standard mechanisms for creating, naming, and discovering transient Grid service instances.

The paper is structured in the following sections: Section 2 presents a theoretical overview of the related literature. Section 3 describes the technology of the proposed service and the proposed solution. Section 4 presents the results while Section 5 is devoted to discussion and future works.

RELATED WORK

Several applications exist that have being developed in the framework of EU funded programs (EQUAL 1 & 2, eTen, etc) or by private initiatives, to support civil servants in their work. The limitation in all these applications is that they are tailor-build to the needs of each agency and the civil servants have to insert in these applications all the information about the programs of training institutions and job offers that they find, in order to match it with their beneficiaries' needs. The proposed solution is not aiming to replace these

applications but to interconnect them in a wider secure knowledge grid offering added value services to all involved.

The innovation of the proposed system is its distributed nature based on the GRID/Web services technology that offers a fast, reliable and effective communication between involved parties, providing at the same time up-to-date information customized each time to the needs of the beneficiaries and vise versa, minimizing at the same time communication expenses, data replication and expiration.

METHODOLOGY

Grid technologies support the sharing and coordinated use of diverse resources in dynamic "virtual organizations" (VOs)—that is, the creation, from geographically and organizationally distributed components, of virtual computing systems that are sufficiently integrated to deliver desired Quality of Service (QoS). The Grid idea emerged initially as a model that combines the resources of many separate computers connected by a network (usually the internet) to solve large-scale computation problems. Web services, as a new, interoperable and effective way to provide functionality over the internet, drew attention to the Grid technology, and especially to Grids build on web services (GRID\WEB services). Following this approach a Grid can be defined as a layer of networked services that allow users single sign-on access to a distributed collection of compute, data and application resources. The above lead to the development of the Open Grid Service Architecture that defines standard mechanisms for creating, naming, and discovering transient Grid service instances, provides location transparency and multiple protocol bindings for service instances and supports integration with underlying native platform facilities. The Open Grid Services Architecture also defines, in terms of Web Services Description Language (WSDL) interfaces and associated conventions, mechanisms required for creating and composing sophisticated distributed systems, including lifetime management, change management, and notification.

This work presents the effort to exploit this innovative architecture so that added value services are offered to the area of civil services. Tools for civil servants (Karadimas and Papastamatiou 2008) are to be developed in a distributed, interoperable and secure infrastructure, namely the Grid infrastructure. Thus the integration of remotely located, disjoint and diverse processing and data storage facilities under a common software architecture will be ensured, and moreover, due to the advanced security mechanism provided by the infrastructure, authentication, authorization and policy management will be facilitated, providing users with the possibility to form Virtual Organizations and share data between them.

The platform on which the Grid/Web services will be built is the open source Globus® Toolkit. Globus®

Toolkit is a fundamental enabling technology for the "Grid," letting people share computing power, databases, and other tools securely online across corporate, institutional, and geographic boundaries without sacrificing local autonomy. The toolkit includes software for security, information infrastructure, resource management, data management, communication, fault detection, and portability. It is packaged as a set of components that will be used for the development of the proposed Grid/web services.

The proposed system is a great opportunity to upgrade the role and efficiency of civil services. It is a cost effective and reliable way to offer accurate and up-to-date information on vocational training courses and job vacancies in the market. It is also a great opportunity for the training organizations to promote their courses and provide training relevant to the real needs of the job-market.

Civil servants have to deal with unemployed people from underprivileged citizen groups who face several problems (lack of skills, disabilities, language problems, family problems,etc). Their efforts focus on, first of all, identifying their problems and then helping them to overcome them. Apart from empowering and supporting individuals to overcome their personal problems, they also try to find for them suitable training programs and jobs.

To be effective, they have to make frequent contacts with training institutions and employment agencies, or directly with employers in order to locate job vacancies and training programs. The problem they face is that they have access to a limited number of resources as many as they can afford to contact or find. This is the reason why their work is focussed more on the empowering part and less in the actually finding of a job for these individuals. It is easier for them to recommend training in one of the training organizations that cooperate with their agencies, and to prepare them for a potential job interview, helping them at the same time to identify what is the appropriate line of work for each individual. Training, however is not directly connected to employment, and most of the courses offered are on basic skills, like communication skills, business English, Information Technology, marketing, etc.

The proposed architecture can solve all those problems that civil servants are facing today by identifing the most reliable and up-to-date sources of information through searching several different on line databases.

The integration of the web services offering access to vocational training courses and job offers is accomplished through the described Grid/Web services architecture. The web services, constituting the GRID, will be custom-made for civil servants needs, performing a specific set of operations that will offer up-to-date information on available Vocational training courses and job offers from different organizations upon request from the civil servants and customized to the beneficiaries' needs.

A Broker (machine that will host the Universal Description, Discovery and Integration (UDDI) directory) is required in order to setup this architecture where all these web services under the GRID will be registered setting up a private UDDI registry.

A web service (W3C Working Group Note 2004) must be register to a UDDI directory in order to be published and reveal its functionality to the world. UDDI is an XML-based (World Wide Web Consortium 2006) standard for describing, publishing, and finding Web services. Service consumers can then require from the broker to locate a required web service and use it to implement a business solution. Each web service, located at the web server of a social agency, will act as one of those service consumers and will use all the web services register to the UDDI registry.

Setting up the "social Grid", specific WSDL (GRID-WSDL) should be developed for each web service to comply to. When each web service is discovered, the GRID-WSDL XML-format protocol will help other systems that need its functionality to understand the way that they can interface it. WSDL describes how to access a web service and not the business logic of this web service.

To utilize the web services functionality a common language between these web services should be defined. XML/SOAP messages will be used for exchanging information. SOAP is a lightweight protocol intended for exchanging structured information in a decentralized, distributed environment (Gudgin et al. 2003).

XML is the technology used with SOAP messages that form the requests and responses from a web service. A client invokes a web service by sending an XML/SOAP message, and then waits for a corresponding XML/SOAP response. Because all communication is in XML, web services are not tied to any one operating system or programming language. The content of these messages will comply with the structure presented in (Karadimas and Papastamatiou 2008) addressing a common approach compliant with the EURES system (The European Job Mobility Portal).

Finally, an application that from a single point will give access to all this information from the registered web services will be developed. The application will enclose a smart search engine that will communicate with the web services subscribed to the UDDI registry and request information according to the beneficiaries needs. The results will be presented to the civil servant along with more information of the organization offering them. All the nodes described on the next diagram communicate with each other through a Transmission Control Protocol/Internet Protocol (TCP/IP).

The advantage of the described architecture is that the pool of available vocational training courses and job offers will grow rapidly as each new organization that desires to join the "social Grid" needs to follow some simple steps in order to develop and register its own web service to the GRID's UDDI. Then the web service will

offer suggestions upon requested information for the organization's courses or job offers.
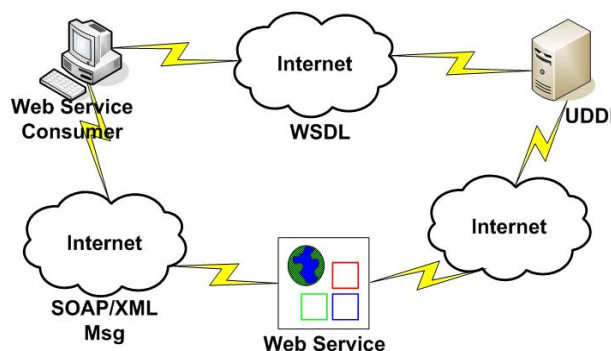
Figure 1: Description of the proposed architecture

Administration costs for the "social Grid" are minimized since the administrator's tasks are limited to observing the successful integration of the new web services to the Grid.

RESULTS

The proposed solution provides civil servants with a decision support tool with access to a tank of educational programs and job offers otherwise difficult to find and filter.

The Grid/Web services solution for social services is a great opportunity to upgrade the role and efficiency of civil servants. It is a cost effective and reliable way to offer accurate and up-to-date information on vocational training courses and job offers. It is also a great chance for the training organizations to promote their courses and provide training relevant to the real needs of the job-market. The proposed architecture is easy to implement, open to the integration of further web services of organizations offering vocational training and job offers, and easy to maintain and extend.

DISCUSSION AND FUTURE WORK

The proposed solution and especially its GRID/Web services architecture, under a common metadata scheme can find implementation in many other sectors, like e-health, e-commerce, the tourism sector, etc. Generally, there is a great need for secure networks for the exchange of data stored in heterogeneous sources. Above such platforms, several other applications can be developed that will make use of the data exchanged through these grid/web services. Other fields of application are the inter-exchange of information between large companies with many branches, public authorities, schools, universities, the military, etc.

REFERENCES

EURES: The European Job Mobility Portal. http://europa.eu.int/eures (Accessed on February 10, 2008).

Gudgin M., Hadley M., Mendelsohn N., Moreau J-J. and Frystyk Nielsen H. 2003. "SOAP Version 1.2 Part1:

Messaging Framework", *W3C Recommendation*, http://www.w3.org/TR/soap12-part1 Accessed on February 23, 2008)

Karadimas N.V. and Papastamatiou N.P. 2008. "Inquiring Training and Employment Offers on the Web Using Web Services". *7th World Scientific and Engineering Academy and Society, International Conference on Applied COmputational Science (ACOS '08)*, Hangzhou, China, 6-8 April, [In Press, Corrected proof].

W3C Working Group Note 2004. "Web Services Architecture". http://www.w3.org/TR/2004/NOTE-ws-arch-20040211/. (Accessed on February 10, 2007).

World Wide Web Consortium (W3C). 2006. "eXtensible Markup Language (XML)". http://www.w3.org/xml (Accessed on February 10, 2007).

AUTHOR BIOGRAPHIES

NIKOLAOS V. KARADIMAS was born in Athens, Greece and he graduated from the Techonogical Educational Institute (T.E.I.) of Patra in 1995 with a Bachelor's degree in Electrical Engineering. He then received a Bachelor's degree with Honours in Electronic Engineering and a Masters degree in Computer Science from Glasgow Caledonian University, Scotland in 1997 and 1998, respectively. He also holds a Masters degree in Distributed and Multimedia Information Systems (1999) from Heriot-Watt University, Scotland and a PhD degree (2007) from the National Technical University of Athens. Since 2007, he is a post-doctoral researcher at the Multimedia Technology Laboratory within National Technical University of Athens and lecturer (407/80) at the Hellenic Military Academy, as well. He is also teaching in Technical NCO Academy and in T.E.I. of Piraeus. He is a Chartered Engineer, member of the Greek Chamber of Engineers, member of IEEE and member of IET. His research interests are in the fields of Databases, Optimization Techniques, Geographical Information Systems, Decision Support Systems and Multimedia.

NIKOLAOS P. PAPASTAMATIOU was born in Athens, Greece and he graduated from Aegean University, Samos in 1998 with a Bachelor's degree on Mathematics. He then received a Masters degree in Distributed and Multimedia Information Systems from Heriot-Watt University, Scotland in 1999. From 2002 he is working for a software house as a web analyst/developer. He has followed several seminars on web technologies (XML Web services, Programming with XML in the .Net Framework, ASP.NET) and he has participated to many European and National research projects. Currently he is following a vocational training course on project management from National and Kapodistrian University of Athens. His research interests are in the fields of Web Development, Databases, Web Services, Grid Computing, Natural Language Processing and Multimedia Applications.

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 Recherhe 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.

M-GOVERNMENT SERVICES IN GREECE

Nikolaos V. Karadimas
Katerina Papatzelou
National Technical University of Athens
Multimedia Technology Laboratory
9 Heroon Polytechneiou, Zografou Campus,
157 80 Athens, Greece
E-mail: nkaradimas@medialab.ntua.gr,
rina310@gmail.com

Agisilaos N. Papantoniou

Senior Consultant
EXIS I.T.
98, Vas. Sofias Ave., Athens, 11528,
Greece
E-mail: apap@exis.com.gr

KEYWORDS

m-Government, e-Government, m-Government Services

ABSTRACT

With the advances in Information Communication Technology (ICT) and the demand for more efficient and effective government, governments in various countries have implemented e-government strategies to offer more information and online services to citizens, government agencies, businesses, and others. E-government initiatives aim to benefit from the use of most innovative forms of information technologies, mainly web-based Internet applications, in improving governments' fundamental functions. It is considered the preferred tool to enhance government services among its customers and government agencies. However, an explosion in the use of wireless internet devices, including internet-ready mobile phones and Personal Digital Assistants (PDAs) in combination with the increasing mobility of people in industrial economies is forcing governments to prepare themselves to transition from e-government to m-government. This paper examines the development of m-government in Greece, which is in an early stage in comparison with other countries all over the world. More specifically, m-government services that are currently used are mentioned. In addition, recommendations are given for the further development of m-government in Greece as well as the benefits to citizens from using various m-government services are elaborated taking into account the initiatives of other countries on the application of m-government.

INTRODUCTION

Technology has created unprecedented changes in the delivery of government services. Along with these changes arise opportunities for improving government services. Improvements in communications and technology give citizens the ability to access government services easily, rapidly and engages people in ways that bolster democracy and citizen involvement. E-government refers to the use of ICTs to transform government operations so that government services are provided electronically to the citizens. Nevertheless, the last years a widespread use of mobile telephony is observed (with over 3 billion mobile users worldwide and over 70% of world's population covered by mobile networks). This fact has contributed to the development of mobile government. The emerging trend in government service delivery through the use of wireless technologies is called mobile government or m-government. Government services are expected to become more accessible to citizens thanks to the wireless technology. Mobiles can reach areas where there are infrastructure constraints for internet service or where wired phone service is not a feasible option. Moreover, the lower cost of mobile telephony enables the expansion of mobile government services to the poorer parts of population in developing countries.

In general, it is argued that m-government's main goals are the resolution of real problems and the improvement of the way that people live. M-government has a positive effect on the economy through infrastructure development, better business practices and enhancements in public sector.

Mobile government strategy seems to be the inevitable path that should be followed by the governments in all countries. Currently, many countries from all over the world have already adopted m-government strategies by implementing various m-government services. Such paradigms are indicatively mentioned in the next paragraph.

Estonia is in the forefront of the world with its e-government and m-government initiatives. This achievement of Estonia is attributed to the willingness of the Estonian government to quickly adopt new technologies (Rannu 2003). Regarding other remarkable m-government initiatives and projects, interesting developments have taken place in various countries such as Germany, Austria, the UK, Ireland, Italy, Malta, Estonia, Sweden, Norway, Denmark, Finland, Korea, Japan, New Zealand Dubai, South Africa and some of the US states such as Idaho and California to name just a few examples.

The major key success factors of the aforementioned initiatives have been political will, cooperation between mobile operators, government, regulators and business entities and dynamic planning.

The present paper is structured in the following sections: Section 2 presents a theoretical overview of the related literature regarding electronic and mobile government deployment in Greece. Section 3 mentions the transition of e-government to m-government services in Greece while in Section 4 m-government services that are expected to contribute to the welfare of the citizens are suggested for the future. Section 5 is devoted to future works.

RELATED WORK

By studying the existing literature, we concluded that research for the field of mobile government development in Greece is in its infancy. Research for the Greek case study is mainly focused on e-government. Hahamis et al. (2005) examined the development and implementation of e-government in Greece both at national and local level. More specifically, the authors concluded that while the progress in the development of e-governement at national level is significant, the results are not so encouraging at local level. Recommendations are made for the improvement of e-government applications at local level. Markellos et al. (2007) emphasized the necessity of efficient e-government applications for the improvement of the public sector quality. They report the best Greek e-government practices and they compare the progress of Greece with other countries of European Union (EU).

Kapogiannis et al. (2006) highlighted the cooperation between e-business and m-business as a crucial factor for the development of m-government applications in Greece. In addition, the paper presents an interesting research on the current needs of the Greek companies to invest in m-business.

CURRENT E-GOVERNMENT AND M-GOVERNMENT SERVICES IN GREECE

In this section, a brief summary of the basic e-government and m-government services that are currently provided by the Greek public sector is presented.

There has been a degree of success in the development of e-Government at a national level. TAXISnet (electronic tax services); the National Printing House; IKAnet (provision of information and transaction services by the Greek Social Insurance Institution); and Syzefxis (the national public administration network) are some of the examples of progress so far (Greek Government, 2002). The Syzefxis programme is a project of the Greek Ministry of the Interior, Public Administration and Decentralization that aims to develop "an effective public administration with a modern information and telecommunications infrastructure and the easier coordination of state processes through Information Technology (IT) and Tele networking" (Informatics Development Agency, 2004).

Looking more specifically at local government, the Ministry of the Interior, Public Administration and Decentralisation implemented an Operational Program (OP) called Politeia which "is the main element of a co-ordinated effort to promote reform of the structure and activity of public administration, with the primary aim to improve services offered to the public" (OECD, 2004).

As far as m-government is concerned, the deployment of m-government services is in an early stage as already mentioned in previous sections. Nevertheless, a beginning has been made. For example - the Mobile-Taxis service, which informs ratepayers regarding the settlement of their tax declaration of income. The development of this service was triggerred by the fact that there is a huge response from citizens on the telephone line that deals with tax matters, and on the corresponding e-government services. Those who are interested in filling their tax declaration by using the mobile service can send an SMS in the M-taxis service, including the number of their tax record, and sending it to 2252. The cost is 0.5 euro. This cost includes all the costs of the SMS. From this registration the citizen ensures that he will be informed regarding tax obligations, and this will happen in real time and in a very easy way - by receiving SMS messages from the M-taxis service. The service is supported by all the mobile carriers operating in Greece. This service is a new attempt by the Ministry of Economy and Public Management to improve its communication with Greek citizens.

Another service that is offered in all European Union countries is a Location Based Service (LBS) which is called 112 emergency call positioning. This service shows on emergency call center worker's computer a map with the location of the person who is making an emergency call to the number 112, which is common to all EU countries. The main benefit of the system is an ability of the emergency service to react quicker in case of emergencies. EU had made compulsory for all operators to be able to offer location-detection of emergency calls by July 2003. It is worth highlighting that the majority of Greek citizens are not aware of this service.

SUGGESTIONS FOR M-GOVERNMENT SERVICES IN GREECE

In this section, various m-government services which could be implemented in Greece as well as their benefits to the welfare of the citizens are discussed. The services that are presented in this paper are selected according to the needs of Greek citizens taking into consideration the social, economic and cultural conditions of the country. In addition, the m-government services that are mentioned in this section were successfully implemented in various countries all over the world. Therefore, the proposed m-government services are expected to be used by the vast majority of the Greek citizens.

Two possible m-government services whose acceptance was satisfactory in other countries include mobile parking and mobile transport ticketing. Mobile parking comprises an alternative, more convenient payment method for parking in big cities. The main benefits of mobile parking to the citizens are convenience since getting out of the car is not necessary anymore and accuracy of payments. The conventional parking ticket systems ask from the customer to determine the parking time in advance, and as a result the amount of time a person pays for parking is usually either bigger or smaller than the amount of time he actually parks. On the other hand, with mobile parking, the time parked and the time paid for is exactly the same. Regarding mobile transport ticketing, commuters will have an alternative option to buy bus, tram, metro and trolley-bus tickets through their mobile phone, offering more convenience and reducing the amount of paper-based tickets in circulation. SMS tickets for public means of transportation can be ordered by sending a text message and users can be charged through their regular mobile phone bills (Rannu 2003).

Another category of m-government services is the provision of time critical public information to citizens via mobile phones, more specifically via SMS; possibility to request information via mobile from government information databases; traffic information (bus, train delays for example), police information about runaway criminals and stolen cars. In addition, during natural disaster such as earthquakes and floods and in general in case of an emergency, mobile technology can be utilized to send alerts to citizens. Reminders also can be sent to the citizens according to their needs and preferences (for example doctor appointment reminders). The use of text-messaging reminders is expected to reduce the number of missed hospital appointments.

It is worthy of note that there are interesting m-government services in the field of education. One application could be sending notifications about child's absence from school notifications to parents' mobiles. Another suggestion is that the delivery of exam results, cancellation of lectures etc. can be performed through SMS-notifications. In this way, students will not have to go to their schools in order to see their results especially in summer-time and the calls to school secretaries are expected to be reduced. This can also be applied to the universities.

One very useful service for our country would be a channel to which everyone can send complaints and suggestions about the poor condition of roads, the possible disorders of public transport system and other issues. The feedback of the citizens about the above issues will be received by the competent authorities who will then take the appropriate measures to maintain and improve the situation. This application will contribute to the participation of citizens to policymaking in a convenient way and to the early detection of problems mentioned by citizens.

Last but not least, mobile voting in local and national elections would enable each citizen to participate in the elections, whatever his or her location is on polling day. This way, the democracy will be enhanced since the participation of the citizens will increase.

From the above descriptions of m-government services, we can derive that there are two kind of services: those which are called "push" type content services in which citizens receive info and "interactive" ones in which there is a "dialogue" between the citizen and the government. Regarding "push" services, the citizens are subscribed to the services of their specific preferences. The "push" services include reminders, alerts in case of an emergency, status info (the status of an application, exam results). Examples of interactive m-government services encompass m-parking, m-ticketing, m-voting.

Table 1: The two types of m-government services

Push services	Interactive services
- Reminders - Alerts (in case of an emergency) - Time critical information - Various notifications	- Mobile parking - Mobile Transport ticketing - Mobile voting - Provision of suggestions - complaints to the authority

FUTURE WORK

Success of m-government requires active engagement by both government and its citizens and so providing services is only one aspect of the m-government equation. Another, and more challenging aspect, is achieving acceptance and widespread persistent use of m-government by citizens. The acceptance of m-government services can be achieved with their proper design and implementation. Services should be delivered in ways already familiar and actively adopted by users. In addition, the mobile phone is perfect device for rapid and brief interaction. Hence, content should be short, targeted and relevant. Another significant parameter that should be taken into account while designing m-government services is the security. The applications that require security must make minimal demands on the user.

Challenges to m-government are considered those related to interoperability (roaming, variety of platforms, etc); usability (mobile devices limitations); privacy protection, etc. To sum up, m-government is inevitable and growing fast. Therefore, the parties involved must adapt to the new reality in order to have responsive public organizations and an "able society" that can benefit.

REFERENCES

Anttiroiko Ari-V. 2005. "Towards Ubiquitous Government: The Case of Finland". *e-Service Journal*.

Caroll J. 2006. "What's in It for Me?: Taking M-Government to the People". *19ᵗʰ Bled eConference, eValues*.

Greek Government 2002. "White Paper: Greece in the Information Society: Strategy and Actions". Athens.

Hahamis P., Iles J. and Healy M. 2005. "e-Government in Greece: Bridging the gap Between Need and Reality", *The Electronic Journal of e-Government*, Volume 3 Issue 4, pp 185-192.

Informatics Development Agency 2004. "Network Syzefxis" http://www.syzefxis.gov.gr/en/project/goals.php [Accessed January 10, 2005]

Kapogiannis G., Touzos M. and Dr Kreps D. 2006. "m-Business & m-Government: Co-operation. The Greek Case Study". In Kushchu I., Broucki C. and Fitzpatrick G. (eds.) *EURO mGOV 2006 - Proceedings of the Second European Conference on Mobile Government*. University of Sussex, Brighton, UK, 30-31 August & 1 September, pp 154-159.

Markellos K., Markellou P., Panayiotaki A. and Stergianeli E. 2007. "Current State of Greek E-Government Initiatives", *Journal of Business Systems, Governance and Ethics*, Vol. 2, No 3, pp 67-88.

OECD 2004. "Country Factsheets". *Paris: OECD/Public Governance and Territorial Development Directorate*, (GOV/PGC(2004)13).

Rannu R. 2003. "Mobile Services in Estonia", *PRAXIS* Working Paper No 8.

Rossel, P Finger, M and Misuraca, G 2006. ""Mobile" e-Government Options: Between Technology-driven and Usercentric". *The Electronic Journal of e-Government* Volume 4 Issue 2, pp 79 – 86.

Summary of Global Dialogue on Mobile Government (November 29, 2007).

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 received a Bachelor's degree with Honours in Electronic Engineering and a Masters degree in Computer Science from Glasgow Caledonian University, Scotland in 1997 and 1998, respectively. He also holds a Masters degree in Distributed and Multimedia Information Systems (1999) from Heriot-Watt University, Scotland and a PhD degree (2007) from the National Technical University of Athens. Since 2007, he is a post-doctoral researcher at the Multimedia Technology Laboratory within National Technical University of Athens and a lecturer (407/80) at the Hellenic Military Academy. He is teaching in Technical NCO Academy and in Technological Educational Institute of Piraeus, as well. He is Chartered Engineer, member of the Greek Chamber of Engineers, member of IEEE and member of IET. His research interests are in the fields of Databases, Optimization Techniques, Geographical Information Systems, Decision Support Systems and Multimedia.

KATERINA PAPATZELOU was born in Larisa, Greece and she graduated from National Technical University of Athens in 2005 with a degree on Electrical & Computer Engineering. She then received a Masters degree in Economics & Administration of Telecommunication Networks in 2007. From 2008 she is working for Vodafone Grece as a Software Engineer. Her research interests are in the fields of Telecommunications, Computing, Natural Language Processing and Multimedia Applications.

AGISILAOS N. PAPANTONIOU was born in Athens, Greece and he graduated from the National Technical University of Athens (NTUA) in 1991 with a Bachelor's degree in Civil Engineering. He also holds a PhD degree (2003) from the National Technical University of Athens in the domain of Organizational Modelling, Knowledge Discovery and Artificial Intelligence. During the last 15 years he has been involved in numerous projects, either as Head of IT or as a consultant, in the areas of OSS/BSS for the Telecommunication sector, Transaction Processing Systems for the Gaming/Lottery sector, Enterprise Resource Planning and Customer Relationship Management. As of 2006 he is Senior Consultant in Exis IT, a Greek based consulting company. He is a member of the Greek Chamber of Engineers. His research interests are in the fields of Knowledge Engineering and Management, Artificial Intelligence, Databases, Decision Support Systems and Multimedia.

Agent-Based Simulation

USING A PROBABILISTIC AGENT TO SUPPORT LEARNING IN SMALL GROUPS

Elisa Boff

Rosa Maria Vicari
Moser S. Fagundes

Computer Science Department
Caxias do Sul University (UCS)
CEP 95.001-970 Caxias do Sul
RS – Brazil
Computer Science Institute
Federal University of Rio Grande do
Sul (UFRGS)
E-mail: eboff@ucs.br

Computer Science Institute
Federal University of Rio Grande do
Sul (UFRGS)
POBox 15.064 – 91.501-970
Porto Alegre – RS – Brazil

E-mail: rosa@inf.ufrgs.br
msfagundes@inf.ufrgs.br

KEYWORDS

Deliberation, Probabilistic Systems, BDI and Social Interaction.

ABSTRACT

Social relationships are highly complex activities that are very difficult to model computationally. In order to represent these relationships, we may consider various aspects of the individual, such as affective state, psychological issues, and cognition. We may also consider social aspects, as how people relate to each other, and to what group they belong. Intelligent Tutoring Systems, Multi-agent Systems and Affective Computing are research areas which our research group have been investigating, in order to improve individual and collaborative learning. This paper focuses on a Social Agent which has been modelled using probabilistic networks and acts in an educational application. Using the Social Agent as a testbed, we present a way to perform the deliberation process in BDI and Bayesian Networks (BN). The assemblage of mental states and Bayesian Networks is done by viewing beliefs as networks, and desires and intentions as particular chance variable states that agents pursue. In this work, we are particularly concerned with the deliberation about which states of affairs the agent will intend. The focus of this paper is on how to build a real application by using the deliberation process developed in one of our previous work.

INTRODUCTION

The research work presented in this paper is the result of the integration of two systems previously developed by our group (AMPLIA and PortEdu). This integration was necessary in order to make AMPLIA, an Intelligent Tutoring System, available on the Web. This new reality brings new challenges, like the need to improve our systems with social facilities to help students that are studying by distance learning.

This paper focuses on the Social Agent's deliberation process in BDI and Bayesian Networks (BN). We are particularly concerned with the deliberation about which states of affairs the agent will intend. The focus of this paper is on how to build a real application by using the deliberation process developed in (Fagundes et al., 2007).

The importance of social interactions in the learning process is already known by educational theoreticians, as in the socio-cultural approach of Vygostky (Vygotsky, 1999), some works of Piaget (1995), theories of collaborative learning (Dillenbourg, 1995), and others. The advances in Intelligent Tutoring Systems (ITS) propose the use of architectures based on agent societies (Giraffa, 1998; Mathoff, 1994; Norman, 2000). The principles of multi-agent systems have shown a very good potential in the development of teaching systems, due to the fact that the nature of teaching-learning problems is easily solved using a multi-agent system.

The design of Collaborative Learning Environments (CLE) may take into account social factors, such as the work presented in Cao et al. (2003). They conclude that it is important to consider sociological aspects of collaboration in order to discover and describe existing relationships among people, existing organizational structures, and incentives for collaborative action. Hence, a learning environment may detect and solve conflicts. In a CLE learners have to be active in order to manipulate objects, to integrate new concepts, to build models and to collaborate with each other. They also have to be reflective and critical.

In this context, our learning environment is composed of two different systems: the AMPLIA system (Vicari et al., 2003), an ITS probabilistic multi-agent environment to support diagnostic reasoning and the modelling of diagnostic hypotheses in domains with complex and uncertain knowledge, such as the medical domain; and the PortEdu system, an educational portal that provides facilities in order to make possible those legacy systems, such as AMPLIA, to work in the WEB.

The main difference between AMPLIA and the related environments is the fact that AMPLIA considers cognitive, social, and affective states in the student

model (Boff et al., 2007). Moreover, the strategies adopted in those systems consider user/system interaction, and not group interaction.

AMPLIA is composed by 4 different agents. The Domain Agent and the Learner Agent are modelled using Bayesian Networks (Jensen, 2001) as internal knowledge representation (beliefs), which have been widely employed to model uncertain domains, as in medicine. The Mediator Agent is modelled using Influence Diagrams (IDs), and the ComServer Agent coordinates the communication activities among other AMPLIA agents (see Figure 1).
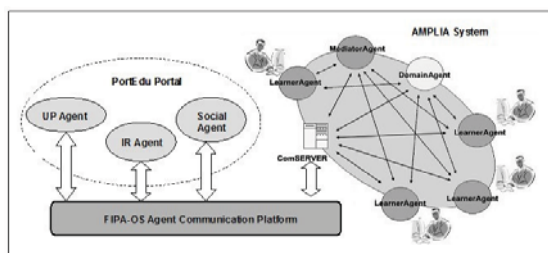
Figure 1. Integration between AMPLIA and PortEdu.

The Learner, the Mediator, and the Domain Agents are cognitive agents, i.e. they have knowledge about how to solve certain problems and they have the ability to reason, making decisions about actions involved in teaching and learning activities. The main idea that guides AMPLIA's design has been the need to offer an open environment in which a student can build a graphical model to represent his/her diagnostic hypothesis for a clinical case using Bayesian Networks. Students can build their diagnostic hypotheses for a specific pathology by themselves or collaborating in a group of students. This construction is supported by the AMPLIA collaborative graphic editor. The student's or the group's network is then compared to that of an expert in the area, and the differences among the networks are monitored by the Mediator Agent, which uses pedagogic strategies based on the constructivist model in order to evaluate the BNs with a satisfactory solution.

As depicted in Figure 1, AMPLIA's ITS is connected to PortEdu. The services provided by the PortEdu agents can be used by any agent-based educational application. In this paper we focus on the Social Agent, which is integrated to the PortEdu architecture to stimulate the interaction among students, tutors and professors. For more information about PortEdu system, see Nakayama (2005).

The main goal of this paper is to present the Social Agent deliberation process, which is a BDI (Bratman, 1988) goal-oriented reasoning that takes into account the probabilistic information through the usage of Bayesian Networks to abstract the mental states. Inside the deliberation process, uncertainty is dealt with in order to decide if an agent believes that a state can be achieved and if desires and intentions are compatible.

In order to detail the Social Agent's deliberation process, this paper is organized as follows. The next section presents AMPLIA's Collaborative Editor, which has been developed to support collaboration among students during learning activities. The Social Agent monitors the Collaborative Graphic Editor in order to obtain information about the students when they are working in a collaborative way. Section "*The Social Agent*" focuses on the Social Agent's reasoning. Section "*Experiments and Results*" shows the preliminary results of our experiments. Finally, the "*Conclusion*" section of the paper presents conclusions and directions for future work.

AMPLIA'S COLLABORATIVE EDITOR

According to learning theories in medicine based on problem-based learning (Peterson, 1997), we decided to extend the AMPLIA editor, that is part of the AMPLIA Learner Agent, to allow several students to operate it simultaneously in a collaborative fashion. Thus, besides the online editing support (see Figure 2), we designed the Social Agent whose main goal was to motivate collaboration and improve group activity.

As depicted in Figure 2, Bayesian networks editing is possible in the system through buttons available in the toolbars. There are menu options to insert nodes, arcs and probabilities.
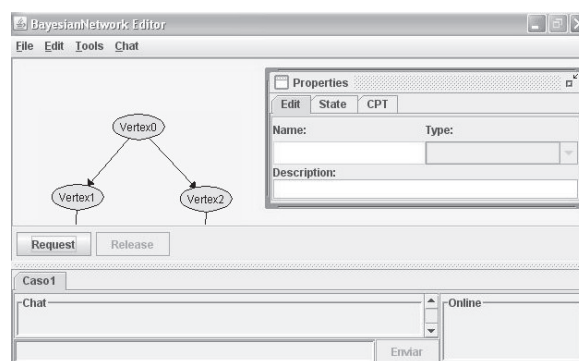
Figure 2. Collaborative Bayesian Net Editor.

In Figure 2 we can see part of the BN that is under development by a group of students. In the second window, on the right, we can see the Node's Properties Editor, where the CPT (Conditional Probability Table) associated with variables (nodes) can be updated. In the bottom of the screen we can find collaborative editing options, including online users' listing and a chat tool.

The Social Agent uses different strategies to suggest a particular student to a workgroup. The interaction between the Social Agent and students occurs through chat tool. The Social Agent acts proactively using the chat tool to send messages to students, such as "you can interact with student *x*", "you could join to group *x*". Students can accept or not the agent's suggestions. Students can also be invited by collegues to join in their group. Students can join different groups whereas each

group can work with different study cases, knowing that within medicine the teaching approach relies mostly on problem-based leaning.

Participation in a group depends on the approval of the student by the members of the group. When the student is invited to join the group, he/she may also accept/decline the offer. When the student refuses to participate in a workgroup, the system may inquire him/her about the reason of the declination by presenting him with the following alternatives: (*i.*) I do not have interest in this subject; (*ii.*) I am temporarily unavailable; and, (*iii.*) I do not have interest in interacting with this group. The actions of the users are stored in the student model (AMPLIA Learner agent). This model is employed when the Social Agent looks for students to join a workgroup. The groups are dynamically formed, based on the task being carried out. The students can participate in several groups simultaneously, according to their interest. Each group must contain at least one student with the leadership role.

The Social Agent also tries to create groups with democratic profiles or with sharing roles, where all team members are able to lead the team. This can occur when the responsibility for the operation of the team is shared. This technique is called role-sharing (Peterson, 1997). Such an approach leads to shared accountability and competencies.

The groups must also be formed by students with different levels of performance. Considering we have six people, for example, including students with performance categorized as excellent, average and regular, it is better to join two classmates of each level.

Since this paper's focus is on the Social Agent, the next section brings detailed information about its reasoning process.

THE SOCIAL AGENT

The Social Agent aims to stimulate student interaction. In order to achieve such goal, the agent keeps a student model that gathers a feature set based on the social and collaborative theories detailed in the "*Introduction*" section. Based on the student model, the Social Agent can suggest workgroups. Workgroups are composed through recommendation of students by Social Agent. The information collected to define a suitable student for recommendation are: Personality Traits, Affective State, Acceptance Degree, Sociability Degree, Net Credit and Leadership.

The Sociability Degree (SD) or Social Profile is built during the students' interaction through a synchronous mechanism (e.g. the chat tool and the collaborative editor). The following information is collected during the students' interaction: number of times that a student had the initiative to talk with another; number of times that a student answered a communication request; individuals with whom the student interacts or has interacted, number of interactions and individuals with

whom the student interacts regularly, and the total number of interactions.

Based on Maturana´s ideas (Maturana and Varela, 1998) we defined the Acceptance Degree (AD), which measures the acceptance between students. Such data is collected through a graphical interface that enables each student to indicate his/her acceptance degree for other students. This value may also be considered from a viewpoint of Social Networks. As the Acceptance Degree is informed by the students themselves based on their affective structures, the value can indicate different emotions, such as love, envy, hatred, etc.

Tutors are suggested by the Social Agent and the students can choose (or not) interacts with the tutor suggested. After the student/tutor interaction, a small questionnaire is submitted to the student who got assistance, with the purpose of collecting information about the performance of the tutor. The questions presented are based on concepts from Social Networks and Sociometry, and may be answered by four qualitative values: "excellent", "good", "regular", and "bad". The questions are presented below:

- How do you classify the sociability of your class fellow?
- How do you classify the help given by your class fellow?

This grade indicates how other individuals see the social capability of this student.

In order to infer the students' personality traits we use the model proposed in Zhou and Conati (2003), based on the OCC model (Ortony, Clore and Collins, 1988). The affective states can be considered as emotion manifestations in a specific time. The research presented in this paper uses the BN to infer emotions proposed in Conati et al. (1997) and Zhou and Conati (2003) to give us states values to Personality Traits and Affective State.

The Net Credit (NC) represents a possible classification for the student's Bayesian Network model (created through the AMPLIA's Collaborative Editor), according to major problems. This student action outcome is received from AMPLIA's Mediator Agent, and it can have the following values: *Unfeasible* (network that does not satisfy the definition of a Bayesian Network, as an oriented acyclic graph and/or a disconnected network), *Incorrect* (network whose model is conceptually incorrect, with presence of an excluder node), *Incomplete* (network that lacks some nodes or relationships considered important), *Feasible* (it is a network different from the built-in model but it satisfies the case study proposed to the learner) or *Complete* (it is identical to the model built by the expert).

Finally, the Leadership represents evidence indicating the students' capacity to lead. For instance, a student that helps other fellow students frequently, or a student that gives his/her opinion during the execution of a task, or a student that makes several changes in the Bayesian Network model built by his/her group.

Next section presents how the Social Agent performs the deliberation process through a Bayesian BDI approach.

Deliberation Process

While Bayesian Networks are a formalized model for representing knowledge, there is not a unique BDI model for agency. The probabilistic BDI model (Fagundes et al., 2007) used to develop the Social Agent employs Bayesian Networks to represent the agent's beliefs. These networks are graphical models that represent causality between variables, allowing the performance of reasoning under uncertainty in a consistent, efficient, and mathematically sound way.

To keep beliefs up to date is a crucial task to agents, since in a dynamic world it is necessary to make decisions and execute actions taking into account the current state of the world. The belief updating corresponds to probabilistic inferences. It is triggered when an agent believes in new evidence.

In Figure 3, we illustrate the Social Agent beliefs. The network was built according to the collected information, including Personality Traits, Affective State, Acceptance Degree, Sociability Degree, Net Credit and Leadership. The Social Agent have such a network for each of the student. The network depicted in Figure 3 relies on a Bayesian Network to assess the suitable group for a student and it includes only a subset of nodes that are necessary to completely specify the BN.
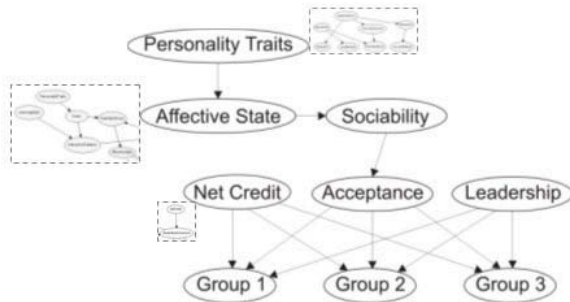
Figure 3. The Social Agent belief network.

The figure's details show the nodes that are assessed by other networks. The node Personality Traits is also a BN that shows the dependence between dominant personality traits, such as Neuroticism, Agreeableness, Conscientiousness and Extraversion (Costa and McCrae, 1992), and students' goals (Exploration, Avoid Mistakes, Prove Hypothesis, Succeed by myself). The BN that assesses the students' affective state (node Affective State) is composed by nodes such as Personality Traits, Goals, Learning Style, Interaction Patterns and Satisfied Goals. This network is based on Conati's model (Zhou and Conati, 2003). Finally, the node Net Credit includes the network where the students' network credibility (node Net Credit) has an influence on the student outcome (students' performance), node Student Action Outcome. Students that believe in the correction of their networks (high Net Credit) use to perform better outcomes (high performance).

The Social Agent represents the Intentional content of the desires through particular states of variables. Consequently, desires are a subset of beliefs. It makes possible to check when a state of affairs is achieved. The nodes Group 1, Group 2 and Group 3, illustrated in Figure 4, represent the beliefs (and desires in this case) of the Social Agent regarding the suggestion of those three different workgroups to the student. Therefore, the network represents how much the student fits on each group, and consequently, how much the agent desires to suggest each group to the student.
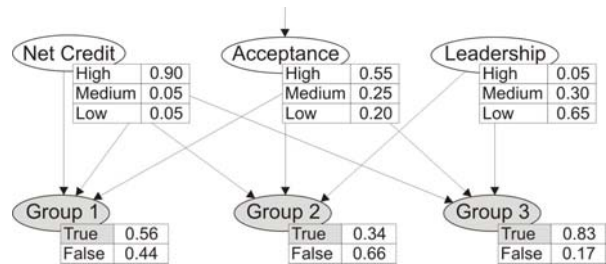
Figure 4. The desires of the Social Agent.

In Figure 4, Group 1 represents the group with high sociability profile. Group 2 is the one in which the dominant feature is positive affective state. Finally, Group 3 has dominant feature high performance.

The first requirement for a desire to become an intention is the belief support. In other words, the agent will not intend a state unless it believes it is possible to achieve that state. This is done by checking the computed marginal probability of the desired state. In Figure 4 the agent will desire the state True of the Group 1 when it believes the current conditions (given by Net Credit, Acceptance and Leadership) support this desire.

A proactive behaviour is exhibited by the agent when it pursues states of the world where desires become feasible. More specifically, this behaviour consists of achieving particular states of parent variables (conditions) based on the values of the conditional probability table of the desired state. They can be viewed as desires connected to a desire through causality.

Intentions correspond to states of affairs that BDI agents have committed to bring about. This commitment makes the agents to check intentions for incompatibility (mutually exclusive states of the world). In our approach, incompatibility between mental states is detected when some evidence indicating the achievement of an intention affects negatively other intention by decreasing its chances of success. This is explained by the fact that desires may share mutually exclusive conditions. For instance, consider the desire to suggest Group 1, whose profile is defined by students with high sociability. As the Social Agent aims to compose heterogeneous groups, it will desire to find

students with negative affective state or students with low or medium performance.

In the deliberation process the BDI agents examine which desires are possible, they choose between competing desires, and commit to achieving them. The process is divided in two stages: the first verifies which desires are possible to be achieved, and the second checks the compatibility among possible desires and intentions. We adopt the terms cautious and bold to describe the agents' behaviour in the following three situations inside of the deliberation process: to decide if the agent believes in particular states, to decide if desires are incompatible with intentions and to decide if desires are incompatible among themselves. By adjusting a threshold value the Social Agent will exhibit a different degree of cautiousness or boldness in those decisions.

For more details about the Bayesian BDI architecture, including the deliberation algorithms and the different agent behaviours, see Fagundes et al. (2007).

Once an agent has committed with some states of affairs, it has to perform actions in order to achieve those intentions. In our testbed, the Social Agent has to interact with the students to suggest workgroups to them.

The three group nodes defined in the belief network do not represent group instances, but three different group profiles. The Social Agent will find a group instance that matches the intended group profile, and then it will suggest that group to the student. If at least one group is found, the intention is successful, otherwise it fails.

EXPERIMENTS AND RESULTS

AMPLIA environment has been tested in seminars involving medical professors and students at Porto Alegre's Clinic Hospital (Hospital de Clínicas de Porto Alegre, Brazil). Each seminar was divided in two phases: (1) discussion of pedagogic resources and of theoretical concepts about uncertain domains, probabilistic networks, and knowledge representation; (2) the medical specialists had to build expert's BN models on their domain areas, which were used by the AMPLIA Domain Agent. We could observe in these seminars that AMPLIA is an efficient resource to improve medical learning. More information about these tests may be found in Flores et al. (2005).

Regarding the Social Agent, we have developed two experiments until this moment.

The first of them was a case study to demonstrate the possibility to exchange Bayesian knowledge between the Social Agent and AMPLIA's Learner Agent. Both are Bayesian agents, belonging to PortEdu and AMPLIA respectively. The idea has been to enable the Learner Agent to send Bayesian information to the Social Agent by employing Semantic Web technologies, namely OWL, Bayesian Networks and Agent Communication Languages. This experiment was described in (Boff et al. 2006).

The second experiment had as a main goal to employ the Social Agent to form workgroups based on the features of twelve students and on the strategies detailed in section "*The Social Agent*". According to these strategies, two scenarios had to be considered, as described in Tables 1 and 2.

Table 1 and 2 present the following headers: ST (Student), DPT (Dominant Personality Trait), S (Sociability), A (Acceptance), NC (Net Credit), L (Leadership), AS (Affective State), DH (Diagnostic Hypotheses) and GS (Group Suggestion).

The AS (Affective State) column can assumes values such as: P (Pride), J (Joy) and D (Distress). In both tables the Dominant Personality Trait can be E (Extrovert), I (Introvert) and A (Agreeableness). The columns related to Sociability, Acceptance, Net Credit and Leadership can assume H (High), M (Medium) or L (Low) values. The Diagnostic Hypotheses in the first scenario 1 correspond to Cardiac Evaluation (CE), whereas in the scenario 2 correspond to Diabetic Neuropathy (DN).

Scenario 1 presents three groups proposed by the Social Agent. The first group shows a balanced profile distribution. The second group put together four students with different profiles. The strategy used has been to compose heterogeneous groups, i.e. to mix students with different personality traits and social roles (leadership). The third group assembled three extrovert students with only one introverted student. But, in this particular case, the likelihood that the introverted student feels intimidated is very small, as he/she has a high value on the Net Credit and his/her Affective State has the value "Joy", which is a positive social feature. In this scenario, all students were working in the same diagnostic hypothesis, which was "Cardiac Evaluation". The Social Agent suggested heterogeneous groups, but with small differences among members. For example: an introverted and distressed student should work better with an agreeable or extroverted student that is happy. It is possible that proud students, in this case, could intimidate the participation of an extremely introverted student.

Table 1. Scenario 1.

ST	DPT	S	A	NC	L	AS	DH	GS
1	E	H	H	M	H	P	CE	1
2	E	M	H	M	H	J	CE	3
3	I	L	L	L	L	D	CE	2
4	A	M	L	M	L	J	CE	2
5	E	M	L	H	M	J	CE	1
6	I	M	M	H	L	J	CE	3
7	I	L	L	M	L	D	CE	1
8	I	L	L	L	L	D	CE	1
9	E	H	M	M	M	J	CE	3
10	E	H	H	H	H	J	CE	2
11	E	M	M	H	H	P	CE	3
12	E	M	M	M	M	J	CE	2

Scenario 2 shows three balanced groups proposed by the Social Agent. Table 2 presents the organization of these groups.

Table 2. Scenario 2.

ST	DPT	S	A	NC	L	AS	DH	GS
1	E	H	M	H	H	P	DN	1
2	E	H	H	L	M	J	DN	2
3	I	L	H	H	M	J	DN	1
4	A	M	L	M	L	J	DN	1
5	A	H	M	H	M	P	DN	2
6	I	L	L	L	L	D	DN	2
7	E	H	H	M	H	J	DN	3
8	I	L	L	L	L	D	DN	1
9	I	L	L	L	L	D	DN	2
10	I	L	L	L	L	D	DN	3
11	I	M	M	M	L	J	DN	3
12	I	M	M	H	L	J	DN	3

In this scenario, all students were working in the same diagnostic hypothesis, which was "Diabetic Neuropathy". The groups created by the Social Agent were composed by three students with different profiles. Each group had as a member a student with a different personality trait and a balanced (but diverse) degree of Sociability, Social Acceptance, Net Credit (or level of performance in the learning subject), Leadership and Affective State. In this example, we had some students with exactly the same profile (students 8, 9 and 10). In order to keep the group balanced, these students have been distributed in different groups by the Social Agent. The ideas described in this paper show our perspective on how to analyse, interpret and model the complex phenomena that occurs in the teaching-learning process, through the modelling of the student. The validation of these ideas and their generalization can only happen over time and within real world application and testing. In the future, we can also validate a group formation with a declined invitation and replan its repartition when some student refuse to join the proposed group.

CONCLUSION

We have built a Social Agent which interacts with the users in order to motivate group formation among students and to promote collaborative learning. The Social Agent identifies suitable students that can play the role of a tutor, and to recommend them to other students needing assistance. The tutor recommendation mechanism explores the social dimension through the analysis of emotional states and social behaviour of the users. In this direction, we aim to contribute to the design of learning environments centered in students' features and collaborative learning.

In a real classroom, students form workgroups considering mainly the affinity between them. Sometimes, workgroups are composed taking into account geographic proximity (especially for Distance Learning), but not always these groups present a good performance in learning activities.

The AMPLIA editor can be considered a collaborative tool which allows the creation of virtual workgroups to solve tasks in a collaborative way. In addition, the AMPLIA environment contributes to CLEs research as it considers in the students' models, some cognitive, affective and social states.

When students are involved in the same task and with the same goal, they are compromised to each other. Suggesting students to help others we can motivate collaboration and initiate students' interaction in the AMPLIA environment. We aim to reduce the professors' involvement and give more autonomy to students.

The strategies adopted by the Social Agent are based on established theories presented in section "*The Social Agent*". All knowledge has been used to infer the student's social skills, and then place them in groups in order to obtain the best individual and collaborative performance. In the future, we can adopt some strategies proposed by Bowen and Jackson (1985-6) in order to improve learning groups. Such as: work at giving good feedback, get silent members involved, confront problems, vary the leadership style as needed, work at increasing self-disclosure, summarize and review your learning from group experiences (analyze the data to discover why the group was more effective or less so, and provide final feedback to members on their contribution) and celebrate the group's accomplishments.

We focused on the Social Agent's deliberation (selection of intentions), leaving outside the paper scope issues such as performance measurements of the agent's architecture, methodology of development, uncertainty on perceptions, and advanced planning techniques. The adopted model has an implicit representation of incompatibility among mental states through causal relations and conditional probabilities. By this we mean that the Social Agent infers incompatible desires by checking mutually exclusive conditions.

The results presented in section "*Experiments and Results*" are based on a method which aims at more effective group composition. An effective group is a group with cohesion, good performance in learning activities and minimum conflicts. That experiment is a starting point to indicate that Social Agent reasoning can be used to make up groups according to group dynamics literature.

Future work is twofold. The first research direction is concerned with the improvement of the approach here presented to cover the reconsideration of intentions and commitment strategies. Still on the deliberation process, we intend to experiment AI techniques to develop a dynamic threshold function that adjusts its value according to the circumstance. The second research direction, concerning the Social Agent design, consists of exploring other intentions in order to improve the workgroup formation.

REFERENCES

Boff, E., Santos E.R. and Vicari, R. M. (2006) 'Social agents to improve collaboration on an educational portal'. In: *IEEE International Conference on Advanced Learning Technologies*, IEEE Computer Society (2006) 896-900.

Boff, E., Flores, C. D., Silva, M., Vicari, R. M. (2007) 'A Collaborative Bayesian Net Editor to Medical Learning Environments'. *Artificial Intelligence and Applications (AIA 2007)*. Innsbruck, Austria. 2007.

Bowen D. D. and Jackson, C. N. (1985-6) 'Curing those 'ol 'Omigod-Not-Another-Group-Class' blues', *Organizational Behavior Teaching Review*, 10(4), 21-23, 1985-6.

Bratman, M.E., Israel, D.J., Pollack, M.E. (1988) 'Plans and resource-bounded practical reasoning'. *Computational Intelligence 4* (1988) 349-355.

Cao, Y., Sharifi, G., Upadrashta, Y. and Vassileva, J. (2003) 'Interpersonal Relationships in Group Interaction in CSCW Environments', *Proceedings of the User Modelling UM03 Workshop on Assessing and Adapting to User Attitudes and Affect*, Johnstown, June 22, 2003.

Conati, C. et al. (1997) 'On-line student modeling for coached problem solving using Bayesian networks', Proceedings *User Modeling: Sixth International Conference - UM97*, Springer, Vienna, 1997, p. 231-42.

Costa, P. T. and R. R. McCrae (1992). 'Four ways five factors are basic'. *Personality and Individual Differences* 1 13: 653-665.

Dillenbourg, P., Baker, M., Blaye, A. and O'Malley, C. (1995) 'The evolution of research on collaborative learning'. In: P. Reimann & H. Spada (Eds). *Learning in humans and machines. Towards an interdisciplinary learning science*, 189-211. London: Pergamon, 1995.

Fagundes, M. S, Vicari, R. M. and Coelho, H. (2007) 'Deliberation Process in a BDI Model with Bayesian Networks'. In: *10th Pacific RIM International Workshop on Multi-Agents (PRIMA 2007)*. Lecture Notes in Artificial Intelligence. 2007.

FIPA. FIPA ACL Message Structure Specification SC00061. http://www.fipa.org/specs/fipa00061/. Accessed in 2006.

Flores, C. D., Seixas, L., Gluz, J. C. and Vicari, R. M. (2005) 'A Model of Pedagogical Negotiation'. In: Multi-Agent Systems: Theory and Applications Workshop - MASTA, 2005, Covilhã. *12th Encontro Portugues de Inteligencia Artificial - EPIA 2005* (EPIA 2005, LNAI 3808). Berlin: Springer Verlag, 2005. v. 1. p. 488-499.

Giraffa, L. M., Viccari, R. M. and Self, J. (1998) 'Multi-Agent based pedagogical games'. *Proceedings of ITS*, 4, 1998.

Jensen, F.V. (2001) 'Bayesian Networks and Decision Graphs'. Springer.

Mathoff J. and Van Hoe, R. (1994) 'Apeall: A multi-agent approach to interactive learning environments'. In: *European Workshop On Modeling Autonomous Agents Maamaw*, 6, Berlin, 1994.

Maturana, H. and Varela, F. (1998) *Tree of Knowledge: The Biological Roots of Human Understanding*. Shambhala Publications, Boston, MA, 1998.

Nakayama, L., Vicari, R. M. and Coelho H. (2005) 'An Information Retrieving Service for Distance Learning', *The IPSI BgDTransactions on Internet Research*, v. 1, n. 1, Jan. 2005.

Norman, T. J. and Jennings, N. R. (2000) 'Constructing a virtual training laboratory using intelligent agents'. In: *International Journal of Continuous Engineering and Life-long Learning*, 2000.

Ortony, A., Clore, G. and Collins, A. (1988) *The Cognitive Structure of Emotions*. Cambridge University Press, 1988.

Peterson, M. (1997) 'Skills to Enhance Problem-based Learning'. Med Educ Online [serial online] 1997; 2,3. Available from: URL http://www.med-ed-online/

Piaget, J. (1995) 'Explanation in sociology'. In: *Sociological studies*, New York: Routledge.

Vicari, R. M., Flores, C. D., Silvestre, A. M., Seixas, L., Ladeira, M. and Coelho, H. (2003) 'A Multi-Agent Intelligent Environment for Medical Knowledge'. *Artificial Intelligence in Medicine*, Elsevier Science B. V., v. 27, p. 335-366, 2003.

Vygotsky, L.S. (1999) 'The collected works of L. S. Vygotsky'. New York: Plenum Press, c1987-c1999 v. 1-6.

Wooldridge, M. (2000) 'Reasoning about Rational Agents'. *Intelligent Robots and Autonomous Agents*. The MIT Press, Cambridge, Massachusetts, 2000.

Zhou X. and Conati, C. (2003) 'Inferring User Goals from Personality and Behavior in a Causal Model of User Affect', Proceedings of IUI 2003, *International Conference on Intelligent User Interfaces*, Miami, FL, U.S.A. p. 211-218, 2003.

OVERUSE OF NATURAL RESOURCES: FINDING SOLUTIONS WITH AGENT-BASED SIMULATION

Hans-Joachim Mosler
System Analysis, Integrated Assessement and Modelling
Eawag
CH-8600 Duebendorf, Switzerland
E-Mail: mosler@eawag.ch

KEYWORDS
Agent-based simulation, natural resources, population model, social simulation, psychology.

ABSTRACT

A computer simulation will be presented, which allows studying populations of 10,000 resource-using people. Simulation experiments can be conducted with populations whose initial parameters lead to resource depletion over the long or short term. Systematic experiments using this approach make it possible to determine which measures may lead to sustainable resource use.

INTRODUCTION

Today the sustainable collective use of natural resources is a greater challenge than ever before. Since the seminal contribution of Garrett Hardin on "the tragedy of the commons" (1968) social scientists have investigated the determinants of individual consumption behavior and developed models to explain individual behavior in collective resource management (e.g., Kopelman, Weber, & Messick, 2002). Despite intensive scientific efforts, the problem of sustainable collective consumption of natural resources has become even bigger today, and it is expected to grow in importance in the future, with further depletion of the world's resources and simultaneous increasing demand due to widespread modernization and over-population.

To study sustainable collective consumption, experiments with different populations under different conditions would be needed, which is nearly impossible to realize within real human populations. A computer simulation, on the other hand, requires only effort to test many different measures designed to encourage sustainable resource use by a population. There are a lot of studies utilizing the method of computer simulation to study resource dilemmas (e.g. Nowak & Sigmund, 1992; Liebrand & Messick, 1996; Macy, 1996; Grant & Thopson, 1997; Deadman, 199; Castillo & Saysel, 2005; for an overview see Jager, 2000). Ernst & Spada (1993), for example, designed a

computerized knowledge-based model of action and interaction in social dilemmas by constructing a model from their own dilemma game data. Jager (2000) simulates consumers using a multi-theoretical framework that integrates various theories that appear to be relevant in understanding consumer behavior. Their simulations investigate the long-term dynamics of resource dilemmas (Jager, Janssen & Vlek, 2002).

For our purpose an agent-based computer simulation was designed based on social science literature. A basic model of an agent was constructed, which serves as the basis for the simulated influencing and resource-use processes. The population has 10,000 identically structured copies of this agent, equipped however, with different individual characteristics. The agent model yields information about the internal psychological processes that occur when people use environmental resources.

The agent-based model is founded on the Social-Ecological Relevance model (the SER model) of Mosler & Brucks (2003). This model assumes that certain individuals set greater store by ecological information, while others attach more importance to social information. This weighting naturally influences their resource use behavior, since there may be considerable disparity between ecological and social information concerning the resource. Aided by this core concept, various findings of social dilemma research were integrated in the model. In the simulation, single agent's behavior corresponds with research findings concerning the variables resource use of others, social values, resource size, resource uncertainty, perceived cause of resource abundance or scarcity (attributions).

The agents in the populations have at their disposal differently structured social contact nets (number of friends, acquaintances, neighbors, and strangers they observe). These personal contacts both influence the individual, and are influenced by him or her.

Experiments can be conducted with populations whose initial parameters lead to resource depletion over the long or short term, e.g. a population containing a high proportion of uncooperative individuals, a population with a higher average resource uncertainty, or populations with unfavorable combinations of parameter values. The measures investigated were advertising campaigns and the employment of

promoters. It is assumed that an advertising campaign affects certain variables across a sector of the population. In the case of promoters, selected and specially trained individuals exert pressure on their social surroundings. Systematic experimentation using the simulation approach makes it possible to determine which measures applied to which populations could lead to sustainable resource use.

AN AGENT-BASED MODEL OF RESOURCE USE

The structure of the model is displayed in Figure 1. Arrows show the variables with their names. The variables are calculated in the numbered blocks.
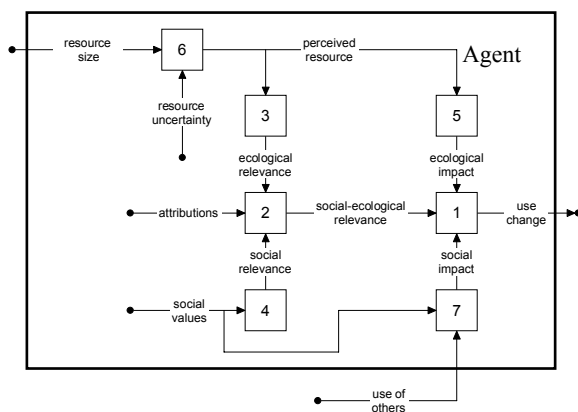
Figure 1: Block diagram of the social-ecological relevance model.

The agent gets as input the size of the resource from the state of the resource and the resource use of others from the contact with other agents in his network. As internal input variables the agent has his estimation of resource uncertainty his attributions and social values. Attribution means that the agent sees the reason of the state of the resource either in the behavior of the population or in the resource itself (e.g. self deterioration). Social values mean that the agent is either cooperative or non-cooperative. The perceived resource defines the ecological relevance which means the importance the agent poses to resource issues. Social values determine the social relevance indicating the importance the agent gives to social issues. The social impact results from social values and use of others and together with the ecological impact it is weighted according to the social-ecological relevance and the change in use is calculated.

The values of all variables range on a scale from 0 to 100. Besides the two end points of 0 and 100, the value of 50 is very important for the bipolar variables (social values, attributions, social-ecological relevance), for it represents the point of neutrality. In the following, the transition functions in the blocks of figure 1 are explained according to Mosler & Brucks (2003).

Block 1
use_change = social_weight * social_impact + ecological_weight * ecological_impact.
social_weight = social_ecological_relevance / 100
ecological_weight = 1 - social_ecological_relevance / 100

The output of the model is use change, which represents the individual's change in resource use. Negative values of use change indicate a decrease in use; positive values show an increase. Zero indicates unchanged use. Use change is dependent on ecological impact and social impact. First, both of these impacts are weighted by an ecological and a social weighting factor, respectively. These weighting factors are computed from social-ecological relevance. Due to the one-dimensionality of social-ecological relevance, the sum of these complementary weighting factors always equals one. The social and ecological impacts, modified by the respective social and ecological weights, determine use change.

Block 2
social_ecological_relevance = (social_relevance – ecological_relevance + 100) / 2 + influence_of_attributions
influence_of_attributions = (50 – attributions) / A.

Parameter A determines the strength of the influence of attributions. Social-ecological relevance is calculated as the average of social relevance and ecological relevance plus the influence of causal attributions (which belongs neither to the ecological nor to the social factors but stands in between them). We assume that the impact of attributions on use change is not direct, but indirect, via social-ecological relevance. People who attribute blame for the state of the resource to others place more weight on social factors, while persons who see the cause for the state of the resource in the resource itself (such as a natural shortage) weigh ecological factors more heavily. It has been shown that when people see the group as the cause of the state of the resource (operationalized by Messick at al., 1983), their use behavior is different than when they hold the resource itself responsible (operationalized by Samuelson et al., 1984). This was demonstrated clearly in an experiment by Rutte, Wilke, & Messick (1987), who found that subjects harvested more from the resource in abundance condition than in the scarcity condition. Furthermore, the difference in harvest size between scarcity and abundance conditions was greater in the resource-caused condition than in the group-caused condition. In accordance with these considerations, social-ecological relevance can be modified by attributions, up or down to a given degree.

Block 3
ecological_relevance = 100 – perceived_resource

Ecological relevance is calculated from perceived resource (the individual's perception of the state of the

resource) on the theoretical assumption that ecological relevance increases with a decreasing perceived resource. This assumption means that the worse the state of the resource, the more relevant people find ecological information. This implies, due to the one-dimensionality of social ecological relevance, that with ecological uncertainty and a higher value of the variable perceived resource, social information will be used more. This agrees with empirical findings by van Dijk, Wilke, Wilke, & Metman (1999), where group members avoided basing their use decisions on environmental information that they were not sure about.

Block 4
social_relevance = social_values

The influence of social values (cooperative vs. non-cooperative) on social-ecological relevance bases on the assumption that non-cooperators disregard ecological factors in a resource crisis. Cooperators, in contrast, are much more likely to adapt their use to ecological factors; they reduce use when resource size decreases in order to prevent depletion of the resource. By postulating for non-cooperators that ecological factors are less important, our model at the same time (due to the one-dimensionality of social-ecological relevance) postulates that they find social factors more relevant. As non-cooperators always place less significance on the meaning of ecological factors, they care less about the resource size as well. Thus they decrease their use less than cooperators when the resource is in a poor state and also increase it less when the state of the resource is good. Overall, non-cooperators harvest more than cooperators do. This corresponds quite well with experimental findings reported by Parks (1994). In the model, therefore, social relevance increases (or ecological relevance decreases) the more cooperative the individual is.

Block 5
ecological_impact = perceived_resource – 50

A number of experiments have demonstrated that use decreases with decreasing values for perceived resource. While this can be explained in terms of people's motivation to preserve a resource (see Messick et al., 1983), supporting empirical evidence is also available. In an international comparison, Samuelson et al. (1984) were able to demonstrate that subjects harvested less in the overuse condition than in the optimal-use or under-use conditions and that harvest sizes tended to increase through time in the latter two conditions but little or not at all in the overuse condition. The agents in the model therefore use the resource in relation to perceived resource. They reduce their use when perceived resource is below the optimum and increase it when perceived resource is above that optimum.

Block 6
perceived_resource = resource_uncertainty * resource size
(uncertainty is calculated through a tabular function)

Resource size is a dynamic variable that is calculated in the simulation model at every point in time. It is modified to become perceived resource through multiplying it by the uncertainty factor resource uncertainty, which is calculated interpolatively using experimental values from Rapoport et al. (1992) with the aid of a tabular function. The uncertainty factor can range from 1 to 1.5. This means that with maximum uncertainty, the resource will by overestimated by 1.5 times.

Block 7
IF social_values > 50 AND use_of_others > 50
DO social_impact = 50 – (100 – use_of_others)
ELSE social_impact = 50 – use_of_others

In addition to temporal dynamics, social dynamics have to be modeled too. To this purpose, we allow any number of individuals in the model to interact with each other in an agent-based simulation (compare Gilbert & Troitzsch, 1999). As a logical consequence of allowing for social dynamics, the use of others (feedback to the individual on how much the others are using) is included in the model. Experimental investigations have demonstrated that knowledge of the resource use of others in a resource crisis has a strong influence upon a person's own behavior. If others show either under-use or optimal use (sustained use), the findings are quite clear, because harvest behavior in both the under-use and optimal-use conditions tends to increase over time (Messick et al., 1983). An experiment by Kramer et al. (1986) also revealed a clear tendency for personal use to increase under sustained use conditions. In the model, therefore, an agent reduces use when others overuse, but increases use when others use less than the optimum. The situation changes if the resource is overused by the group and is in danger of being depleted in the future. It seems that in this case, the individual characteristics of the user play a greater role in influencing use than under sustained use conditions. Individual characteristics include social values, as was established by Kramer et al. (1986) as follows: When the future of the common resource was threatened by collective overuse, non-cooperators' behavior showed virtually no adjustment to decline of the resource level across trials. Cooperators, on the other hand, demonstrated personal restraint, and the magnitude of this restraint increased until the final trial block. Our model accords with these experimental findings by assuming for non-cooperators that instead of reducing their use, they increase use when they note that others are overusing.
The model was implemented in THINK PASCAL on a Macintosh computer.

VALIDATION OF THE MODEL

The complete model was validated by analyzing whether or not it is possible to replicate the findings of real experiments in the literature. Tests to date have shown that the model replicates the following experimental findings successfully (see Mosler & Brucks, 2003): for resource size, the findings of Samuelson et al. (1984); for resource uncertainty, the findings of Budescu et al. (1990) and Rapoport et al. (1992); for social values, the findings of Kramer et al. (1986); for use of others the findings of Messick et al. (1983); and for attributions, the findings of Rutte et al. (1987). Because the model appears to implement the effects of the variables adequately, it follows that the model is a valid representation of the findings on these variables.

THE POPULATION MODEL

A population with 10,000 agents is simulated (see figure 2). The parameters of the agent variables are assigned at random according to a normal distribution around a predefined mean and standard deviation. Each agent makes a decision and uses the resource. Resource use problems within this population are caused by the behavior of these many agents. The using behavior of each agent is perceived by the other agents and influences their decisions. The range and number of the agent's network can be defined differentially; this means that each agent can have 1 - 50 contacts.

Each agent uses a certain rate of the resource; all the rates are summed up and give the total use of the population of the resource. In each run after the total use is subtracted from the resource it regenerates with a certain factor (in the presented examples the regeneration factor is 1.2). As the resource is used by the whole population its size changes and this is also noticed by the agents. In dependency of the changed resource size and the use of the other agents the agent decides on how to change his use of the resource for the next run.
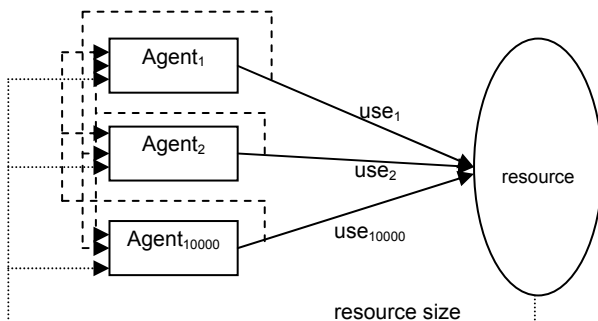
Figure 2: A population of 10'000 agents uses a resource. The solid arrows indicate the use of the agents of the resource; the dotted arrows indicate the perceived remaining resource size after the use of the population and the regeneration; the dashed arrows

indicate that the agent perceives the use of other agents in his network (use of others).

INTERVENTIONS IN POPULATIONS

For the experiments presented here those cases were chosen which are likely to be found in reality and in which the behavior of the population is detrimental to the resource. The baseline experiment (see figure 3) is conducted with a population which is on average non-cooperative (value 60) this leads to a constant depletion of the resource until its value drops down below 10, which means that the resource is destroyed (as for example fish stocks which under a certain population number cannot reproduce).
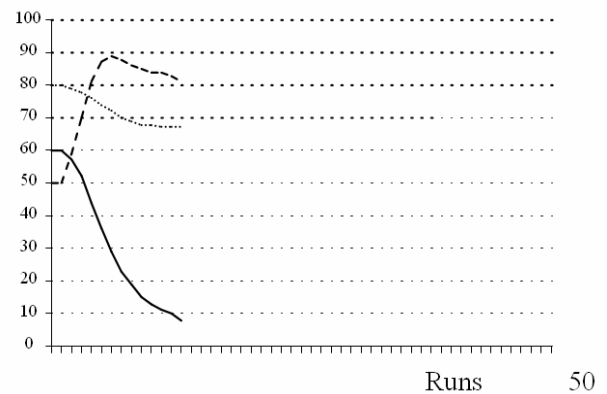
Figure 3: A population of 10'000 that is non-cooperative (social values=60) and with attribution 'group' (20); the resource size starts with 60 (solid line), the average use with 50 (dashed line), and the relevance with 80 (dotted line)

In the following experiment (see figure 4) the same population is used with the same starting values but now we have 20% of the agents which attribute the decline of the resource to the resource (attributions=90).
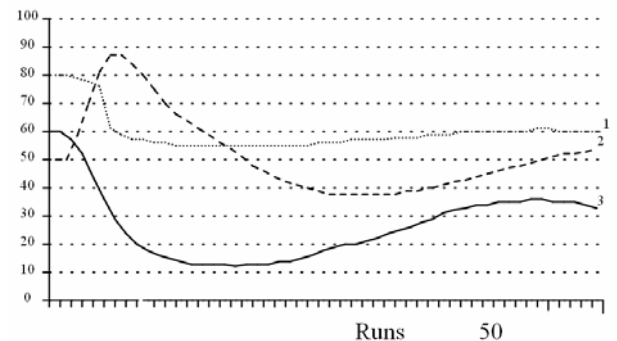
Figure 4: Same population as in figure 3 but 20% of the agents acquire an attribution 'resource' (90); the resource size starts with 60 (solid line), the average use with 50 (dashed line), and the relevance with 80 (dotted line)

This could be the case when parts of the population come to the conclusion that the decline of the resource happened rather due to causes in the ecological system than to social reasons. With this measure it is possible to stop the detrimental development and to improve the size of the resource in a sustainable way.

To demonstrate the importance of the social network another experiment was carried out where the network of the agents was reduced from 5 to only 1 contact agent (see figure 5).
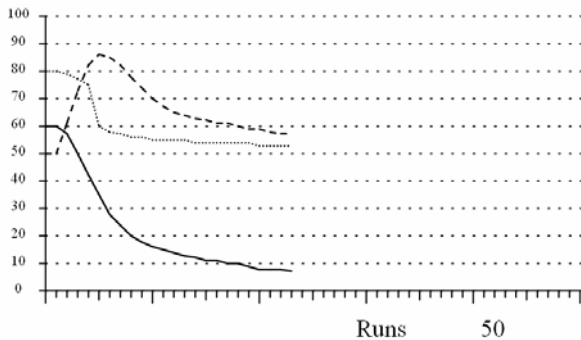
Figure 5: Same population as in figures 3 and 4 and 20% of the agents acquire an attribution 'resource' (90), the agents have only one contact; the resource size starts with 60 (solid line), the average use with 50 (dashed line), and the relevance with 80 (dotted line)

Here it is not possible to stop the detrimental process because the 20% of agents with a changed attribution have no far-ranging effect on their social surrounding.

With another series of experiments the effect of uncertainty is demonstrated. The population is also mainly non-cooperative (social value=60), with attribution 'group' (20), and with high resource uncertainty (85); the resource size starts with 50, the average use with 50, and the relevance with 20 which means that the state of the resource is more important for the decision to use the resource than the social variables. As a result, the size of the resource is kept at a constant but low level (see figure 6).
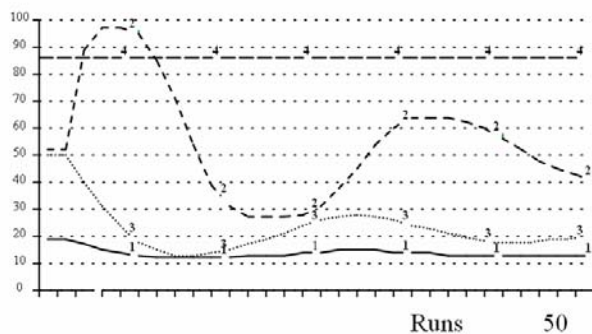
Figure 6: Population of 10,000 which is non-cooperative (social value=60), with attribution 'group' (20), and with high resource uncertainty 85 (dashed line-4); the resource size starts with 50 (dotted line-3), the average use with 50 (dashed line-2), and the relevance with 20 (solid line-1)

By reducing the uncertainty of the population about the size of the resource (see figure 7) the resource can be used more adequately and it is transferred continuously into a higher size which is beneficial for the entire population. The decisive effects of uncertainty can thus be demonstrated.
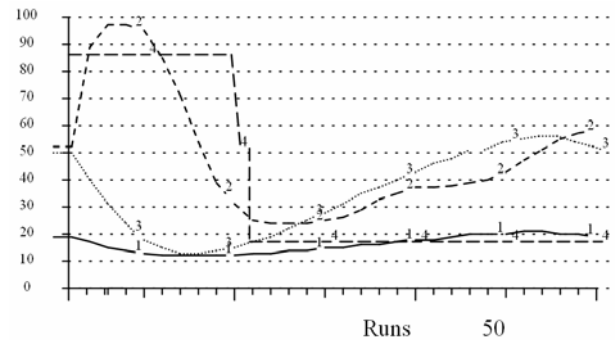
Figure 7: Same population as in figure 6 but resource uncertainty (dashed line-4) is reduced markedly; the resource size starts with 50 (dotted line-3), the average use with 50 (dashed line-2), and the relevance with 20 (solid line-1).

DISCUSSION

An agent-based model of human resource use was built on the basis of theories found in the relevant scientific literature. This has the advantage that the model is not biased through the modeler's preferences and moreover it can be validated by comparing the simulation results with published findings. From the successful validation it can be concluded that the agents in the simulation behave like real individuals in the experiments.

Based on this valid agent model, simulations with populations of 10'000 agents were conducted. The population simulations revealed measures with which it could be possible to bolster the detrimental effects of a mainly uncooperative population upon a resource by changing the perceptions of a certain percentage of the population. Also the essential effects of the visibility of the changed behavior in the social network could be confirmed. It could be demonstrated that the uncertainty about the resource size could play an important role for the sustainable conservation of an environmental resource.

These population simulations show the usefulness of a validated agent-based simulation. Insights could be gained into the possibly detrimental interplay between an overusing population and an environmental resource. Measures were tested which could lead human populations to a sustainable use of resources.

REFERENCES

Budescu, D. V., Rapoport, A., & Suleiman, R. 1990. Resource dilemmas with environmental uncertainty and asymmetric players. European Journal of Social Psychology, 20, 475-487.

Castillo, D. & Saysel, A.K. 2005. Simulation of common pool resource field experiments: a behavioral model of collective action. Ecological Economic, 55, 420-436.

Deadman, P.J. 1999. Modelling individual behaviour and group performance in an intelligent agent-based simulation of the tragedy of the commons. Journal of Environmental Management, 56, 159-172.

Ernst, A. M., & Spada, H. 1993. Modeling actors in a resource dilemma: A computerised social learning environment. In D. M. Towne, T. de Jong, & H. Spada (Eds.), Simulation-based experiental learning (pp. 105-120). Berlin: Springer.

Gilbert, N., & Troitzsch, K. 1999. Simulation for the social scientist. (1 ed.). Buckingham, PA: Open University Press.

Grant, W. E. & Thompson, P. B. 1997. Integrated ecological models: simulation of socio-cultural constraints on ecological dynamics. Ecological Modelling 100, pp. 43–59.

Hardin, G. 1968. The tragedy of the commons. Science, 162(3859), 1243-1248.

Jager, W., 2000. Modelling consumer behaviour. Doctoral dissertation. Groningen: University of Groningen, Dept. of Psychology. http://docserver.ub.rug.nl/eldoc/dis/ppsw/w.jager/

Jager, W., M. A. Janssen, & Vlek, Ch.A.J. 2002. "How uncertainty stimulates over-harvesting in a resource dilemma: Three process explanations." Journal of Environmental Psychology 22(3): 247-263.

Kopelman, S., Weber, M. J., & Messick, D. M. 2000. Factors influencing cooperation in commons dilemmas: A review of experimental psychological research. In E. Ostrom, T. Dietz, N. Dolsak, P. C. Stern, S. Stovich & E. U. Weber (Eds.), The drama of the commons (pp. 113-156). Washington: National Academy Press.

Kramer, R. M., McClintock, C. G., & Messick, D. M. 1986. Social values and cooperative response to a simulated resource conservation crisis. Journal of Personality, 54, 576-592.

Liebrand, W. B.G. & Messick, D. M. 1996. Computer simulation of cooperative decision making. In: W. B. G. Liebrand and D. M. Messick, Editors, Frontiers in Social Dilemmas Research, Springer, Berlin.

Macy, M. 1996. Natural selection and social learning in a prisoner's dilemma: co-adaptation with genetic algorithms and artificial neural networks. In: W. B. G. Liebrand and D. M. Messick, Editors, Frontiers in Social Dilemmas Research, Springer, Berlin.

Messick, D. M., Wilke, H., Brewer, M. B., & Kramer, R. M. 1983. Individual adaptations and structural change as solutions to social dilemmas. Journal of Personality and Social Psychology, 44(2), 294-309.

Mosler, H. J., & Brucks, W. M. 2003. Integrating commons dilemma findings in a general dynamic model of cooperative behavior in resource crises. European Journal of Social Psychology, 33(1), 119-133.

Nowak, M. A. & Sigmund, K. 1992. Tit for tat in heterogeneous populations. Nature 359, pp. 250–253.

Parks, C. D. 1994. The predictive ability of social values in resource dilemmas and public goods games. Personality and Social Psychology, 20(4), 431-438.

Rapoport, A., Budescu, D. V., Suleiman, R., & Weg, E. 1992. Social dilemmas with uniformly distributed resources. In W. Liebrand, D. Messick, & H. Wilke (Eds.), Social Dilemmas - Theoretical issues and research findings (pp. 43-57). Oxford: Pergamon Press.

Rutte, C. G., Wilke, H. A. M., & Messick, D. M. 1987. Scarcity or abundance caused by people or the environment as determinants of behavior in the resource dilemma. Journal of Experimental Social Psychology(23), 208-216.

Samuelson, C. D., Messick, D. M., Rutte, C. G., & Wilke, H. 1984. Individual and structural solutions to resource dilemmas in two cultures. Journal of Personality and Social Psychology, 47(1), 94-104.

van Dijk, E., Wilke, H., Wilke, M., & Metman, L. 1999. What information do we use in social dilemmas? Environmental uncertainty and the employment of coordination rules. Journal of Experimental Social Psychology, 35(2), 109-135.

Wilke, H. A. M. 1991. Greed, efficiency and fairness in resource management situations. In W. Stroebe & M. Hewstone (Eds.), European Review of Social Psychology (Vol. 2, pp. 165-187). Chichester: John Wiley & Sons, Ltd.

AUTHOR BIOGRAPHY

HANS JOACHIM MOSLER is an associate professor of social and environmental psychology at the University of Zurich, Switzerland and head of the research group "Modelling Social Systems" at EAWAG (Swiss Federal Institute of Aquatic Science and Technology). His current research is on behavioral change in large populations using agent-based simulations.
Web-page:
http://www.eawag.ch/research/siam/sozialesysteme/group_websites/team/team.xml

SIMULATION OF INTELLIGENT SYSTEMS – BLOCK STRUCTURE REVISITED

Eugene Kindler
Ivan Krivy
University of Ostrava
Department of Informatics and Computers
CZ – 701 03 Ostrava, Dvorakova 7
Czech Republic
E-mail: ekindler@centrum.cz
ivan.krivy@osu.cz

KEYWORDS

Agents, Local Agents, Nested Agents, Simulation, Object-oriented programming, SIMULA

ABSTRACT

In this paper essential aspects of block-oriented world viewing are emphasized and presented in a relation to simulation models of systems managed by complex and sophisticated decision rules. The construction of such models can be essentially simplified when one combines the agent-oriented, object-oriented and block-oriented paradigms of programming. The use of the described way leads to certain paradoxical relations among computing agents. Some applications are added.

BASIC SUPPOSITIONS

Main Technique of Constructing Simulation Models

The technique of computer simulation is commonly applied in case the studied system is complex. The consequence is that the simulation models are complex, too, and there are difficulties with their programming and debugging. Since the early sixties, the simulation programming languages have enabled evading such difficulties. Instead of algorithmizing the run of the simulation models, the users of these languages had only to describe the simulated system and the language processor automatically translated such descriptions into corresponding algorithms (computer programs).

Already one of the first simulation languages, GPSS, offered their users to declare certain sets of elements equipped by similar configurations of their private entities (called "parameters") and influencing the computer run by commonly formulated "life rules", i.e. algorithms switched according to a common "scheduling system" related to common modeled Newtonian time flow (Gordon 1961, Florea and Kalisz 2000).

The first object-oriented language SIMULA 67 (Dahl and Nygaard 1968) introduced also the life rules, generalized them to "co-routines" and led to the agents. In spite of they essential help it to be successful in simul-

ation of very large systems, the next popular existence of the object-oriented programming (in C++ and Smalltalk) refused the life rules. Long time after, the agent orientation slowly returned, but under other titles and other metaphors taken from the world around us.

Principles of Block Orientation

True block orientation was introduced in ALGOL 60 (Backus et al. 1960), considering blocks as parts of algorithms with own entities (namely "local" variables, procedures and – for some good authors also destinations of transfers in algorithm run). In fact, such parts of algorithms, called "textual blocks", are only abstract patterns of "block instances", i.e. components of the relating computing process. The block instances are in a certain case "copies" of the textual blocks, which are able to influence the computing according to the statements occurring in the related textual blocks. In some cases (e.g. recursive calls of procedures) more instances of the same textual block exist in the same time and influence the computing process.

An essential aspect of the block orientation is block nesting: Textual block b is considered as a certain sort of statement and occurs among the other statements forming another textual block B (and so on recursively, as B can be also among the statements of another block). In such a case, b is called ***nested*** in B. The consequence of this nesting is that the statements (operations) of the nested block b can manipulate not only the entities local in it but also in the block B in that it is nested (and so on recursively).

Block nesting carries a problem of "name conflict". If an entity accessible under name e.g. N, is local in a block b nested in block B, and if an entity accessible under the same name N exists also in block B, then a question arises how to interpret name N incident in the statements of block b. The common rule, respected both in ALGOL 60 and in SIMULA 67 tells that in such a case the name N is interpreted as that of the entity introduced in the "nearest" block in the hierarchy of nesting, i.e. as that of the entity local to b. In such a case, the entity introduced under name N for block B is not accessible by the statements occurring in block b.

Algol 60 and SIMULA 67 do not allow to give names to instances of blocks. Nevertheless, according to the definition of SIMULA 67 the class declarations are viewed as analogies to blocks or – more exactly – as procedures and the instances of classes are interpreted as block instances, differing from the other block instances so that they can get names during the computing run.

Synthesis of Block Orientation with Object Orientation

The declaration of a class has many similar aspects as the declaration of a variable or a procedure and thus SIMULA 67 introduced the rule that a class can be declared as a component of another class or as a local entity of a block. The first variant can be considered as "direct" nesting of a class in another class. Let class *c* be local in a block that occurs among the statements forming life rules of another class *C*. That is an example of another class nesting, which we could call "indirect" one.

The consequence of the fact that class instances can get names is, that an instance *b* of the mentioned class *c* can get a name e.g. *R1*, and an instance *B* of class *C* can get a name e.g. *R2* different from *R1*. If a local entity in *C* is introduced under name *N*, and another local entity in *c* is introduced under the same name *N*, both the entities can be distinguished by using expressions telling "*N* of *b*" and "*N* of *B*".

Note that the concept of block is related to that of statement. In other words, a synthesis of block orientation and object orientation is fully meaningful, only when also agent orientation enters the same synthesis. Nowadays, one could mention BETA (Madsen et al. 1993) and – with a certain shutting eyes – JAVA, as certain examples of programming languages based at the synthesis of both three mentioned orientation.

SYSTEMS WITH COMPLEX AND SOPHISTICATED DECISION RULES

Internal Models

Many systems are viewed as complex not only for they are composed of many elements of many sorts differing by the rules of their mutual interactions, but also that the interactions are themselves complex even in case such an interaction may exist among few elements. Examples are systems in that one or more humans operate and make decisions that influence not only them but also their environment, which can cover the whole system.

At the present time, the human decisions are replaced (and/or modeled) by computer activities. In such a case, the capacity of computation and storing information enables the computer to analyze the instantaneous state of the given system and to optimize the system's reaction to it much better than a human being could do that. This situation leads to the task of constructing computer models of the human decisions. In principle, the models do not need to be simulation ones. Nevertheless, human often makes a decision oriented to a certain profit in future. In other words, a system with such a human "factor" is an anticipatory system according to a definition formulated by (Rosen 1985) or anticipatory system in a week sense according to a later definition by (Dubois 2000). Humans often use imagining what could happen as a sequence (or tree) of consequences of such a decision. According to the imagined events, the concerned human evaluates his/her possible decisions and chooses the optimum one.

The computer analogy of such an imagining process is a true simulation model. Therefore, if we simulate a system *S*, the dynamics of which is influenced by decisions made by element *C* that uses a model *m*, then *m* should be an instance of a class *s* nested in an instance *E* of class *c* reflecting the element *C*; and *c* should be nested in an instance *M* of a class Σ of the models of *S*. Using diagrams designed e.g. in (Mejtsky and Kindler 1980, 1981), this network of *nesting* relations is illustrated in Fig. 1, so that the relation "*x* is nested in *y*" is represented by the relation "circle-image of *x* is inside the circle-image of *y*". Note that the described nesting fully represents the access relations existing in system *S*, where *m* has access to the contents of the element *C* inside that it exists, and – having use of *C* – *m* has access to every element of system *S*. (The names of the circles that represent the instances are not in brackets, the corresponding classes are in the square brackets and the names of the corresponding modeled "things" are in the round brackets.) The circles in the dashed lines, i.e. *H1*, …*H7*, represent other elements of the model *M*, i.e.
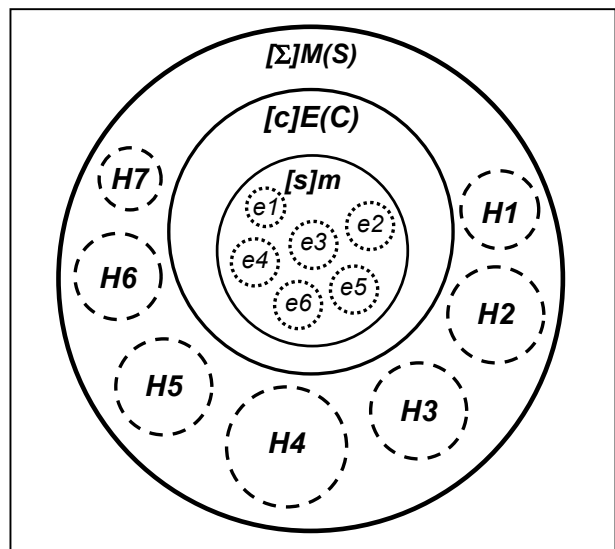
Figure 1: Mejtsky's diagram of the described case

models of components that exist in S currently with C; the small circles in dotted lines, i.e. *e1,…, e6,* represent the elements of model m, i.e. no "hard physical components" of S.

Examples

Not later than in 1990 (Kindler and Brejcha 1990), a production system containing automatically guided vehicles was simulated as follows: when a vehicle asked for its trace, it computed the shortest way by simulating a fictitious system of propagated pulse, according to an idea developed by Lee and by Dijkstra (see Dahl 1966). A similar technique was applied later, namely in simulating container yards with internal operation transport performed by ground moving vehicles (Kindler 1997). In the first case, the vehicles were viewed as really automated, while the second case started with a conception to model the intelligent decisions of human drivers, but the task of the studies was to transform human thinking to computer automating with a prospectus to have use of the computer that would control the whole system of the yard – the capacity of its memory would enable considering also the information hidden for the drivers' sights and minds. In both the cases, the simulation models reflected systems containing elements that embody a certain intelligent behaving in a labyrinth of possible paths. The intelligent decisions of the elements was based on models "privately used by the elements" and more or less reflecting the instantaneous state of their environment and the demands for future.

The last case was later completed by including anticipation of possible conflicts with other elements. The vehicles had other "privately used" models that simulated the following use of computed paths and so discovered the possible barriers that a vehicle could prepare to another one (Kindler, 2000). (Novak 2000) performed a further step in the development by synthesizing both the models in order to introduce more complex movements of the vehicles (with returns and mutual keeping away, in order to allow more complex computing shorter paths). In that way, a model of a vehicle obtained an ability to view, anticipate and improve the future not only of itself but of other vehicles as well.

The same technique was used by (Bulava 2002) for the simulation of public transport by busses in a certain North-Moravian region. The model reflected the intelligent behavior of the passengers; in general, each of them envisions several possible combinations of the buss lines to access his target, and accordingly chooses the best one. In (Bulava 2007), the same author applied a similar way for simulation of demographic development of a region. His model contains analogies of the processes of the decisions of the humans about changing their states (like residences, marital states, children, schools, employers), based on their

envisioning and/or using simulation models installed in consulting rooms.

In cooperation with French specialists in designing configurations of industrial conveyors, tailored according to the particular user demands, the simulation of "intelligent" cyclic-form conveyors was started (Kindler et al. 2004). Several simulation models were nested into the whole model of the conveyor; they served as follows:

(a) when a new parcel P comes to be served by the conveyor: to anticipate the dynamics of the parcels present at the conveyor and accordingly to decide whether P should be placed onto the conveyor immediately or later;

(b) when a parcel P was just processed at a certain working area and a choice among several working areas for the future processing of P exists: to anticipate the state of each of the plausible working areas at time when it would be accessed by P and accordingly to decide on the best target for it;

(c) in case a fault comes so that a certain working area is not accessible by the conveyor: to anticipate whether the current orders (sets of parcels) can be processed by the remaining working areas and accordingly to decide whether to continue or to stop immediately the conveyor and repair it;

(d) when the decision in case (c) is to stop immediately the conveyor: to try continuing so that the technological programs for various orders are modified, i.e. to anticipate what could happen when each of them were applied, and possibly to determine the optimum choice of new programs.

A bed ward in a hospital is a dynamic system influenced in certain states by some employees who could decide e.g. on putting together patients placed in several rooms or on the room to place a patient there (in case there are more rooms where the patient could be placed). The intelligence of such decisions could concern anticipating their possible consequences. That was included into simulation models of bed wards (Krivy and Kindler 2006) but until nowadays without particular applications.

In the last months, one makes the first steps in simulation models of reconfigurable computing networks that contain one or more computers, the function of which is to modify the links among the other computers with due regard partly for the instantaneous state of the network and partly for its possible future development (Kindler et al. 2007).

DEPENDING AND SIMULATING AGENTS AND AGENTS-MODELS

At conference Agent-Based Simulation (Urban 2000), an interesting discussion took place whether the conference title means simulation with use of agents or

simulation of agents. Since that time, the further development has shown the "agent-basing" being more complex.

Already GPSS, which represented the first paradigm of "embryonal stage" agents shown the possibility of simulation by means of agents. The computing agents were not too autonomous but they were almost "isomorphic images" of the "material agents" existing in the simulated system and in that manner they could be formalized by the GPSS users. The lack of autonomy of the agents served to support their social behaving, namely in a similar way as that of their material origins.

Let such agents be called *depending* ones. While in GPSS their social behaving was focused around common system time flow and queue discipline, the further development lead to offer the user tools for expressing other aspects and events in social behaving, like mutual exchange of data (including the case that an agent's behavior could be switched in the dependence on any component of the instantaneous states of other agents). It was developed into the most perfect way in the inspection constructs (connection statements) of the "old SIMULA" (SIMULA I), i.e. in the discrete event simulation language implemented in 1964, to that time still without object-orientation (Dahl and Nygaard 1966). The further language SIMULA (formerly called SIMULA 67) introduced – beside the object orientation – also "dot notation" (remote identifying) that allowed complete freedom in expressing interactions among the depending agents (Dahl and Nygaard, 1968).

As it was mentioned above, the block structure of the object-oriented SIMULA (i.e. SIMULA 67) allowed to formulate another type of agents, which could be called *simulating* ones. When the dynamics of a depending agent *D* enters a block with local simulated time and local class of depending agents, the block represents a simulation experiment that is to be performed while *D* operates inside the block. Thus a depending agent (to a certain model *M*) can become a simulating one, applying a model *m* that differs from *M* and exists only as some "information process" linked to *D*. When the dynamics of *D* entered the mentioned block and *D* became simulating agent, it does not cease being depending one, and when its dynamics leaves the block, *D* ceases being the simulating agent and further exists only as depending one. Naturally, the dynamics of *D* can perform e.g. a cycle and return to the block and so *D* could repeat being simulating agent. Moreover, the dynamics of *D* can contain more than one block of the given properties, and so *D* could be considered to handle different simulation models. An example is offered in the simulation of container yard presented above, where any vehicle is represented by a depending agent, which handles one model when computing the shortest path, and another model when testing it against barriers. In Fig. 1, *H1*, …, *H7* and *E* can be viewed as depending agents forming *M*, while *E*

is depicted there as being just simulating agent and handling model *m* composed of its own depending agents *e1*, …, *e6*.

The described structure can be generalized so that in the dynamics of *D* the mentioned block need not be equipped by simulated time flow and the classes local in the block need not be related to such a time flow. Then the block represents a model that is not simulation one and *D* could be called *modeling agent*.

The standard tool for simulation offered by (the object-oriented) SIMULA as its standard class called *SIMULA-TION* does not allow to identify simulation model. The reason is to provide the program product against illogical mixing of models; it is difficult to locate such a programming error when it occurs. The consequence of the restriction is that no simulation model can be named and thus handled as an object and assigned from one depending agent to another one. We discovered and implemented a class called *SIMULAT* that is not restricted in the mentioned manner (Krivy and Kindler 2007). Using this class, the model itself becomes an agent, which one can call *agent-model*. In the real world, a simple interpretation of the changing of assignments of an agent-model is changing models among different computers. In Fig. 1, the agent-model *m* is depicted as assigned to depending agent *E*. Note that a depending agent that is just simulating is behaving similarly as if an agent model would be linked to it.

CONCLUDING REMARKS

Application Inducement

As it was mentioned in section Basic Suppositions, the most suitable situation to construct a simulation model of a complex system consists in describing the simulated system and having an occasion of automatic conversion of the description into the program the run of that would realize the computer simulation experiment and – possibly – a simulation study in case the program is e.g. repeated as a subprogram in a cycle. Let us return to that idea and analyze it in details. As illustration of the analyzis let us use examples oriented to transport.

The usual process of abstraction from reality to systems about that one decides to simulate them concerns interactions of their elements, which do not directly depend of their possible individual abilities to process information in complex manner; such interactions can be viewed as "material" or "physical". In case of transport systems, their examples are moves (changing position, carrying, loading, unloading, manipulating other elements), duration of such interactions, impenetrability of mass objects, local destination, etc. They can be simply built into the life rules of transport tools and sometimes such life rules are not much more than a formalization of such interactions (namely when they

operate by use of some data sets that complete them – like the paths and the destinations). Such systems reflect parts of the reality that react to the given data and to the instantaneous environment rather bluntly, but they are often in focus of simulation, as a large system of such bluntly interacting elements (irrelevantly whether transport tools, customers in a department store, pests or molecules) may offer hard difficulties to techniques different from simulation.

To express such life rules is a natural reflection of what we view at the dynamics of the elements and what we classify according to the sort of them. Thus it is rather simple to express the life rules for different classes as algorithms composed of cycles, assignments, branchings and sending messages to other elements, and sometimes interleaved with scheduling statements. The life rules make the instances depending agents; the classes themselves are to occur out of them and may be considered as figuring at the level of the block containing the description of the whole system (i.e. representing the whole simulation system).

Now let us assume the task is to model a certain class *H* as a class of "intelligent" elements, e.g. that we have to reflect the fact that the driver of any transport tools is able to determine its path and to perform a bit a criticism of what he determined. Determining the path, he thinks and uses some concepts that do not need to reflect something of his environment but that are partially related to this environment. In system abstraction, every driver can be unified with the transport tool he drives and the rules for the life of the transport tool can be understood as those for the driver. A natural way to model the driver's thinking is to insert a block *B* at the given place of the mentioned life rules, so that the private concepts used by the driver are set as local in this block. *B* can be viewed as a good image of the driver's "state of mind", in which some concepts occur. Note that if the life rules of two drivers are in *B* simultaneously, each "life" generates its own instance of *B*, which is a good image of the fact that if two persons think in a similar way they use their own means with their "privat" interpretations of the concerned concepts. As *B* is nested in the life rules of a driver, all entities introduced for the driver's life rules are accessible in *B* and, therefore, the description of the computing inside *B* can be described also in natural way – the driver perceived his environment, determines the path and interprets it as data assigned for something that is not limited to *B*.

Let us abandon the driver's psychology and assume a computer *C* replaces it – irrelevantly whether it is in the transport tool or in some remote but accessible place. Then block *B* can represent the corresponding process of *C* (e.g. applying Lee/Dijkstra method, as mentioned above in section Examples). The technical aspects of links between the transport tool and *C* may be simply mapped by using a metaphor as if *C* was a part of the driver's mind (in system abstraction, only the

corresponding task of *C* can be so interepreted), which enables to understand the metaphor that two instances of *B* exist inside the minds of two different drivers.

The thinking of a driver can continue by a criticism whether the computed path is suitable. This represents a new phase of the driver's thinking and, therefore, it can be naturally represented as another block *B** nested in the same life rules at a place following *B*. Other concepts figure in that phase of the driver's thinking and. therefore, other classes may occur in *B**. If a computer *C* is used to make the criticism in a better way than the human mind, i.e. if is used to simulate the just computed path in order to test whether it will not lead to a conflict (see section Examples), the whole simulation model used for that purpose can be described in *B** together with the used classes. The classes and other entities introduced in *B** can be named in the same manner like those used for the description of the whole model; the reason consists in the rule for solving the name conficts (see section Principles of Block Orientation). Namely, when a name *N* is introduced for the whole model and in *B**, too, then *N* occurring inside *B** is interpreted as a name of an entity local to *B**, while *N* occurring outside *B** is interpreted as a name of an entity that coud occur in the whole model. Note that "dot notation" of SIMULA offers access to such an entity also from inside of *B** and, therefore, one can simply describe what makes a driver (possibly "amplified" by *C*), when he is constructing a copy of entity *N* of his environment so that the copy exists in his own mind and is also called *N*.

PROSPECTUS

The authors' further plans concern preparing SIMULA classes that would enable programming depending agents as completely autonomous program modules so that their environment and social behavior would be defined by means of a single pointer to "world". The tools for module compilation, for virtuality (not only of the methods but also of the targets of transfers in the dynamics) and for the branching in the inspections (Simula Standard 1989) make that idea real.

ACKNOWLEDGEMENTS

REFERENCES

Backus, J. W. et al. 1960. "Report on the Algorithmic Language ALGOL 60." *Numerische Mathematik* 2, 106-136.

Bulava, P. 2002. "Transport system in Havirov". In *Proceedings of 28th ASU Conference* (Brno, Sept. 26-30). University of Technology, Brno, 57-62.

Bulava, P. 2007. *Nested Simulation of Demographic Processes* (in Czech). Doctoral thesis, University of Ostrava, Ostrava.

Buxton, J. N. (Ed.). 1968. *Simulation Programming Languages Proceedings of the IFIP working conference on simulation programming languages, Oslo, May 1967.* North-Holland, Amsterdam 1968.

Dahl, O.-J. 1966. *Discrete Event Simulation Languages*. Norwegian Computing Center, Oslo. Reprinted in (Genuys 1968), 349-395.

Dahl, O.-J.; and K. Nygaard. 1966. "SIMULA – an ALGOL-based Simulation Language". *CACM* 9, No. 9 (Sept.), 671-678

Dahl, O.-J.; and K. Nygaard. 1968. "Class and Subclass Declarations." In (Buxton, 1968), 158-174.

Dubois, D. 2000. "Review of Incursive, Hyperincursive and Anticipatory Systems – Foundation of Anticipation in Electromagnetism." In *CASYS'99 – Third International Conference,* D. Dubois (Ed.). The American Institute of Physics, Melville, New York, 3-30.

Genuys, F. (Ed.). 1968. *Programming Languages*. Academic Press, London – New York.

Gordon, G. 1961. "A General Purpose Simulation Program." In *Proc. 1961 EJCC*. MacMillan, New York, 81-98.

Kalisz, E. and A. M. Florea. 2000. "A GPSS simulation model of interactions in a market-based multi-agent system." In: (Urban, 2000), 145-150.

Kindler, E. 1997. "Classes for object-oriented Simulation of Container Terminals". In *Managing and Controlling Growing Harbour Terminals*, E.Blümel (Ed.). The Society for Computer Simulation International, San Diego, Erlangen, Ghent, Budapest, 175-278.

Kindler, E. 2000. "Nesting Simulation of a Container Terminal Operating With its own Simulation Model". *JORBEL (Belgian Journal of Operations Research, Statistics and Computer Sciences)*, 40, No.3-4, 169-181

Kindler, E. and M. Brejcha. 1990. "An application of main class nesting – Lee's algorithm." *SIMULA Newsletter*, 13, No. 3 (Nov.), 24-26

Kindler, E.; T. Coudert; and P. Berruet. 2004. "Component-Based Simulation for a Reconfiguration Study of Transitic Systems," *SIMULATION* 80, No. 3 (March), 153-163.

Kindler, E.; C. Klimes; and I. Krivy. 2007. "Simulation Study With Deep Block Structuring". In *Modelling and Simulation of Systems MOSIS 07,* J. Stefan (ed.). MARQ, Ostrava, 26-33.

Krivy, I. and E. Kindler. 2006. "Reflective Simulation of In-Patients Dynamics." In *5th International Conference Aplimat* (Bratislava Febr. 7-12). Part 1. Slovak University of Technology, Bratislava, 613-617.

Krivy, I. and E. Kindler. 2007. "New Flexible Simulation Tool in SIMULA." In *ESM'2007 – The 2007 European Simulation and Modelling Conference Modelling and Simulation '2007* (St. Julian's, Malta, Oct. 22-24). EUROSIS-ETI, Ghent, Belgium, 124-128.

Madsen, O. L.; B. Møller-Pedersen; and K. Nygaard. 1993. *Object-Oriented Programming in the Beta Programming Language*. Addison Wesley, Harlow – Reading – Menlo Park.

Mejtsky, J.; and E. Kindler. 1980. "Diagrams for quasi-parallel sequencing – Part I". *SIMULA Newsletter* 8, no. 3 (Nov.), 46-49

Mejtsky, J.; and E. Kindler. 1981. "Diagrams for quasi-parallel sequencing – Part II". *SIMULA Newsletter* 9, no. 1 (Feb.), 17-19

Novak, P. 2000. **"**Reflective Simulation with Simula and Java". In *Simulation und Visualisation 2000*, T. Schulze, V. Hinz, P. Lorenz (Eds.). The Society for Computer Simulation International European Publishing House, Ghent, 183-196.

Rosen, R. 1985: *Anticipatory Systems*. Pergamon Press, New York.

Simula Standard. 1989. SIMULA a.s., Oslo.

Urban, Ch. (Ed.). 2000. *Agent-Based Simulation. Proceeding of Workshop 2000, Passau, Germany, May 2000*. The Society for Computer Simulation International, San Diego.

AUTHOR BIOGRAPHIES

EUGENE KINDLER was born in 1935 in Prague (Czechoslovak Republic). He studied mathematics at Charles University in Prague and there he got grades of Doctor of philosophy in Logic and Doctor of sciences in theory of programming. The Czechoslovak academy of sciences granted him the grade of Candidate of sciences in physics/mathematics. During his employment in the Prague Research Institute of Mathematical Machines (1958-1966), he participated at the design of the first Czechoslovak electronic computer and designed and implemented the first Czechoslovak ALGOL compiler for it. Then, working at the Ïnstitute of Biophysics at the Faculty of General Medicine of Charles University (1967-1973), he designed and implemented the first Czechoslovak simulation language and then introduced the object-oriented programming into Czechoslovakia. Nowadays, as professor emeritus of applied mathematics, he teams up with University in Ostrava. He was visiting professor at the University in Italian Pisa, at the University of South Brittany in French Lorient, at Blaise Pascal University in French Clermont-Ferrand, and at West Virginia University in American Morgantown. His main interest consists in object-oriented simulation of systems that handle complex models to improve their own anticipatory abilities and intelligence.

IVAN KRIVY graduated at the Czech Technical University (1962) and the Charles University (1974). In 1975 he obtained the degree Ph.D. in solid state physics. He worked 13 years at the Nuclear Research Institute (1962-1975) and later (1975-1977) at the Computer Art Establishment as a system programmer. Since 1977 he was teaching courses in programming, probability theory and statistics at the Pedagogical Faculty at Ostrava. After the University of Ostrava having been established in 1991, he entered the Mathematics Department of the Faculty of Science and nowadays he works at the Computer Science Department as Professor in applied mathematics. Since 1992 his research activities are oriented to dynamic modelling and computer simulation as well as to stochastic algorithms and their use in the global optimization. Krivy is a member of the American Mathematical Society, the Association of SIMULA Users, and the International Association for Statistical Computing. He is author of about 190 publications including 4 monographs. Stays abroad: Silesian University of Katowice, Portsmouth Polytechnic, University of Wroclaw, Universidad de Cordoba, Lulea University of Technology and Blaise Pascal University of Clermont-Ferrand.

A TRASS-BASED AGENT MODEL FOR TRAFFIC SIMULATION

Ulf Lotzmann
Michael Möhring
Institute of Information Systems Research
University of Koblenz
Universitätsstraße 1, Koblenz 56070, Germany
E-mail: {ulf, moeh}@uni-koblenz.de

KEYWORDS

Simulation framework, traffic simulation, multi-agent based model, normative agents.

ABSTRACT

This paper focuses on traffic simulation generated by the multi-agent simulation framework TRASS (Traffic Simulation System). TRASS-based simulations conceive agents as individual traffic participants moving in an artificial environment and which can be described in a three-layer agent model. There is no restriction on any particular kind of road users such as cars or pedestrians, but a concept is presented which is appropriate to model different kinds of traffic participants and have them interact with each other in one single scenario which may not only include roads, but also stadiums, shopping malls and other situations where pedestrians or vehicles of any kind move around.
Simulation results will be presented by using TRASS as a testbed for analyzing norm emerging processes, ported in the context of a simple traffic scenario (i.e. learn traffic rules by observation and experience).

INTRODUCTION

The design of a simulation model particularly in terms of agent design is affected by multiple factors. An important one is, without doubt, the selection of the simulation software to be used. For almost every serious simulation project, a ready-made, perfect fitting and mature software is not available. In fact a modeller has to search for the most satisfying compromise and will be likely to face three alternatives to choose among:

- The usage of a domain-specific tool. This implies that the simulation model can be adapted to the specifications of the tool. An example from the traffic simulation domain is SUMO, a software system for the microscopic simulation of road traffic, applying a microscopic car-following model (Krajzewicz *et al.*, 2006).

- The usage of a general-purpose simulation tool. In this case most likely a lot of (programming) work has to be done in order to match the tool with the model's requirements. Some important agent-based frameworks are Repast, Mason, Netlogo and Swarm. Evaluations of these tools were published by Railsback *et al.* (2006) and Tobias & Hofmann (2004). In Oliveira & Bazzan (2006) a project that deals with the coordination of traffic light using the Swarm platform is presented.

- The development of the "optimal fitting" software "from scratch".

In this paper we present the multi-agent simulation framework TRASS (Traffic Simulation System). Although the name suggests a domain specific concept, the underlying design makes it applicable for universal agent-based simulation scenarios, where the term "agent" refers to the definition given in Gilbert & Troitzsch (2005). The associated software system was implemented using JAVA with the aim to introduce novel properties like parallel execution incorporating multi-threading techniques and a continuous space. Thus, the approach relies on the third item in the list above.

Furthermore, this paper shows an approach for a broad range of traffic simulation applications. It aims at discussing a concept where agents represent human actors, "playing" different roles of traffic participants in an artificial spatial environment.

The underlying agent model defines a set of different layers for the representation of physical, technical and mental aspects. The physical layer characterizes the appearance of an agent within the environment – defined by a topography model – and comprises properties like shape, facilities for perception and attributes of motion. The technical layer determines the actions an agent is able to perform on the environment by modifying the physical properties. How to use these technical actions strategically in order to achieve a desired behaviour or a predefined goal is the concern for the mental layer.

While the TRASS framework specifies the topography model and the characteristics of the agent physics, appropriate methods have to be employed for the technical and the mental layer. In our implementation, the former is called "robotic layer" as it uses the technique of the finite state machine that is utilised frequently in association with autonomous robots. The latter is called "AI layer" and incorporates methods originating from artificial intelligence research.

To demonstrate how this approach can be applied, an example on simulating the emergence of norms in a simple traffic scenario completes the paper.

THE TRASS AGENT MODEL

The usage of agent-based simulation in the social sciences as a method for description, understanding or prediction of social phenomena has become more and more important over the last two decades. The agents involved in the majority of simulation scenarios represent restricted (and sometimes simple) models of human individuals. Although these agent models cover only a few aspects of humans and are driven by minimal sets of explicit goals, the simulation outcomes in terms of emergent structures reproducing complex coherences of real societies seem very impressing and promising. While the typical domains of social simulation generally refer to market analysis and policy, it seems reasonable to transfer the concept into other fields, in particular traffic simulation.

An agent model describing a traffic participant has to consider further aspects besides the deliberation capabilities. Especially the complex interaction between a traffic participant and the environment is of importance. In other words, the AI potential needs a fundament of physical and technical capabilities.

We structure the different aspects of our agent model for traffic participants in three distinct layers: the physical layer, the robotic layer and the AI layer (Figure 1).
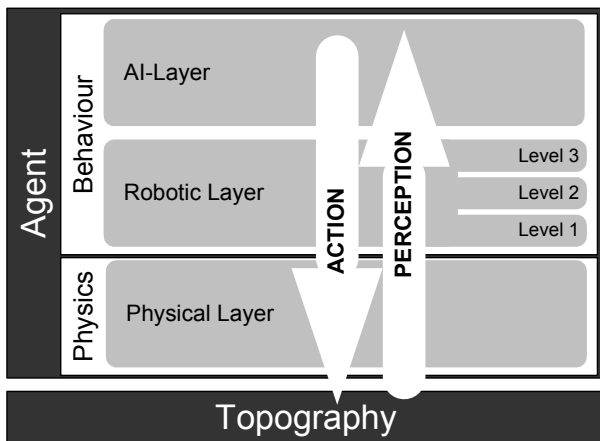
Figure 1: Layers of the TRASS agent model

Physical Layer

The physical layer describes the abstraction of the static properties and the parameters of motion. Thus it defines the physical appearance of the agent within the artificial environment.

This layer constitutes the interface to the spatial environment defined by the topography model. It is

obvious that the physical layer and the topography model must be coordinated. For this reason both model descriptions are part of the TRASS core framework.

The topography model is a two-dimensional continuous space that can be split into districts and that is structured by a mesh of polygon-shaped regions. Each region describes one distinct area of the topography, such as a lane of a road section, a crossing or building (see Figure 4). The corresponding physical layer of the agent model is composed of five elements, which are in detail:

- The shape of the agent is constituted by an arbitrary number of circles (with different radii) approximating the outline of the entity to be represented. The usage of circles is due to performance issues when executing necessary calculations during simulation runs (e.g. distance between agents).

- The sensor unit is constituted by an arbitrary number of (various) circle sectors, defining the zones where an agent is able to perceive the environment.

- The communication area is similar to the sensor unit in terms of the geometrical shape involved. It is used to define zones in which an agent can appear for other (observing) agents in different manners. For example, a traffic light agent sends its information (red, yellow or green light) in one distinct direction, affecting only car agents arriving from one distinct way. For any other agent the traffic light is just an obstacle.

- The reference point defines the current position in the two-dimensional world.

- Attributes of motion are velocity and direction of motion.

Robotic Layer

The robotic layer fills the gap between the "low-level" attributes of the physical layer and the abstract strategic decisions treated by the AI layer. Thus, the design of the robotic layer is of utmost importance for the entire system.

The name "robotic layer" is referring to the methods that are used here to implement the transformation process: As in many autonomous robot systems, a form of enhanced finite state machine (Hopcroft & Ullman, 1979) controls the parameter changes depending on the active state and triggered by incidents perceived in the environment (resulting in reactive behaviour) or generated by the AI layer (resulting in proactive behaviour).

The design of the finite state machine concept used in the robotic layer is inspired by an evolutionary classification model of human needs, originated by Max-Neef (1992) and translated into an agent model for

market simulation by Norris & Jager (2004). In this model, a distinction is made between

- survival needs, controlled by the brain stem and cerebellum and resulting in automated decisions (reptile mode);

- social needs, controlled by the limbic system based on social heuristics (mammal mode);

- identity needs, controlled by the neo cortex via individual heuristics (primate mode).

In the robotic layer a related concept of different "levels of mind" is used where each level corresponds with a stage in a nested hierarchy of automata. The following levels are considered here:

- basic actions which humans execute "automatically" without thinking (e.g. turning the steering wheel);

- elementary activities (composed of basic actions) which are conducted intuitively by humans (e.g. hold the centre of a lane);

- complex activities as sequences of elementary activities where a human's attention is required (e.g. lane change operation).

Each level is represented by a self-contained automaton that operates on the instance below and is controlled by the instance above. These instances can be other automata within the hierarchy, except the bottom-level and the top-level automata. While the former executes parameter changes on the agents' physical layer, the administration of the latter is done by the AI layer.

The administration of an automaton is done by the substitution of state transitions. A state transition is a function that receives the active automata state and the current perception of the environment as input and calculates the successive state (including parameters for configuration of state activities) as output.

The perceptive input of the transition function is generated by the agent sensor. At each automata level perception filter units are attached for categorizing and condensing the perceived objects.

The execution of a state transition is triggered by events:

- every time-step event for the bottom-level automaton (since we use a time discrete simulation concept);

- otherwise, the trigger event for a transition on level n is generated within the transition function on level n-1.

Figure 2 gives an overview on the entire concept for the robotic layer.
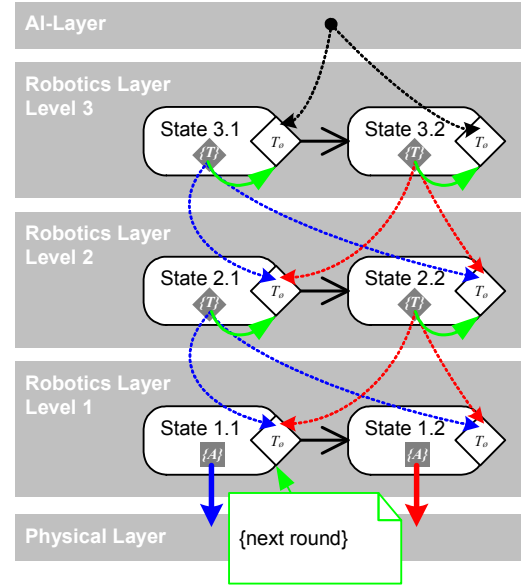
Figure 2: Levels of the robotic layer. $T_\emptyset$ is the abbreviation for "transition template", $\{T\}$ and $\{A\}$ indicate sets of transitions and actions, respectively.

AI Layer

In order to show "intelligent" – and thus realistic – behaviour, the AI part of an agent model for traffic participants requires a control process that is comprised of the following activities:

- Perceive the environment containing geographical elements and neighbouring agents.

- Compose and memorize an agent-individual view of the simulation world based on the perceived environment.

- Communicate with a subset of the agent population.

- Reason about different alternatives to act in order to achieve predetermined goals, considering the current environmental state and the agent memory.

- Execute "physical" action based on the capabilities offered by the robotic layer (e.g. driving a curve or changing a lane).

- Estimate the success of action, leading to revaluation of agent memory.

While most of the issues mentioned in the list above are of a more or less technical nature in terms of processing and memorizing perception data and adjusting of parameters, the subject of "reasoning about alternatives" constitutes the actual AI capabilities.

Such a reasoning process can be expressed in a function where the input parameters contain quantified information about the present state of the perceived environment and the agent-internal memory, and the

output is a set of probabilities for the alternatives to choose.

Since in most cases it seems possible to split a set of three or more options to choose between into the nested execution of different functions, each designed for two concurring alternatives, an approach originated in the social sciences is applicable, namely the Weidlich-Haag opinion forming model (Weidlich & Haag, 1983).

The original version of this simple model includes two levels. At the macro level the current ratio of (the subset of) agents which have chosen alternative a in the past (P_a), to agents relying on alternative b (P_b) is calculated embracing the entire population of agents P, and stored within the real-valued variable x:

$$x = \frac{|P_a| - |P_b|}{|P_a| + |P_b|}$$

The range of x is within [-1, 1], where the minimum stands for "all agents rely on alternative a", the maximum for "all agents rely on alternative b", respectively.

At the micro level each agent "decides" (based on the macro variable x) in every time step whether to change or keep its actual attitude. This is done by calculating the probability for an opinion change. The probability function depends on the present attitude and reads for alternative b as

$$\mu = \nu \exp(\pi + \kappa x)$$

and analogously for alternative a:

$$\mu = \nu \exp(-(\pi + \kappa x))$$

The meaning of the parameters involved is as follows:

- ν – flexibility (higher values increase the probability for attitude changes);
- π – preference (for alternative a when > 0, for alternative b when < 0);
- κ – coupling (higher values increase the influence of variable x).

Basis for the attitude change decision is the comparison of the calculated probability μ with a uniform random number m (in the range [0; 1]).

We use this model in a couple of traffic oriented simulation scenarios one of which is presented in this paper later on.

Besides the Weidlich-Haag model exemplarily described above, the adoption of various other approaches from artificial intelligence research is imaginable as basis for the AI layer.

THE TRASS SOFTWARE

The agent model presented in the previous section was developed with and on the basis of the TRASS

framework. While the robotic and AI layers of the agent model are built upon the TRASS software in terms of extensions, the topography model and the physical layer are determined by the software design of the TRASS core. The following list gives an overview on the key properties of the TRASS core:

- continuous space (with a full set of geometric algorithms and tools to handle corresponding topographies);
- discrete time (controlled by a timer component);
- multi-threaded architecture resulting in parallel simulation execution on multi-processor machines;
- message-based inter-agent communication with single and group recipients, enabling the agents to exchange symbolic messages;
- mediator-based communication infrastructure fosters distributed execution.

The TRASS core does not bring any visualization capabilities or graphical user interface. It defines an programming interface that allows for the integration of external components. The interface consists of sections for:

- definition of agent types with specific behaviour by writing Java code;
- XML based configuration of simulations;
- access to the internal data structure of the simulation kernel for collecting attribute values while executing a simulation;
- administration of simulation runs.

In order to offer a complete and concerted interface for user interaction, an Integrated Development Environment has been developed. This software incorporates functionality like topography designer, agent modeller as well as a runtime environment for simulation execution and visualization. Figure 3 shows the component structure of the TRASS system.
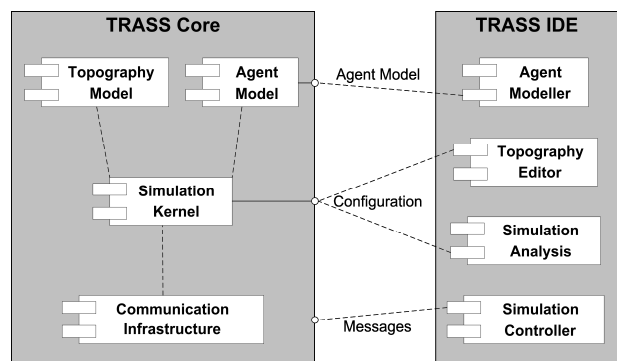
Figure 3: Component diagram of TRASS

EMERGENCE OF NORMS IN A TRAFFIC SCENARIO

The development of concrete model examples plays an important part in the validation process of the TRASS concept and the embedded three-layer agent architecture described in the preceding chapters. This includes typical simulation models of traffic flows in a real topography based on empirical data as well as using TRASS as a *testbed* for the analysis of (more general) dynamic processes between interacting agents, ported in the context of a traffic scenario. The model example presented here follows the latter by introducing simple normative agents, who are able to learn traffic rules by observation and experience in a simple traffic scenario provided by TRASS and save them as explicitly internalized norms.

The EMIL Project

The development of this simulation model is part of the EMIL (EMergence In the Loop: Simulating the two-way dynamics of norm innovation) project funded by the EU (FP6) in the framework of the initiative "Simulating Emergent properties in Complex Systems". This project is especially focussed on modelling norm innovation processes in societies by introducing intelligent agents on different levels, which allows both modelling and observing the emergence of properties at a macro-level and their immergence into the micro-level (Castelfranchi, 1998). Furthermore, the development of a simulator for exploring and experimenting norm-innovation is part of the EMIL project (Andrighetto *et al.*, 2007).

Model Framework

Following the described modelling concept of TRASS, the static structure of a TRASS model requires a topography and the description of model agents.

Topography

The topography consists of a single *one-way road* heading from "north" to "south", possibly embedded in a simple road network.
On both sides of the lane *rectangular areas* exclusively *allowed for pedestrians* are joined, which may represent playgrounds as well as shopping zones. The three edges of each pedestrian zone that are not aligned to the road are bordered by regions marked as "wall".
The road features one to three (configurable) sections marked as *crosswalks*.

Agents

In a first version, the scenario includes the following types of agents:

- *Pedestrians* represent humans with the capability to walk with random direction and velocity, but are also able to head for a distinct position (some

"point of attraction"). They are able to recognize and (if desired) to avoid collisions with other pedestrians and cars, and to change from one pedestrian zone to another by crossing the road (using a crosswalk or not).

- *Cars* are container agents which define the physical properties of (real) cars and are controlled by driver agents (one per car). *Drivers* represent humans in the role of car drivers. They are (technically) capable at least to drive on a road with variable velocity, to keep a safety distance to a car driving ahead, to stop in front of or to run over pedestrians, respectively, to recognise collisions, to observe agents (pedestrians) walking close to the road and to distinguish between "normal" road sections and crosswalks
(Figure 4 shows the described traffic scenario including pedestrians and car/drivers with their defined perception area).

- One *supervisor* observes the complete scenario, keeps track of the (normative) behaviour of pedestrians and reacts to it (e.g. hands out sanctions)

Future versions might involve further types of agents:

- *Sign posts* to communicate deontics (e.g. permissions, obligations, forbearances) like "crosswalk ahead" for example.

- *Observers* ("cops") to replace part of the supervisor agent's job by an internal model agent
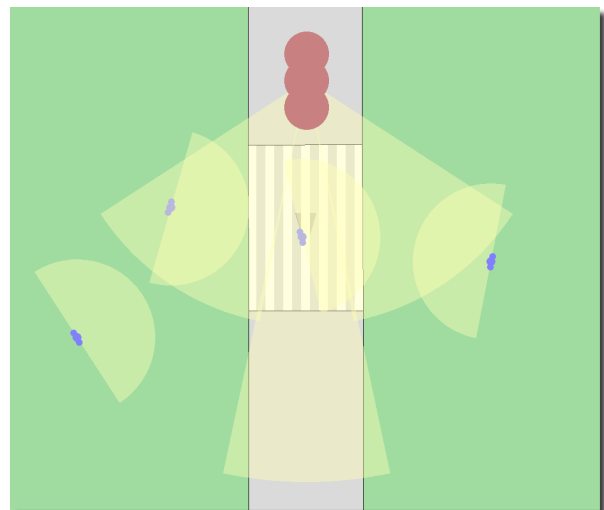
Figure 4: Screenshot of a running simulation. A car is waiting at a crosswalk while a pedestrian is crossing the road and three other pedestrians are approaching at the crosswalk. The circular sectors in front of the agents are visualizations of the perception sensors.

Model Dynamics

The dynamics of the simulation is constituted by the pedestrian agents, car driver agents and the supervisor.

Agents

- *Pedestrian*: *Initially* the pedestrians walk within the pedestrian zones randomly. From time to time a desire to reach some target within the opposite pedestrian zone arises within a pedestrian. This results in goal-oriented movement towards this target, either on the shortest path or including a detour to one of the crosswalks. In any case the road traffic is ignored. This leads to incidental collisions with cars (with the effect of sanctions for both the pedestrian and the car driver involved, depending on using a crosswalk or not).

- *Car Driver*: *Initially* the car drivers steer their cars on the road trying to avoid crashes with other cars, but ignore pedestrians. This leads to incidental collisions with pedestrians (with the effect of sanctions for both the pedestrian and the car driver involved, depending on using a crosswalk or not).

During simulation the decisions of both *pedestrians* and *car drivers* are influenced by sanctions they got and by observing the behaviour of other traffic participants (via *supervisor*). Basically, the dynamics between the micro level (individual agents) and the macro level (agent populations) follow the Weidlich/Haag approach, described in section "AI Layer".

Speaking in EMIL terms, each agent includes an ability called *norm recognizer* (CNR-ISTC, 2007), in which each agent checks whether the individual decision would comply or violate a decision, more or less accepted by the whole agent community.

Supervisor

The supervisor keeps track of the (normative) behaviour of pedestrians and reacts on the following incidents:

- Collision between pedestrian and car within a crosswalk region leads to a *hard* sanction (sanctions are scaled between 0 and 1) against the car driver. In more advanced model versions a sanction against the pedestrian regarding additional parameters (e.g. car velocity, velocity of pedestrian as gauge for attention) might be reasonable as well.

- Collision between pedestrian and car outside crosswalk regions leads to an *average* sanction against the car driver and the pedestrian as well with a grade depending on the agents velocity.

- Driving with undamped velocity within a crosswalk region and pedestrians waiting for crossing the road at the same time leads to a *mild* sanction against the car driver.

Therefore, a supervisor acts as an (normative) authority with the ability to sanction defective behaviour (according to predefined or emerged norms) of pedestrians and car drivers. This includes also the conversion from a widely accepted behaviour (*pre-norm*), emerged from the interactions between *pedestrians* and *car drivers*, into an official prescribed behaviour standard (*norm*):

When the supervisor agent notices that a number of agents beyond a threshold comply with a pre-norm, the pre-norm is converted into a norm by the supervisor. This results in harder sanctions against norm-violating behaviour of pedestrian and driver agents.

Preliminary Findings

As a result of the interactions between *pedestrians* and *car drivers* some stable behaviour of agents, which represent (emerged) *pre-norms/norms*, is observable (e.g.):

- *Pedestrians*:

  "Use crosswalks to cross the road instead of walking the shortest/direct path!"

- *Car Drivers:*

  "Do not run over pedestrians even when they do not use a crosswalk!"

  "Slow down and stop before crosswalks when pedestrians are waiting!"

Although, the development of this simple traffic scenario is in a very early state and more work on this model is necessary, experience with the modelling process and also first simulation experiments show that the universal multi-agent framework TRASS combined with the described layer-based agent modelling concept, supports a fast and structured development of simulation models in the context of a traffic scenario (e.g. description of modeller-defined robotics and AI capabilities built upon already provided agent capability on the physical layer).

Additionally, this layer-based agent modelling concept explicitly supports a step-by-step modelling strategy starting with very simple agent architectures and interactions. With reference to the introduced model example, the developed prototype and first simulation experiments support a reliable (and promising) estimation how much work to invest in modelling more complex norm emergence/immergence processes (e.g. agent architecture with */norm recognition, norm adoption, decision making/* modules) described in the EMIL proposal.

CONCLUSION

In this chapter, an approach to traffic simulation is presented that incorporates an agent model built upon mental and cognitive processes of humans. These

properties in combination with the overall agent autonomy allow for interaction between agents representing human traffic participants in multiple roles within an artificial environment.

The distinct aspects of the agent model are represented by different layers:

- Physical layer for agent appearance within the topography model,

- Robotic layer covering technical aspects of agent behaviour and

- AI layer for representing mental properties of human traffic participants.

This agent model allows for the usage within a wide range of different application fields. Besides the pure research-oriented scenario presented in this paper, several ongoing projects related to practice make use of the TRASS concept. Subject of one important project is the simulation of traffic flows within an urban area in maximum load situations, where different types of traffic participants have to be considered. This simulation – developed in cooperation with the local city administration – is set up in the context of traffic planning activities within a major reconstruction project in the city centre of Koblenz, Germany.

REFERENCES

Andrighetto, G., Conte, R., Turrini, P., & Paolucci, M. 2007. Emergence In the Loop: Simulating the two way dynamics of norm innovation. *In: Dagstuhl Seminar on Normative Multi-agent Systems, Dagstuhl, Germany*.

Castelfranchi, C. 1998. Simulating with cognitive agents. *In:* Sichman, J.S., Conte, R., & Gilbert, N. (eds), *Multi-Agent Systems and Agent-Based Simulation*. Springer, Berlin.

CNR-ISTC. 2007. *Deliverable 1.2 EMIL-M: Report of models of norms emergence, norms immergence and the 2-way dynamic*. Tech. rept.

Gilbert, Nigel, & Troitzsch, Klaus G. 2005. *Simulation for the Social Scientist. 2nd edition*. Open University Press, McGraw-Hill, Maidenhead.

Hopcroft, John, & Ullman, Jeffrey. 1979. *Introduction to Automata Theory, Languages and Computation*. 1st edn. Addison-Wesley.

Krajzewicz, D., Bonert, M., & Wagner, P. 2006. The Open Source Traffic Simulation Package SUMO. *In: RoboCup 2006 Infrastructure Simulation Competition, RoboCup, Bremen*.

Max-Neef, M. 1992. Development and human needs. *In:* Ekins, P., & Max-Neef, M. (eds), *Real-life economics: Understanding wealth creation*. Routledge, London, New York.

Norris, Gregory A., & Jager, Wander. 2004 (October). Household-Level Modeling for Sustainable Consumption. *In: Third International Workshop on Sustainable Consumption, Tokyo*.

Oliveira, D., & Bazzan, A. L. C. 2006. Traffic Lights Control with Adaptive Group Formation Based on Swarm Intelligence. *In: 5th International Workshop on Ant Colony Optimization and Swarm Intelligence, ANTS, Brussels*.

Railsback, S. F., Lytinen, S. L., & Jackson, S. K. 2006. Agent-based Simulation Platforms: Review and Development Recommendations. *SIMULATION*, **82**(9), 609–623.

Tobias, R., & Hofmann, C. 2004. Evaluation of free Java-libraries for social-scientific agent based simulation. *Journal of Artificial Societies and Social Simulation*, **7**(1).

Weidlich, W., & Haag, G. 1983. Concepts and Models of a Quantitative Sociology. The Dynamics of Interacting Populations. *Springer Series in Synergetics*, **14**.

AUTHOR BIOGRAPHIES

ULF LOTZMANN obtained his diploma degree in Computer Science from the University of Koblenz-Landau in 2006. Already during his studies he has participated in development of several simulation tools. Since 2005 he has specialized in agent-based systems mainly within the domain of traffic simulation and is developer of the TRASS system. He is also involved in several other recent projects of the research group. His e-mail address is ulf@uni-koblenz.de.

Dr. MICHAEL MÖHRING received his first and his PhD degree from the computer science faculty of the University of Koblenz-Landau and has worked in the research group for about 20 years. He developed a multi-level simulation tool called MIMOSE and participated in all the research projects of the group, currently specialising in both multi-agent simulation and in data mining; he has been teaching both information society and data mining for more than a decade. His e-mail address is moeh@uni-koblenz.de.

SIMULATING THE EFFECT OF THERAPIES AND TRAUMAS ON EMOTION REGULATION

Matthijs Pontier

Vrije Universiteit, Department of Artificial Intelligence
De Boelelaan 1081, 1081 HV Amsterdam, The Netherlands

Center for Advanced Media Research Amsterdam
Buitenveldertselaan 3, 1082 VA Amsterdam, The Netherlands

E-Mail: mpr210@few.vu.nl

KEYWORD

Agent-Based Modelling, Emotion Regulation

ABSTRACT

Previous research suggests that therapies can help people to regulate their emotions, and on the other hand, traumas can cause impaired emotion regulation. This paper introduces a computational model for simulating the effects of events like traumas or therapies on emotion regulation. This model is an extension of an existing computational model of emotion regulation (Bosse et al. 2007c). A number of simulation experiments has been performed and evaluated.

INTRODUCTION

In science, emotions were historically seen as neural activation states without a function (Hebb 1949). However, recent research indicates that emotions are functional. Emotions have a facilitating function in decision making (Oatley and Johnson-Laird 1987), prepare a person for rapid motor responses (Frijda 1986), and provide information regarding the ongoing match between organism and environment (Schwarz and Clore 1983). Emotions also have a social function. They provide us information about others' behavioural intentions, and script our social behaviour (Gross 1998).

In the past two decades, psychological research has started to focus more on *emotion regulation* (e.g., Gross, 1998; 1999; 2001; 2002; Thompson 1994), i.e., the process humans undertake to increase, maintain or decrease their emotional response.

Gross has described an informal process model for emotion regulation (Gross 2001). Based on this model, Bosse et al. (2007b; 2007c) have introduced a computational model of emotion regulation. In this model, the personal flexibility in changing emotion regulation behaviour, based on meta-cognition about the emotion regulation process, is fixed. In real life, however, this personal tendency to adjust behaviour can be changed by certain events. For instance, if someone has a very low tendency to change his behaviour in order to regulate his emotions, a therapy could help that person to change this tendency, and help him learn to adapt this regulation of emotions in a more flexible manner. Previous research suggests that therapies can help people to regulate their emotions. For instance,

Beck and Fernandez describe, based on 50 studies, that people who were treated with a cognitive behavioural therapy as an approach for anger management were better off than 76% of untreated subjects, in terms of anger reduction (Beck and Fernandez 1998). In addition, an article by Burns et al. suggests that a structured anger management training programme is useful for forensic inpatients with learning disability (Burns et al. 2003). Finally, Deschner and McNeil describe an experiment, in which families that experienced violence, followed anger control training. After the training, 85% of the families were free of further violence and remained so, according to an independent survey completed 6-8 months later (Deschner and McNeal 1986). On the other hand, an event like a trauma could decrease the tendency to change one's behaviour in order to regulate his emotions significantly. Schore describes that an early trauma can cause impaired affect regulation (Schore 2001).

This article presents an extension of the emotion regulation model by Bosse et al. (2007c) that makes it able to simulate events that influence the emotion regulation process, like traumas or therapies. Simulation experiments have been performed to test whether events with a positive effect on the personal tendency to change behaviour in favour of emotion regulation, like therapies, facilitate emotion regulation, and events with a negative effect on the tendency to change behaviour, like traumas, impair emotion regulation.

This model could be used for several purposes. For example, from a Cognitive Science perspective, a model that simulates emotion regulation could give insight in the process of emotion regulation. An advanced model could also be helpful to make predictions about emotions, about behaviour that is a consequence of emotions, and about how to influence certain behaviours with, e.g., an anger management therapy. This could be useful in for instance work with forensic inpatients. In addition, a model for emotion regulation can be used in the field of Artificial Intelligence, see e.g. (Bates 1994). For example, in the virtual reality domain it can be used to let virtual agents show human-like emotion regulation behaviour. Similarly, in the gaming industry, there is much interest in manners to let game characters emotionally behave like humans.

EMOTION REGULATION

Gross has informally described a process model of emotion regulation (Gross 2001). According to this model, people can use strategies to influence their level of emotion at different time points. Gross mainly differentiates between *antecedent-focused strategies* and *response-focused strategies*. Antecedent-focused strategies are applied before response tendencies have become fully activated. These are strategies to prevent a person from becoming too emotional, or too little emotional. Response-focused strategies are applied when an emotion is already under way, and response tendencies have been generated. These strategies focus on the way emotions are expressed, and are mainly driven by social factors.

In his model, Gross distinguishes five different emotion regulation strategies, which can be applied at five different time points. Four of them are antecedent-focused strategies: situation selection, situation modification, attentional deployment and cognitive change. The fifth one, response modulation, is a response-focused strategy. All strategies are shown in Figure 1.
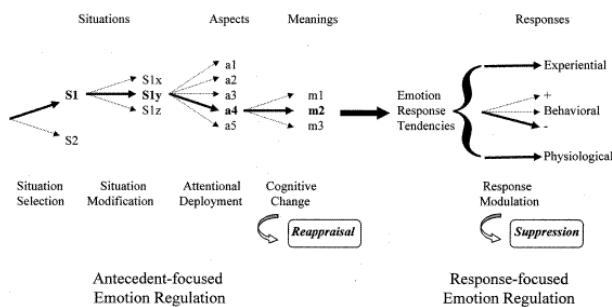
Figure 1: Process Model of Emotion Regulation by Gross (2001)

The first emotion regulation strategy in the model is *situation selection*. When this strategy is applied, a person chooses to be in a situation that matches the emotional level he wants to experience for a certain emotion. For example, a person could refuse an invitation to a party, because he is in a conflict with someone who is going to that party, and he would feel angry when seeing him. This is an example of down-regulating one's emotion. Deciding to make a bungee jump is an example in which the level of emotion is raised.

The second emotion regulation strategy in the model is *situation modification*. When this strategy is applied, a person modifies a situation so that it better fits to a person's optimal level of emotion. For instance, when a person is watching a television interview with a very annoying political leader, he might zap to another channel.

The third emotion regulation strategy is *attentional deployment*. This strategy refers to shifting your attention to a certain aspect. An example of attentional deployment is when a person is watching an important

soccer game that ends in a penalty shoot-out, and he chooses to put his hands before his eyes, instead of watching the final penalty, because it is too exciting.

The fourth emotion regulation strategy is *cognitive change*. This is a strategy in which one selects a cognitive meaning to attach to an event. An example of cognitive change is when a team loses a soccer match, and a player blames this on his teammates, instead of on his own play.

The fifth emotion regulation strategy, *response modulation*, is a response-focused strategy, and is applied after the emotion response tendencies have been generated. When a person applies response modulation, it tries to influence the process of response tendencies becoming a wanted behavioural response. An example of response modulation is hiding that you are nervous when giving a presentation.

Cognitive-Behavioral Therapies have the purpose to facilitate beneficial use of emotion regulation strategies. They focus on cognitive aspects, as well as behavioural aspects. The behavioural part focuses on replacing counterproductive emotional driven behaviours with alternatives. This has a facilitating effect on beneficial use of situation selection, situation modification, and attentional deployment. The cognitive part focuses on substituting irrational negative appraisals for evidence-based appraisals. This has a facilitating effect on beneficial use of cognitive change (Campbell-Sills & Barlow 2006).

MODELLING APPROACH

Modelling the various aspects involved in emotion regulation in an integrated manner poses some challenges. On the one hand, qualitative aspects have to be addressed, such as decisions to regulate one's emotion (e.g., by selecting a different situation). On the other hand, quantitative aspects have to be addressed, such as levels of emotional response. The modelling approach based on the modelling language LEADSTO (Bosse et al. 2007a) fulfils these needs. It integrates qualitative, logical aspects such as used in approaches based on temporal logic (e.g., Barringer et al. 1996) with quantitative, numerical aspects such as used in Dynamical Systems Theory (e.g., Ashby 1960; Port and Gelder 1995).

In LEADSTO, direct temporal dependencies between state properties in two successive states are modelled by executable dynamic properties defined as follows. Let a and b be state properties of the form "conjunction of literals" (where a literal is an atom or the negation of an atom), and e, f, g, h non-negative real numbers. Then in the *leads to* language $a \rightarrow\!\!\!\rightarrow_{e, f, g, h} b$, means:

If state property a holds for a certain time interval with duration g, then after some delay (between e and f) state property b will hold for a certain time interval of length h.

Here, atomic state properties can have a qualitative, logical format, such as an expression desire(d), expressing that desire d occurs, or a quantitative, numerical format such as has_value(x, v) expressing that variable x has value v.

OVERVIEW OF THE SIMULATION MODEL

This model is an extension of a previous model (Bosse et al. 2007c). This section gives an overview of this model. For convenience, the model concentrates on one specific type of emotion. In principle, this can be any emotion that is considered to be a basic human emotion, e.g., sadness, happiness, or anger (Ekman et al. 1972).

In order to be able to formalise Gross' model, for any given type of emotion a number of variables have been introduced. We have chosen to express the *emotion response level* ERL in a real number, in the domain [0, 2]. A higher emotion response level means a person feels more emotion. Humans are always aiming for an optimal level of emotion. The location of this optimum depends interpersonally. For instance, there are people who enjoy extreme sports, or roller coasters, while others prefer a more quiet kind of recreation. The optimal level of emotion also depends on the type of emotion. Most people have a relatively high optimal level of emotion for happiness, while they have a lower optimal level of emotion for fear. In the model, there is a fixed optimal level of emotion ERL$_{norm}$, also expressed in a real number in the domain [0, 2].

In the model of Gross, five different strategies for five different elements can influence the emotion response level. Because in the experiments of Gross (Gross 2001) the strategy response modulation had no effect on the emotion experience, it is not considered in this paper. However, the model is constructed in such a way that the set of strategies can easily be adapted. The four remaining strategies that are considered and their corresponding elements are shown in Table 1.

Table 1: The Four Considered Emotion Regulation Strategies and their Corresponding Elements

Strategy	Element
Situation selection	situation
Situation modification	subsituation
Attentional deployment	aspect
Cognitive change	meaning

In the real world, a person can always choose between a limited number of options for each strategy. Due to simplification, this model assumes that at every time point, for each element, a person makes a certain choice, which has a certain *emotional value* v$_n$ attached. In the computational model, we also express the emotional values of the four different elements in real numbers in the domain [0, 2]. This emotional value contributes to the emotion response level via an element-specific weight factor w$_n$, thereby taking into account a

persistency factor β, indicating the degree of persistence or slowness of adjusting the emotion response level when new emotional values are obtained. Someone whose emotions can change very rapidly (e.g., who stops being angry in a few seconds) will have a very low β.

The regulation process compares the actual emotion response level ERL to the emotion response level aimed at ERL$_{norm}$. The difference d between the two is the basis for adjustment of the choices made for each of the elements n; based on these adjusted choices, each element n will have an adjusted emotional value v$_n$. The strength of such an adjustment is expressed by a *modification factor* α$_n$, which represents the flexibility to change one's emotional value for a certain element.

In order to obtain a model that can adapt itself to various circumstances, the modification factors α$_n$ have been made adaptable. The flexibility to choose different emotional values v$_n$ can be adapted to an assessment of the emotion regulation process: a sort of reflection or meta-cognition about the emotion regulation process based on the history of differences d. The *adaptation factor* γ$_n$ mediating in this adaptation process represents the personal flexibility to adjust the emotion regulation behaviour based on such an assessment. It takes some effort to change behaviour in favour of emotion regulation. This effort, or the costs of adjusting the modification factor for element n, is represented by c$_n$.

The model is shown in a qualitative manner in the graph depicted in Figure 2. The variables above the dashed line are added in this paper, and will be explained in the next section. Note that the model contains two cycles. One is the basic emotion regulation cycle from the v$_n$ to ERL via d back to the v$_n$. The other one is the adaptation cycle from the α$_n$ to the basic regulation cycle and back (via v$_n$, ERL and d back to α$_n$).
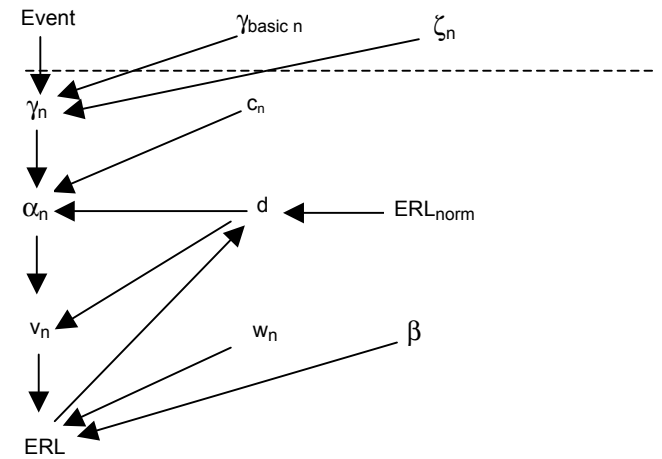
Figure 2: Dependencies between the Variables.

The processes modelled here are in real life continuous. Since simulation needs some form of discretisation, this model works with a fixed step size s that can be taken any size as desired.

ADDING THE POSSIBILITY TO SIMULATE EVENTS LIKE TRAUMA'S AND THERAPIES

In order to simulate events that can change the personal tendency to adjust behaviour in favour of emotion regulation γ, we have chosen to express these events in real numbers in the domain [-1, 1]. If an event has a high value, for instance a successful therapy, it will lead to a higher tendency to adjust behaviour in favour of emotion regulation. If the value gets closer to 0, it will have a smaller effect, and when it reaches 0 it will have no effect at all. An event with a negative value, for instance a trauma, will result in a lower tendency to adjust emotion regulation behaviour. The following formula is used to let events influence the tendency to adjust behaviour in favour of emotion regulation:

$$\Delta\gamma_n = \zeta_n * Event / (1 + (\gamma_n - \gamma_{basic\ n}) * Event) * \Delta t$$
$$new\_\gamma_n = old\_\gamma_n + \Delta\gamma_n$$

These formulas are represented by the following LEADSTO properties:

LP7 (Keep old gamma)
gamma(gamma)
and not change_gamma
$\rightarrow_{0,\ 0,\ s,\ s}$ gamma(gamma)

LP8 (Change gamma)
experience(exp)
and gamma(gamma)
$\rightarrow_{0,\ 0,\ s,\ s}$ gamma(gamma + zeta * exp / (1 + (gamma-gamma_basic) * exp))

In these formulas, new_γ_n is the new personal tendency γ_n, and old_γ_n is the old personal tendency γ_n. $\Delta\gamma_n$ is the change of γ_n. The new γ_n is derived by adding $\Delta\gamma_n$ to the old γ_n. The variable 'Δt' is the time step, which is taken 1 in this paper. ζ_n is a variable that determines the speed with which the personal tendencies are adjusted by events. In simulation experiments, a ζ_n in the range 0.10 – 0.20 seemed to be most realistic. Event is the value that is attributed to a particular event that is simulated in the model. $\gamma_{basic\ n}$ is a person's basic personal tendency to change its behaviour in favour of emotion regulation. Assumed is that a person is born with a basic personal tendency to change behaviour in favour of emotion regulation, and this personal tendency can be changed by events. However, when the γ_n deviates more from $\gamma_{basic\ n}$, and an event influences γ_n to deviate even more from $\gamma_{basic\ n}$, γ_n will be changed less than when it is influenced by an event with the same strength in the different direction. So for instance, when a person has a very low $\gamma_{basic\ n}$, but a series of events made the γ_n rise to a much higher level, an Event with the value 0.5 will make the γ_n raise only a little bit more, while an Event with the value -0.5 will make the γ_n decrease significantly. In other words, events can change a person's personal tendency, but it gets harder when the personal tendency has already changed much. Table 2 shows a summary of all the treated variables. The variables depend on each other as shown in a qualitative manner in the graph depicted in Figure 2.

Table 2: Summary of All Variables

Variable	Meaning
ERL	Level of emotion
ERL_{norm}	Optimal level of emotion
d	Difference between ERL and ERL_{norm}
β	Slowness of adjustment ERL
w_n	Weight of element n in adjusting the ERL
v_n	Chosen emotional value for element n
α_n	Modification factor that represents the 'willingness' to change the emotional value of element n
γ_n	Personal tendency to adjust the emotional value of element n much or little
$\gamma_{basic\ n}$	Basic personal tendency to adjust the emotional value of element n much or little
c_n	Costs of adjusting emotional value v_n
Event	Value of an event that reflects the impact it has on personal tendency γ
ζ_n	Variable that determines the speed with which events influence personal tendencies
Δt	Time step

SIMULATION EXPERIMENTS

A number of experiments have been performed on an agent, to test whether the model can simulate the effects of events like traumas or therapies on emotion regulation. The variables that were used in the experiments are summarized in Table 3. The values of the fixed variables and the initial values of the remaining variables are shown.

Table 3: A Summary of the Variables that Have the Same Value for All Experiments

Variable	Value	Fixed / Initial
ERL_{norm}	0.5	Fixed
β	0.7	Fixed
ERL	1.8	Initial
v_1–v_4	1.8	Initial
w_1	0.35	Fixed
w_2	0.30	Fixed
w_3	0.20	Fixed
α_1 - α_4	0.01	Initial
c_1	0.7	Fixed
c_2	0.4	Fixed
c_3	0.4	Fixed
c_4	0.6	Fixed
γ_1 - γ_4	0.01	Initial
$\gamma_{basic\ 1}$ - $\gamma_{basic\ 4}$	0.05	Fixed
ζ_1 - ζ_4	0.15	Fixed

The experiments in this paper simulate the behaviour of an individual. The simulated emotion is anger, and the agent's optimal level of anger is 0.5. The agent starts with a very high emotion response level of 1.8, and very high emotional values, all set to the same level of 1.8. So at the start of the simulation, our agent is very angry, and is in a situation that keeps him angry. The weights, and the costs attached to the various elements are set to

the same values as in the previous model (Bosse et al. 2007c). The γ_n's, which represent the personal tendency to change behaviour in favour of emotion regulation, are initially set to 0.01, which is a very low value. The value for all $\gamma_{basic\ n}$'s is set to 0.05. This is somewhat lower than average, which means that by nature the agent has a relatively low personal tendency to change behaviour in favour of emotion regulation. The initial γ_n is even lower, which means that before the start of the simulation, the agent has had some experiences, for example a trauma, which decreased its personal tendency to change behaviour in favour of emotion regulation. The ζ_n are all set to 0.15. Experiments showed that this is a normal value. In the experiments in this chapter, the manipulated variables are the events that influence the personal tendency to change behaviour in favour of emotion regulation.

Experiment 1: The effect of a successful anger management therapy

In this simulation, we let our agent experience an event that will increase its personal tendency to change behaviour in favour of emotion regulation very much: a successful cognitive-behavioural anger management therapy. This event takes place at time point 40, and has the value of 0.9 in the domain [-1, 1]. The results of this simulation can be seen in Figure 3. Because all emotional values and modification factors show similar behaviour, and the only difference is made by the costs, only the graphs of the element with the highest costs, situation, and of one of the elements with the lowest costs, subsituation, are shown.
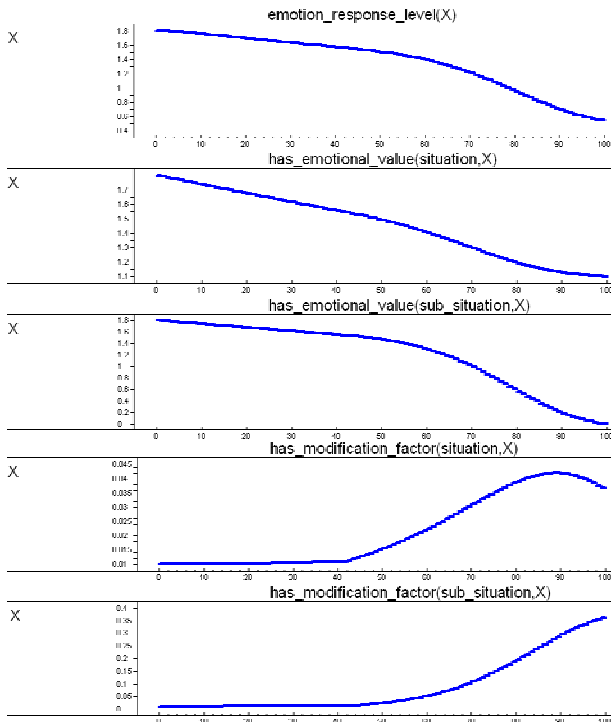
Figure 3: Simulation of the Emotion Response Level, the Emotional Values and the Modification Factors in Experiment 1

As can be seen in Figure 3, the emotion response level first decreases very slowly. Later in the simulation, after the therapy has taken place, the emotion response level starts to descend more quickly, and at the end of the simulation the optimal level of emotion is reached. So at the beginning of the simulation, the agent is not able to let its level of anger decrease to its optimal level, but at the end of the simulation it is. This can also be seen in the simulations of the emotional values. First these values decrease very slowly, and later in the simulation they decrease much quicker. This is the clearest in the emotional values of elements with lower costs, such as subsituation. These emotional values decrease much more than the emotional values of elements with higher costs. At the end of the simulation, the emotional values of elements with lower costs have almost reached 0, while the emotional value of situation selection, with higher costs, has decreased only until 1.1.

The modification factors α_n increase very slowly at the beginning of the simulation, and start to increase more quickly after the therapy has taken place. The modification factors of elements with lower costs increase much quicker than the elements with higher costs. At the end of the simulation, the modification factors of elements with lower costs have increased until 0.37, while the modification factor of situation selection, with higher costs, has increased only until 0.042, and has started to decrease again.

It makes sense that the modification factors increase very slowly at first, and start to increase quicker at a later time point in the simulation. At time point 40, an event takes place, which makes the personal tendency to change behaviour in favour of emotion regulation, represented by the γ_n, rise from 0.01 to 0.15. These γ_n have a direct effect on the modification factors, as can be seen in Figure 2. Especially in the simulation of the modification factor of situation selection, the impact this has on the modification factors can be seen. A few steps after time point 40, the modification factor starts to increase much quicker. So after the anger management therapy, it immediately starts to increase its willingness to change its behaviour in favour of emotion regulation.

The impact this has on the emotional values can be seen very clearly. After time point 40, the emotional values start to decrease much quicker. After a while, the emotion response value has decreased enough to make the emotional value for situation selection decrease more slowly again. So our agent is not able to reach its optimal level of emotion by choosing different situations, cognitive meanings, etc. in the first part of the simulation, but after the anger management therapy at time point 40, it starts to change its behaviour, and at the end of the simulation it has reached its optimal level of emotion, and is able to keep it stable.

Experiment 2: The effect of a series of events

In this simulation, we let our agent experience various events that change its personal tendency to change behaviour in favour of emotion regulation. First, at time point 20, the agent experiences an event that is just as

'strong' as the event at time point 40 in experiment 1. So the same agent follows the same cognitive-behavioural anger management therapy with the same amount of success as in experiment 1, only now it already takes place at time point 20. At time point 40, the agent experiences an event that is just as strong as the event at time point 20, but now in the opposite direction, so that it will decrease its personal tendency to change behaviour in favour of emotion regulation. In real life, this event could be for instance a severe traumatic experience. At time point 60, the agent experiences a positive event with the strength 0.4, for instance a therapy that helps it deal with the traumatic event he experienced at time point 40. Finally, at time point 80, the agent experiences a negative event with the strength of 0.3.

Table 4: A Summary of the Events in Experiment 2

Event at time point 20	0.9
Event at time point 40	-0.9
Event at time point 60	0.4
Event at time point 80	-0.3

The results of this simulation can be seen in Figure 4. Again, because all emotional values and modification factors show similar behaviour, and the only difference is made by the costs, only the graphs of the element with the highest costs, situation, and of one of the elements with the lowest costs, subsituation, are shown.
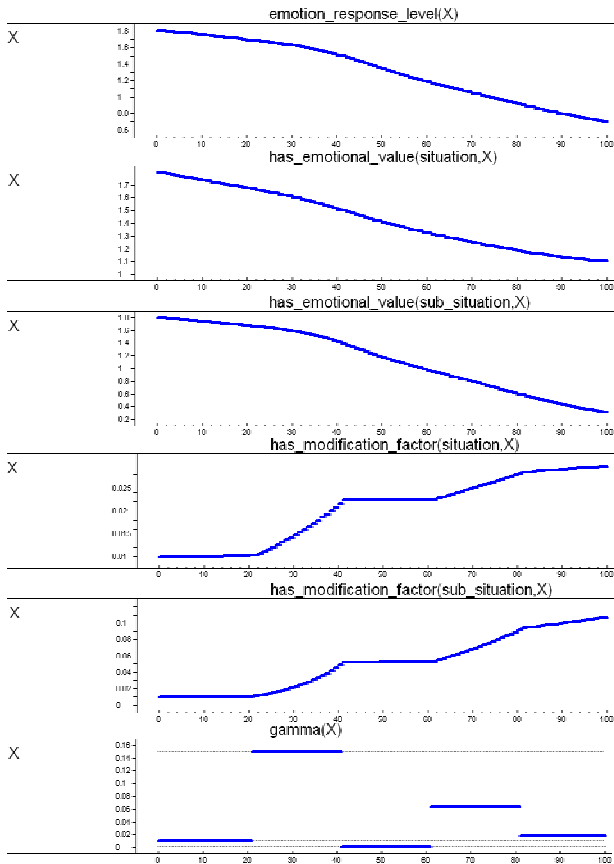
Figure 4: Simulation of the Emotion Response Level, the Emotional Values, the Modification Factors, and the γ_n in Experiment 2

As can be seen in Figure 4, the emotion response level decreases very slowly at the start of the simulation. After the anger management at time point 20, which makes the γ_n rise to 0.15, the emotion response level starts to decrease somewhat quicker. Because the traumatic event at time point 40 makes the γ_n decrease to an even lower level than it was at the beginning of the simulation, this decreasing trend does not proceed the way it did in experiment 1. The effects of the less powerful events at time point 60 and 80 can clearly be seen by the kinks in the graphs of the modification factors, but the effects on the emotional values and the emotion response level are less clear. It can clearly be seen that the less powerful events have a smaller impact on the γ_n.

DISCUSSION

In this paper, a computational model for simulating the effects of events like traumas or therapies on emotion regulation has been described. This model is an extension of the model by Bosse et al. (2007b; 2007c), and was inspired by the process model of emotion regulation as informally described by Gross (Gross 2001).

The simulation experiments have demonstrated that the model can simulate the effects of events that influence the personal tendency to change behaviour in favour of emotion regulation, like trauma's, or therapies. In experiment 1, the agent has at first a low tendency to change its behaviour, and is because of this low tendency not able to reach its optimal level of emotion. After an anger management therapy, its personal tendency to change behaviour in favour of emotion-regulation has increased, and the agent is able to regulate its emotions, and reach its optimal level of emotion. In experiment 2, a series of events influence the agent's emotion regulation. In this experiment, the relatively 'stronger' events have a bigger impact on the emotion regulation process. These results are consistent with the literature (e.g., Beck and Fernandez 1998; Deschner and McNeal 1986). Validation involving extensive comparison with detailed empirical data is left for future work.

Possible improvements to this model could be introducing decay for the γ_n, which represent the personal tendency to change behaviour in favour of emotion regulation. This way, the γ_n would slowly return to the value of $\gamma_{basic\ n}$ if no events occur that influence the γ_n. Also, in the formulas that are used to calculate the new emotional values, the emotion response level could be modified by a random factor. The size of this random factor could be changed in order to simulate emotion regulation in people that are not able to recognize their level of emotion very well. Furthermore, in this model, Gross' notion of response modulation is not considered, and thus has no effect on the emotion regulation at all. It could be the case that response modulation at the long term, for instance the suppression of a traumatic experience, could have an

effect on emotion regulation. The model could be changed in order to make it able to simulate this. Finally, in the current model, a trauma cannot cause a relapse in the emotion regulation process. It can only slow down the process. The model could be changed so that a trauma would not only influence the personal tendency to change behaviour in favour of emotion regulation, but also the willingness to change behaviour in favour of emotion regulation, and the chosen emotional values.

REFERENCES

Aarts, E., Harwig, R., and Schuurmans, M. 2001. "Ambient Intelligence." In *The Invisible Future: The Seamless Integration of Technology into Everyday Life*, McGraw-Hill, 2001.

Ashby, R. 1960. *Design for a Brain*. Second Edition. Chapman &Hall, London. First edition 1952.

Barringer, H., Fisher, M., Gabbay, D., Owens, R., and Reynolds, M. 1996. *The Imperative Future: Principles of Executable Temporal Logic*, John Wiley & Sons, 1996.

Bates, J. 1994. "The role of emotion in believable agents." *Communications of the ACM*, Vol. 37, No. 7, pp. 122-125.

Beck, R., and Fernandez, E. 1998. "Cognitive-Behavioral Therapy in the Treatment of Anger: A Meta-Analysis." *Cognitive Therapy and Research*, 1998, Vol. 22, No. 1, pp. 63-74.

Bosse, T., Jonker, C. M., Meij, L. van der, Sharpanskykh, A., and Treur, J. 2006. "Specification and Verification of Dynamics in Cognitive Agent Models." In *Proceedings of the Sixth International Conference on Intelligent Agent Technology, IAT'06*. IEEE Computer Society Press, 2006, pp. 247-254.

Bosse, T., Jonker, C.M., Meij, L. van der, and Treur, J. 2007a. "A Language and Environment for Analysis of Dynamics by Simulation." *International Journal of Artificial Intelligence Tools*. Vol. 16, 3, 2007, pp. 435-464.

Bosse, T., Pontier, M., and Treur, J. 2007b. "A Computational Model for Adaptive Emotion Regulation." In Lin, T.Y., Bradshaw, J.M., Klusch, M., Zhang, C., Broder, A., and Ho, H. (Eds.) *Proceedings of the Sixth IEEE/WIC/ACM International Conference on Intelligent Agent Technology, IAT'07*. IEEE Computer Society Press, 2007, pp. 289-293.

Bosse, T., Pontier, M., and Treur, J. 2007c. "A Dynamical System Modelling Approach to Gross´ Model of Emotion Regulation." In Lewis, R.L., Polk, T.A., and Laird, J.E. (Eds.) *Proceedings of the 8th International Conference on Cognitive Modeling, ICCM'07*. Taylor and Francis, 2007, pp. 187-192.

Burns, M., Bird, D., Leach, C., and Higgins, K. 2003. "Anger management training: the effects of a structured programme on the self-reported anger experience of forensic inpatients with learning disability." *Journal of Psychiatric and Mental Health Nursing*, 2003, Vol. 10, pp. 569–577.

Campbell-Sills, L. and Barlow, D. H. 2006. "Incorporating Emotion Regulation into Conceptualizations and Treatments of Anxiety and Mood Disorders" In Gross, J. J. (Ed.) *Handbook of Emotion Regulation*, Guilford press, New York

Deschner, J. P., and McNeil, J. S. 1986. "Results of Anger Control Training for Battering Couples." *Journal of Family Violence*, 1986, Vol. 1, No. 2, pp. 111-120.

Eisenberg, N. 2000. "Emotion, regulation, and moral development." *Annual Reviews of Psychology*, 2000, Vol. 51, pp. 665–697.

Ekman, P., Friesen, W.V., and Ellsworth, P. 1972. *Emotion in the human face: guidelines for research and integration of Findings*. New York: Pergamon.

Frijda, N. H. 1986. *The emotions*. Cambridge, England: Cambridge University Press.

Goldsmith, H. H., and Davidson, R. J. 2004. "Disambiguating the Components of Emotion Regulation." *Child Development*, March/April 2004, Vol. 75, No. 2, pp. 361-365.

Gross, J. J. 1998. "The Emerging Field of Emotion Regulation: An Integrative Review." *Review of General Psychology*, Vol. 2, No. 3, pp. 271-299.

Gross, J. J. 1999. "Emotion Regulation: Past, Present, Future." *Cognition and emotion*, 1999, Vol. 13, No. 5, pp. 551-573.

Gross, J. J. 2001. "Emotion Regulation in Adulthood: Timing is Everything." *Current directions in psychological science*, Vol. 10, No. 6, pp. 214-219.

Gross, J. J. 2002. "Emotion regulation: Affective, cognitive, and social consequences." *Psychophysiology*, 2002, Vol. 39, pp. 281–291

Hebb, D. O. 1949. *The organization of behavior: A neuropsychological theory*. New York: Wiley.

Oatley, K., and Johnson-Laird, P. N. 1987. "Towards a cognitive theory of emotions." *Cognition and Emotion*, Vol. 1, pp. 29-50.

Port, R.F., and Gelder, T. van (Eds.) 1995. *Mind as Motion: Explorations in the Dynamics of Cognition*. MIT Press, Cambridge, Mass.

Schore, A. N. 2001. "The effects of early relational trauma on right brain development, affect regulation, and infant mental health." *Infant mental health journal*, 2001, Vol. 22, No. 1, pp. 201-269

Schwarz, N., and Clore, G. L. 1983. "Mood, misattribution, and judgments of well-being: Informative and directive functions of affective states." *Journal of Personality and Social Psychology*, Vol. 45, pp. 513-523.

Thayer, J.F., and Lane, R.D. 2000. "A model of neurovisceral integration in emotion regulation and dysregulation." *Journal of Affective Disorders*, Vol. 61, pp. 201-216.

Thompson, R.A. 1994. "Emotion regulation: A theme in search of definition." In N.A. Fox (Ed.) *The development of emotion regulation: Biological and behavioral aspects. Monographs of the Society for Research in Child Development*, Vol. 59 (Serial No. 240), pp. 25-52.

AUTHOR BIOGRAPHY

MATTHIJS PONTIER was born in Amsterdam, The Netherlands, and went to the VU University of Amsterdam. He studied Artificial Intelligence as well as (Cognitive) Psychology, and obtained his Masters degree in the direction of Cognitive Science in 2007. Currently, he is employed as a PhD student at the VU University of Amsterdam, both at the department of Artificial Intelligence and the Center for Advanced Media Research Amsterdam. His e-mail address is: `mpr210@few.vu.nl`.

PROPOSING A FRAMEWORK FOR BUILDING HUMAN–BASED AGENTS FOR DISCRETE EVENT SIMULATION:
A CASE STUDY IN SCREENING PROGRAMS

Nasim GhanbarTehrani and Mohammad Aghdasi

Tarbiat Modares University

Industrial Engineering Department

Tehran - Iran P.O.Box:14115-111

Email: nasim.tehrani@modares.ac.ir

KEY WORDS

Simulating human behavior, Human-Based Agents, Reference Model, Health Belief Model (HBM)

ABSTRACT

By knowing more about the special characteristics of human behavior in complex environment, especially understanding the communication processes, social interactions and their role in forming human behaviors, the need for capturing these attributes in problem solving techniques will be increased. The ability of implementing complex computational models, calls for a trade-off between the complexity of representing personal attributes from one side, and the complexity of model structure from the other side. One of the main ideas for solving this problem is using object-oriented techniques in order to create Human-Based Agents, and substitute these agents as the human representatives in the simulation.

In this article, some of the main theories that describe the human behavior – especially decision making in the health–care context, are explained and a simulation model of the implementation of the screening process is structured.

The case refers to a real situation of blood–pressure screening in Iranian villages. For this purpose, a reference framework and the related mathematical equation are developed in two categories: internal structure of the agents and the external structure, that is the environment that agents live and behave in. Finally, the model is simulated by using a Delphi-based computer program.

1-INTRODUCTION

Simulation is a suitable tool for problem solving in the different contexts, researches and evaluation of new theories and also training. The ability of simulation practices in building and developing simplified replicas of the real world has been appreciated by many researchers. As the main objective of the most health care processes is serving customers, there are lots of benefits for the use of simulation in this context. (Standbridge, 1999). Most problems in this area are related to the human behavior (Brailsford, 2003). As an example, in most epidemic models, not only biologic and physiologic factors are important, but also behavioral elements (such as tendency to do risk) are really dominants. Prevention and dermatological models in most disease contain the factors that affect the decision makings about using drugs and doing the prescribed treatments. In screening programs, the factors affecting the patient's acceptance are really critical.

In the most simulation models, human behaviors encountered by the model are limited to the statistical data gathered from the real world experiments. Obviously, this could not be done in every case. In the proposed framework, human behavior is produced by the human-based agents; due to the external factors affect their behavior. A simple schema of these approaches is shown in figure 1.
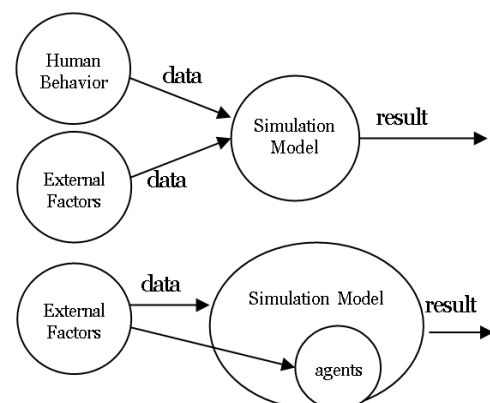
Figure 1: Comparing Simulation Models, with and without Human-Based Agents

The disease patterns are usually different with what observed in the society means the major part of the iceberg is hidden and just special cases (in terms of the patient's accessibility, the severity of disease and etc.) are recorded. Screening programs are proposed to estimate the real situation of infection. As the costs of implementing these overall programs are really high, there should be a great focus on planning and designing phases of these programs in order to gain more acceptance, as the result of the program is dictated by the human behaviors. In this research, a screening program is selected as a basis for developing the context of simulation. A descriptive scheme of screening program is shown in figure 2.

and robbing the supermarkets. (Klugl, 98).Urban uses this method for simulating group learning (Urban, 2000), forming supportive networks, Prad and Predator model (Urban, 2001)

Sibbel implies a model for Hospital management (Sibbel,2001). Schmidt uses an agent named Adam for investigating the instinctive behaviors (Schmidt, 2000), and screening program. (Ohler & Roger)

Some mathematical structures are designed for simulating human behaviors, such as PECS(Schmidth,2000), and BDI (Pokahr et al, 2003)

Figure 2: Screening Process

Human beings commonly are a part of complex systems. For describing these systems in details, they should be modeled by the same details as non-human parts.

A human has the ability to behave diversely in a social context. As different Physical, cognitive, sensitive, psychological, cultural, historical and social factors affect it; the description of human behavior is really complicated. When considering the complexity of the environment together with the complexity of the behavior of people with interaction together, the modeling seems to be impossible.

In the most problem–solving models, human and human characteristics are modeled by adding some extra variables, constraints, figures or simple entities. (Klugl, 98). Searching the literature revealed that the more detailed organizational structures, results in the more simple modeled humans. (Moss,nd)

The Human Agents Applications in simulation are extending their context (Urban, 2000). In recent years, agents have been had a critical role in modeling and simulation. (Schmidt, 2000). Agents have been used for modeling social structures in organization by Moss (Moss, nd). Klugl uses SeSAm software for simulating social behavior of Ants and bees, and also for simulating some human behaviors such as fire fighting

Many Human behavior models have been developed in different disciplines. Some researchers use simulation models in conjunction with these descriptive models for examining new theories and understanding the existing ones.

The applicability of simulation in analyzing Health care systems has been experimented many times. The simulation models have been used for patient scheduling and other problems in operations management in hospitals. (Fether & Thompson, 98) (Robinson, 95) (Wing & David, 98)

Simulation systems used in Health care, are very similar to the Manufacturing simulation models.

The main problem in these systems is the ignorance of the human behavior – which has a great effect on the system performance. Physicians, nurses and other personnel decide upon their experiences and personalities and change the process by their decisions. The personal attributes of patients affects their caring and working Processes. (Sibbel and Urban, 2001)

2-DESCRIPTIVE HUMAN BEHAVIOR MODELS IN HEALTH CARE-Literature Review

Human behavior models are generally derived from the behavioral science and Psychology. They provide a

framework for prediction, policy evaluation and help decision makers to describe the reason of human behavior. Based on the literature review performed by authors, the most used models in the health care are Ajzen's planned behavior theory, HBM models and HLC model.

Although these models have been developed widely, they have limited ability to understand, predict and control human behavior. The purpose of these behavioral models is to describe why people do/ don't do health-related behaviors (such as using drugs, doing the prescribed behavior, stop smoking and etc.). They also explain the relationship between the encountered variables. These relations could be evaluated by various tests and the new variables and relationships could be added respectively.

2-1-HBM Model

HBM (Health Belief Model) is one of the most applicable conceptual frameworks in health-related behaviors. Proposed in 1950s, HBM has a successful career in promoting health care behaviors (like using seat belt, medical acceptance, participation in the screening programs, and etc.). A schematic view of this model is described in figure 3.
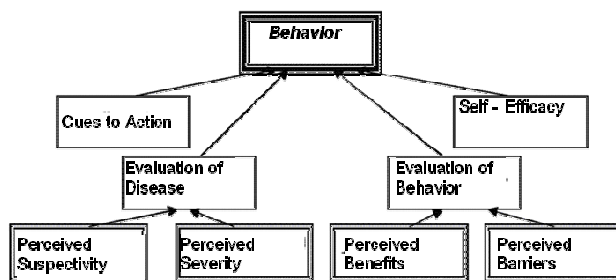
Figure 3: HBM Model

A brief definition of main concepts is given in table 1.

Table 1: Main concepts of HBM

Concept	Definition
Perceived susceptibility	One's belief of the changes of getting a condition
Perceived severity	One's belief of how serious a condition and its consequences are
Perceived Benefits	One's belief in the efficacy of the advised action to reduce risk or seriousness of impact
Perceived Barriers	One's belief in the tangible and psychological costs of the advised behavior
Cues to Action	Strategies to activate readiness
Self - Efficacy	Confidence in one's ability to take action

This model is really simple and reasonable, and the main concepts could be understood by non-expert users. This model has been used widely, and has high validity.

2-2-Stages of Change Transtheorical Model

This theory has been developed by Caro Diclemente and James O. Prochaska in 1979. They proposed the change of human behavior (changing undesired to desired behavior) has general stages that could be formulated in a model. In this theory, people are modeled by their decision making process. This theory claims that the change is a process consisting of many general stages. Change, means doing the right thing at the same time. These stages of change have been conceptualized for a variety of problem behaviors:

- Pre-contemplation ; is the stage at which there is no intention to change behavior in the foreseeable future (about 6 months)

- Contemplation is the stage in which people are aware that a problem exists and seriously think about overcoming it but have not yet made a commitment to take action.

- Preparation is the stage that combines intention and behavioral criteria. Individuals in this stage are intending to take action in the next month and have unsuccessfully take action in the past year.

- Action is the stage in which individuals modify their behavior, experiences or environment in order to overcome their problem.

- Maintenance is the stage in which people work to prevent relapse and consolidate the gains attained during action.

- Termination: change has been occurred and the new behavior is now one of the ordinary procedures of the person.

2-3-Social Learning Theory (SLT) & Health Locus of Control (HLC)

Social learning theory focuses on the learning that occurs within a social context. It considers that people learn behaviors from each other. They observe other's behavior and the outcomes of these behaviors, and could learn to accept or reject these behaviors. Learning may occur, but the behavior could be fixed. Although Behaviorist imply that learning means changing the behavior, SLT theorists insist on learning may not necessarily be shown in the behavior. They also focus on "cognition" as a key issue in learning. Awareness and expectations of future reinforcements/punishments may have a major effect on this concept.

Health locus control (HLC) has been developed on SLT concept. HLC has a set of statements that could

measure the belief of Individuals to control their behaviors internally and externally; these statements are evaluated by scores and the sum of these scores determines the level of person's health belief. Due to many problems in prediction, this theory has been developed, and multi–dimensional HLC introduced by Wallston and Kaplan. In this model, by standardizing measures, researches could repeat their survey, and compare the results.

2-4-Theory of Reasoned Action (TRA) / Theory of Planned Behavior (TPB)

TRA developed in 1967 by Martin Fishbein and Icek Ajzen. It has been revised in 1970s, and has been used for studying human behavior and design of experiments since 1980s. In 1988, the concept of TPB has been added to TRA to overcome its inefficiencies.

This theory is developed to predict and understand the effect of motivations on behaviors, analyzing the strategies for changing behavior and embedding human behavior in virtual environments. A schematic view of this model is shown in figure 4. In this theory, the main determinant of human behavior is behavior intention. Intention is a combination of Individual's attitude to implementation of behavior and mental norms.
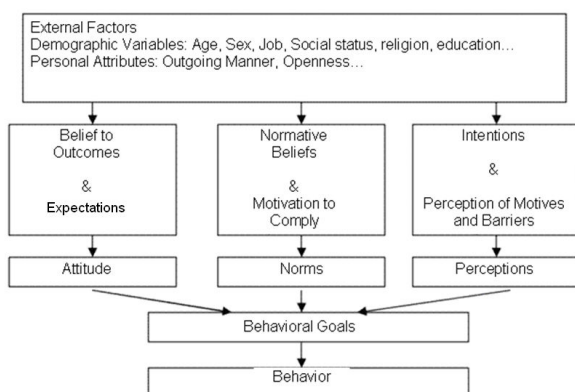
Figure 4 : Theory of Planned Behavior

2-5-Social Cognition Theory (SCT)

Behaviorists insist that a specific boundary exist between stimuli and response and believe that human behavior is just a response to the external stimulus. SCT implies cognition as a medium between stimuli and response, which adjusts control on the Individual behavior.

SCT considers human behavior as the result of dynamic interaction between personal, behavioral and contextual factors. It also implies that people could learn most behaviors and reality means interaction of environment and Individual cognition. SCT has been used to investigate a wide range of health–related issues such as medical acceptance, alcohol abuse, vaccination, and etc. Most STC techniques are used in

modeling and skills training in many interventions, but complexity of this model makes it unsuitable in various contexts.

3-HUMAN – BASED AGENTS IN SIMULATION

Simulation is a suitable tool to solve most problems and has been used in various contexts of application, research and also to assess and train new theories. The ability to build and run a simplified replica of real world situations and evaluate the methods and solutions without implementation is the main source of interests to this technique.

Human beings, are the most important part of most systems for describing these systems, these element should be modeled at least as the other elements. A human lives in a social context and does the various behaviors. The human behavior is a complex system, influenced by several physical, cognitive, sensual, cultural, and social factors. This complexity is more evident when studying the behavior in a complex context, considering human interactions. In most problem solving methods, such as OR modeling, or discrete event Simulation, usually the characteristics of human behavior are ignored, or simply modeled by a few parameters, variables, or some constraints. There should be a trade–off between the complexity of modeled human, and the complexity of modeled environment (Klugl, 98). In the literature, the more detailed organizational structure has been resulted in the more simplified persons. (Moss,nd). One of the most popular techniques to resolve this paradox is using object oriented programming to create human-based agents and to put them in the Simulated environment, as representatives of human beings. (Sibbel, 2001).Using human based Agents is spreading in Simulation experiments (Urban, 2000a). In the recent years, agents have been had a critical role in Simulation models (Schmidt, 2000).

The first matter in Agent–Based Simulation is the internal structure of agents. This structure, named "Reference Model", (Schmidt, 2001) is built upon the theory describing human behavior, describing the behavior in the specified context.

In health care, the theories explained in the second part of this article could be used for designing the reference model. By examining more, the HBM model has been selected, because of its many applications in the literature, and also having the capability to be represented by the mathematical equations.

In the next section, the proposed structure of agent used in the Simulation of screening process is proposed. In this model, the human behavior is modeled by different decision points in the screening process, including the decision to reject or continue the process. Despite the PECS model (described by Urban (Urban, 2000)) in these agents the internal psychological mechanism for making behavior is not

modeled, but like the BDI agents (Pokahr et al, 2003), the decision making process, and the different effecting factors have been modeled. As a result, the psychological details are less included, but the decision making process in the complex context is mentioned.

3-1-THE INTERNAL STRUCTURE

The internal structure of agents is consisted of some types of variables, as described below:

- Personal variables: Indicates the variables that are not related to the other variables which are quantified by a random distribution. These variables are of different types:

- Constants; which does not change in a Simulation run. (12 steps, each one is a month) These include age, sex, marital status, number of children, job and income.

- The next variables are behavioral variables, indicating some behavior like smoking, doing exercise, diets and drug use. These are aggregated in a single variable indicating the overall status of a person.

- The third type of personal variables includes the effects of disease aggregated in a single variable showing the heart stroke, headaches, visual inefficiencies and other results of blood– pressure.

- The fourth type indicates the blood–pressure and results from the distribution of blood–pressure in the selected community.

- Social variables: Indicating the characteristics of the specified society (mainly concerning the attitudes) and also the health care system. These variables include social attitude to disease, (fear of others to know about the disease), time schedule (the ability of being present and having appointment), and drug accessibility.

- Internal variables: these variables, indicate the result on the decision variables. These are mainly resulted from the HBM model, including self–efficacy, perception about infection, the ability to perform, awareness about the blood–pressure, perception of Benefits and Barriers and perception about severity.

- Decision variables: these variables indicate the decisions model by the agent, and directly influence the progress of screening. These variables include having appointment with. Nurse or doctor, performing care and drug–consumption.

- External variables: The external variables, which refer to the health care system, affect the internal and social variable.

- Program Progress variables, which will be explained in 3-3.

3-2-The causal effects

The casual relationships between the variables have been shown in the diagrams by some lines (such as figure number 5). 3 key decisions have been entered in the model:

- Decision for drug–consumption, which is influenced by drug accessibility to construct drug–consumption by the person.

- Decision to perform prescribed care by the doctor or nurse (such as not smoking, having diet and etc…) that cause the gradual (not radical) change of behavior.

- Decision for having appointment with doctor or nurse, by considering the ability to go (presence of Doctor in conjunction with the presence of patient), shows the incidence of appointment.

One of these decisions (having appointment with doctor) is shown in figure 5 as an example. As shown, the personal variables lead the social variable, which in turn shape the internal variables and by interacting

Figure 5- decision tree related to make decisions about having appointment with doctor

these variable, the decision variables are formed, which influence the health–status of the agent.

3-3-Program process

The screening Program consists of four stages which are indicated by Program process variable. These variables are defined as the program process (also presented in figure 2) as follows:

- Program Announcement

- Appointment with nurse: in this stage, persons who decide to have an appointment with nurse could be examined by nurse. If the nurse diagnoses the blood-pressure is more than normal, the person goes to the next stage.

- Appointment with doctor: in this stage, some of agents that have high blood-pressure, decide to go to the doctor, and the ones who could be examined by the doctor (if both are present at the time), are examined. If doctor diagnoses the high blood-pressure, the agent goes to the periodic care cycle. In this stage, doctor prescribes some drugs, as well as physical exercises and diet.

The agents that pass this stage go to the care cycle. There are two stages in this cycle:

- Appointment with nurse (monthly) that includes decision- making about having appointment again.

- Appointment with doctor (once in 3 months): in this stage: patient could decide again about having appointment. Doctor could change his prescription, and also could refer him to the hospital, if necessary.

After each appointment, agent could decide again about drug-consumption and performing the medical care.

4-Model execution and results

A Delphi Program has been used to simulate the designed structure described in the previous session. By executing this Program, user could assign the initial value of variables and could see the characteristics of agents in the right-side box. Then simulation runs and the results are available via an excel file. So, the information could be drilled down to diagnosis the causes of observed behavior.

The main screen of the software program, as shown in figure 6, consists of some places for entering value of parameters and also buttons for run the model. The right-hand side refers to the main variables that should be fed to the program by user. Executing the simulation program using the right button in the bottom of page would show the agents in the text box. By accepting the agents and running the program, results would be shown in an excel file.

The program has been used as a pre-test for blood-pressure screening program which performed in Iranian villages in winter 2004. This program gave decision makers and managers of health-care divisions in the related areas some insights about the program inefficiencies prior to the real execution. As the result, some refinements especially about the announcement and also about the time-frame of the related health-care personnel were performed.

5-CONCLUSION

As this method focuses on the Program progress (not the illness progress), could be used as a suitable tool for professional managers to conduct more suitable decisions.

This view to simulate as a decision-aiding tool is a new approach. Most simulation models concerning analysis

Figure 6- screen shot of the program

of organizational problems, just consider balancing and time-delays in resource (man, material and machine) utilization. In this research, behavioral characteristics interactions are considered. Using simulation for assessing health-care policies is a new idea that has been used in few models for the cancer screening assessment focusing on the physiologic aspects related to the illness progress. But this model is focused on health-care service, in order to design more effective programs.

This model, by ignoring waiting time and available resources, simulates the target group behavior, without system service consideration. So, this agent-based simulation could be used to design a policy for improving the quality of problem solving and decision making in health-care. In human-based agents, by limiting the complexity of human behavior to the internal structure of agents, modeler could use the simulation to anticipate the success and main bottlenecks in the implementation of any program for increasing the efficiency and effectiveness.

In the application of human based agents, the irrelevant research (such as examining the relationship between self–efficacy and the personal characteristics – such as age, sex, marital status and the number of children) could help in building and validating the reference model that could be used for Simulating complex situations. As this method is focused on the progress of program (not on the progress of disease); it could be used as a tool in decision–making for Generalist Managers in the health care. As this model lacks these experiments, it limits the utility of this model, which could be considered in the next studies.

REFERENCES

Brailsford, Sally; Schmidt, Bernd; (2003) Towards incorporating human behavior in models of health care systems: An approach using discrete event simulation, European Journal of Operational Research 150 (2003) 19–31, available from elsevier

Klügl, F., Puppe, F.(1998) The Multi-Agent Simulation Environment SeSAm, Available from Elsevier

Moss, S. (nd), Gaylard, H., Wallis, S., Edmonds, B., SDML: A Multi-Agent Language for Organizational Modelling , Available from : Elsevier

Ohler,P.P;Reger,K.(1999),Emotional Cooperating Agents, In Modeling and Simulation-A tool for the next Millenium, 13th European Simulation Multiconference (proceedings),Available from Elsevier

Pokahr, A., Braubach,L.,Lamersdorf,W. ,2003,Jadex: Implementing a BDI-Infrastructure for JADE Agents, in: EXP - In Search of Innovation (Special Issue on JADE), Vol 3, Nr. 3, Telecom Italia Lab, Turin, Italy, September 2003, pp. 76-85., available from http://vsis-www.informatik.uni-hamburg.de/projects/jadex/papers/jadex-exp03_final_revised.pdf [last accessed feb 2007]

Robinson G.H, P. Wing, L. E. Davis. 1965. Computer simulation of hospital patient scheduling systems. *Health Services Research* 3:130-141.

Schmidt, B.,(2000) Modeling of Human Behavior [online], available from web site of Passau University , URL: www.or.uni-passau.de/5/Publik/Scmidt [Last Accessed 1th May 2003]

Schmidt, B.,(2001) Human Factors in Simulation Models ,available from www.or.unipassau.de/2/Human_Factors.pdf [Last Accessed 21th Feb 2003]

Sibbel , R., Urban , C.(2001) Agent-Based Modeling and Simulation for Hospital Management, [online] available from www.or.uni-passau.de/5/publik/urban /Artikel_Agents_in_Hospital_Management.pdf [Last Accessed 1th March 2003]

Standridge ,Charles R.,1999, A TUTORIAL ON SIMULATION IN HEALTH CARE: APPLICATIONS AND ISSUES, Proceedings of the 1999 Winter Simulation Conference, P. A. Farrington, H. B. Nembhard, D. T. Sturrock, and G. W. Evans, eds.

Urban, C, (2000) PECS:A Reference Model for Human-Like Agents, [online] available from : www.or.uni-passau.de/5/publik/urban /Artikel_Avatars_2000.pdf [Last Accessed 2nd Nov 2003]

Urban, C, (2001) PECS:A Reference Model for Human-Like Agents, [online] available from : www.or.uni-passau.de/5/publik/urban /Artikel_Avatars_2000.pdf [Last Accessed 22th Jan 2003]

Urban, C., (2000) PECS: A Reference Model for the Simulation of Multi-Agent Systems. [online] Available from :Author's HomePage on University of Passau

Urban, C., Schmidt B.(2000) PECS – Agent-Based Modelling of Human Behaviour , [online] available from : www.or.uni-passau.de/5/publik/urban/CUrban01.pdf [Last Accessed 20th June 2003]

AUTHOR BIOGRAPHIES

NASIM GHANBARTEHRANI was born in Tehran, Iran. Now she is a PhD student in Tarbiat Modares University, where she studies Industrial Engineering. She has worked on Simulating Human Behavior as her MS thesis in 2004 in Tarbiat Modares University with Dr Aghdasi. Her Email Address is: Nasim.Tehrani@modares.ac.ir

MOHAMMAD AGHDASI,PhD. has studied Industrial Engineering and gained his Phd from tsuioka university, Japan, in 1988. He is now Professor of Industrial Engineering Faculty at Tarbiat Modares University. He has been worked in Simulation , Productivity ,and Quality area. His Email Address is: Aghdasim@modares.ac.ir

INTERDISCIPLINARY APPROACH TO AUTOMATED NEGOTIATION: A PRELIMINARY REPORT

Fernando Lopes, A. Q. Novais
INETI, Dep. de Modelação e Simulação
Est. do Paço do Lumiar, 1649-038 Lisboa, Portugal
{fernando.lopes, augusto.novais}@ineti.pt

Helder Coelho
Universidade de Lisboa, Dep. de Informática
Campo Grande, 1749-016 Lisboa, Portugal
hcoelho@di.fc.ul.pt

KEYWORDS

Multi-agent systems, autonomous agents, automated negotiation, game theory, bargaining, efficiency.

ABSTRACT

Autonomous agents with negotiation competence are becoming increasingly important and pervasive. This paper follows an interdisciplinary approach to build autonomous negotiating agents by considering both game-theoretic techniques and bargaining procedures from the social sciences. The paper presents a generic model that handles bilateral multi-issue negotiation, describes equilibrium strategies for the bargaining game of alternating offers, and formalizes important strategies used by human negotiators. Autonomous agents equipped with the model are able to negotiate under both complete and incomplete information, thereby making them very compelling for automated negotiation.

INTRODUCTION

Automated negotiation systems with computational agents representing individuals or organizations and capable of reaching mutually beneficial agreements are becoming increasingly important and pervasive. Examples, to mention a few, include the business trend toward agent-based supply chain management, the pivotal role that electronic commerce is increasingly assuming in many organizations, and the industrial trend toward virtual enterprises.

Artificial intelligence (AI) researchers have paid some attention to automated negotiation over the last years and a number of models have been proposed in the literature. These models can be classified into three main classes (Jennings et al. 01):

- *game-theoretic models* − provide clear analysis of specific negotiation situations and precise results concerning the optimal strategies negotiators should choose, *i.e.*, the strategies that maximize the negotiation outcome (see, e.g., Kraus 01, Fatima et al. 06);

- *heuristic models* − provide general guidelines to assist negotiators and beneficial strategies for moving toward agreement, *i.e.*, strategies that lead to good (rather than optimal) outcomes of negotiation (see, e.g., Lopes et al. 04, Li et al. 06);

- *argumentation-based models* − allow negotiators to argue about their mental attitudes during the negotiation process. Thus, in addition to submitting proposals, negotiators can provide arguments either to justify their negotiation stance or to persuade other negotiators to change their negotiation stance (see, e.g., Rahwan et al. 04).

Game-theoretic models have some highly desirable properties, such as Pareto efficiency and the ability to guarantee convergence. However, most models make the following restrictive assumptions: (i) the agents are rational, (ii) the set of candidate solutions is fixed and known by all the agents, and (iii) each agent knows either the other agents' payoffs for all candidate solutions or the other agents' potential attitudes toward risk and expected-utility calculations.

Heuristic models are typically based on informal models of interaction and negotiation from the social sciences. They exhibit the following desirable features: (i) are based on realistic assumptions, and (ii) make use of moderate computational resources to find acceptable solutions. However, most models have a number of limitations. Firstly, they lack a rigorous theoretical underpinning − they are essentially ad hoc in nature. Secondly, they often lead to outcomes that are sub-optimal. Finally, they need extensive evaluation.

Argumentation-based models attempt mainly to marry the exchange of offers with the exchange of arguments. This permits great flexibility since, for instance, it makes possible to persuade agents to change their view of an offer during the course of negotiation. However, most models make considerable demands on any implementation, mainly because they appeal to very rich representations of the agents and their environments.

Against this background, this paper follows an interdisciplinary approach to build agents with negotiation competence. Its main purpose is to present a generic model of negotiation for autonomous agents. The model handles bilateral multi-issue negotiation, describes equilibrium strategies for the bargaining game of alternating offers, and formalizes a set of negotiation strategies frequently used by human negotiators.

Thus, this paper uses both game-theoretic techniques and negotiation procedures from the social sciences as a basis to develop autonomous negotiating agents. On the one hand, it considers two fully informed agents about the various aspects of the bargaining game of alternating offers and employs game-theoretic techniques to define equilibrium strategies. On the other hand, it considers two incompletely informed agents and formalizes important strategies studied in the social sciences (a fruitful area for future work will be to evaluate the effectiveness of these strategies in a number of different situations).

The remainder of the paper is structured as follows. The next section presents the key features of a generic model of negotiation for autonomous agents. In particular, we describe the negotiation protocol and the preferences of the negotiators, present equilibrium strategies for the bilateral multi-issue bargaining game of alternating offers, and introduce a number of concession strategies and their associated tactics. Following this, we discuss related work, present concluding remarks, and outline a number of issues which require further investigation.

THE NEGOTIATION MODEL

Negotiation is usually understood as proceeding through three distinct phases (Lewicki et al. 03): a beginning or initiation phase, a middle or problem-solving phase, and an ending or resolution phase. The initiation phase focuses on preparation and planning for negotiation and is marked by each party's efforts to posture for positions. The problem-solving phase seeks a solution for the dispute and is characterized by movement toward a final agreement. The resolution phase focuses on implementing the final agreement.

Effective preparation and planning involves the creation of a well-laid plan specifying the activities that negotiators should attend to before starting to negotiate. That plan, and the confidence derived from it, is often a critical factor for achieving negotiation objectives. Accordingly, we describe below various activities that negotiators make efforts to perform in order to carefully prepare and plan for negotiation (see Lopes et al. 02, for an in-depth discussion).

Let $Ag = \{ag_1, ag_2\}$ be the set of autonomous negotiating agents. Let $Agenda = \{is_1, \ldots, is_n\}$ be the negotiating agenda − the final set of issues to be deliberated. Effective planning requires that negotiators prioritize the issues. In this work, priorities are set by ranking-order the issues, *i.e.*, by defining the most important, the second most important, and so on. The priority pr_{il} of an agent $ag_i \in Ag$ for each issue $is_l \in Agenda$ is a number that represents its order of preference. The weight w_{il} of is_l is a number that represents its relative importance.

Effective planning also requires that negotiators define two key points for each issue at stake: the limit or reservation price and the target point or level of aspiration. The limit is the point where a negotiator decides that it should stop to negotiate, because any settlement beyond this point is not minimally acceptable. It is set by determining realistic alternatives to a negotiated agreement and selecting the best alternative. The target point is the point where a negotiator realistically expects to achieve a settlement. It is set by first defining the limit and then brainstorming about what can be realistically possible. The limit of ag_i for is_l is denoted by lim_{il} and the target point by trg_{il}.

Additionally, effective planning requires that negotiators agree on an appropriate protocol that defines the rules governing the interaction. The protocol can be simple, allowing agents to exchange only proposals. Alternatively, the protocol can be sophisticated, allowing agents to provide arguments to support their negotiation stance. As noted, most sophisticated protocols make considerable demands on any implementation. Thus, in this work we consider a simple protocol (see subsection "The Negotiation Protocol").

Finally, effective planning requires that negotiators be able to select appropriate strategies that account for their individual actions. Negotiation strategies can reflect a variety of behaviours and lead to strikingly different outcomes. Some strategies are in equilibrium, meaning that no designer will benefit by building agents that use any other strategies when it is known that some agents are using equilibrium strategies. Thus, for some situations of complete information, the agents can be designed to adopt equilibrium strategies (see subsection "Equilibrium Strategies"). However, for situations of incomplete information, the problem of strategic choice is rather complex. In these situations, many bargaining models have different equilibria sustained by different assumptions on what an individual in the game would believe if its opponent took an action that it was not supposed to take in equilibrium. Hence, our study differs from this line of work. As stated, we address the challenge of building

agents that are able to negotiate under incomplete information by formalizing relevant strategies used by human negotiators and empirically evaluating the effectiveness of these strategies in different situations (see subsection "Concession Strategies and Tactics").

At this stage, we hasten to add an explanatory and cautionary note. This paper primarily focuses on the problem-solving phase of negotiation. Most AI researchers have also focused on this phase of negotiation and have addressed various issues associated with the design of negotiation protocols and strategies. Their work has resulted in some advances in automated negotiation, but it is clear that much more research still needs to be performed. At present, the deployment of agents that are able to negotiate under both complete and incomplete information is still in its infancy.

The Negotiation Protocol

The protocol is an alternating offers protocol (Osborne and Rubinstein 90). Two agents or players, ag_1 and ag_2, bargain over the division of the surplus of $n \geq 2$ issues (or goods). The players determine an allocation of the issues by alternately proposing offers at times in $\mathcal{T} = \{1, 2, \ldots\}$. This means that one offer is made per time period $t \in \mathcal{T}$, with an agent, say ag_i, offering in odd periods $\{1, 3, \ldots\}$, and the other agent ag_j offering in even periods $\{2, 4, \ldots\}$.

The negotiation procedure, labelled the "joint-offer procedure", involves bargaining over the allocation of the entire endowment stream at once. An offer is a vector $(x_1, \ldots, x_n)$ specifying a division of the n goods. Once an agreement is reached, the agreed-upon allocations of the goods are implemented. This procedure permits agents to exploit the benefits of trading-off concessions on their less preferred goods for concessions by their opponent on the more preferred goods.

The players' preferences are modelled by assuming that each player ag_i discounts future payoffs at some given rate δ_i^t, $0 < \delta_i^t < 1$, (δ_i^t is referred to as the discount factor and the preferences as time preferences with a constant discount rate). The cost of bargaining derives from the delay in consumption implied by a rejection of an offer. Practically speaking, the justification for this form of preferences takes into account the fact that money today can be used to make money tomorrow.

Let U_i be the payoff function of ag_i. For simplicity and tractability, we assume that U_i is separable in all their arguments and that the per-period delay costs are the same for all issues:

$$U_i(is_{i1}, \ldots, is_{in}, t) = \delta_i^{(t-1)} \sum_{l=1}^{n} w_{il} \, u_{il}(is_{il})$$

where w_{il} is the weight of an issue is_{il} and u_{il} is the component payoff function for is_{il}, i.e., the function that gives the payoff that ag_i assigns to a value of is_{il}. The function u_{il} is a continuous, strictly monotonic, and linear function. The distinguish feature of time preferences with a constant discount rate is the linearity of the function u_{il}. The payoff of disagreement is normalized at 0 for both players.

Equilibrium Strategies

Game theory can provide sound design principles for computer scientists. Consider two fully informed players bargaining over four distinct goods $\{X_1, \ldots, X_4\}$. Each good is modelled as an interval $[0, 1]$ (or as a divisible pie of size 1). The players' preferences are as follows:

$$U_i = \delta_i^{(t-1)} \, (a \, x_1 + b \, x_2 + x_3 + x_4)$$

$$U_j = \delta_j^{(t-1)} \, [(1 - x_1) + (1 - x_2) + c(1 - x_3) + d \, (1 - x_4)]$$

where x_l, $l = 1, \ldots, 4$, denotes the share of ag_i for each pie. The parameters a, b, c, and d allow the marginal utilities of the players to differ across issues and players. We consider $a > b > 1$ and $d > c > 1$, i.e., ag_i places greater emphasis on goods X_1 and X_2 while ag_j values goods X_3 and X_4 more. Also, we consider that δ_i and δ_j are close to 1 and the parameters a, b, c, and d are close to one another. Let $p_{j \to i}^{t-1}$ and $p_{i \to j}^{t}$ denote the offers that ag_j proposes to ag_i in period $t-1$ and ag_i proposes to ag_j in period t, respectively. Consider the following strategies for ag_i and ag_j:

$$str_i^* = \begin{cases} \text{offer } (1, 1, x_{i3}^*, 0) & \text{if } ag_i\text{'s turn} \\ \text{if } U_i(p_{j \to i}^{t-1}) \geq U_i^* \text{ accept else reject} & \text{if } ag_j\text{'s turn} \end{cases}$$

$$str_j^* = \begin{cases} \text{offer } (1, x_{j2}^*, 0, 0) & \text{if } ag_j\text{'s turn} \\ \text{if } U_j(p_{i \to j}^{t}) \geq U_j^* \text{ accept else reject} & \text{if } ag_i\text{'s turn} \end{cases}$$

where $U_i^* = U_i(1, x_{j2}^*, 0, 0)$, $U_j^* = U_j(1, 1, x_{i3}^*, 0)$, and the shares are the following: $x_{i3}^* = \frac{\delta_i \delta_j (a+b) - \delta_j (a+b+bc+bd) + bc + bd}{bc - \delta_i \delta_j}$ and $x_{j2}^* = \frac{\delta_i (\delta_i \delta_j (a+b) - \delta_j (a+b+bc+bd) + bc + bd) + (bc - \delta_i \delta_j)(a \delta_i + b \delta_i - a)}{b(bc - \delta_i \delta_j)}$.

Remark 1. For the two-sided four-issue bargaining game of alternating offers with an infinite horizon, in which the players' preferences are as described above, the pair of strategies (str_i^*, str_j^*) form an equilibrium. The outcome is the following:

$$x_1^* = 1, \qquad x_2^* = 1, \qquad x_4^* = 0,$$

$$x_3^* = \frac{\delta_i \delta_j (a + b) - \delta_j (a + b + bc + bd) + bc + bd}{bc - \delta_i \delta_j}$$

Agreement is immediately reached with no delay. The outcome is Pareto optimal. Letting $\delta_i \to 1$ and $\delta_j \to 1$, the equilibrium division is $(1, 1, 0, 0)$.

The formal proof is based on the familiar necessary conditions for equilibrium: ag_i is indifferent between waiting one period to have its offer accepted and accepting ag_j's offer immediately, and ag_j is indifferent between waiting one period to have its offer accepted and accepting ag_i's offer immediately. Let $\mathbf{x}_i^* = (x_{i1}^*, \ldots, x_{i4}^*)$ and $\mathbf{x}_j^* = (x_{j1}^*, \ldots, x_{j4}^*)$ be the equilibrium proposals of ag_i and ag_j, respectively. The problem for ag_i is to find an offer that maximizes its payoff (because it is a payoff maximizer) subject to being acceptable to its opponent, *i.e.*,

maximize:
$$U_i(x_1, \ldots, x_4, t) = \delta_i^{(t-1)} \, (ax_1 + bx_2 + x_3 + x_4)$$
subject to:
$$(1-x_{i1}^*)+(1-x_{i2}^*)+c(1-x_{i3}^*)+d(1-x_{i4}^*) =$$
$$\delta_j[(1-x_{j1}^*)+(1-x_{j2}^*)+c(1-x_{j3}^*)+d(1-x_{j4}^*)]$$
$$0 \le x_{il}^* \le 1, \quad 0 \le x_{jl}^* \le 1, \quad \text{for} \quad l=1,\ldots,4$$

The problem for ag_j is stated in a similar way and is omitted. Solving both maximization problems yields the outcome specified in the statement of the Remark. In the limit, letting $\delta_i \to 1$ and $\delta_j \to 1$, the outcome of the equilibrium is $(1,1,0,0)$. This outcome is on the Pareto frontier and corresponds to the utility pair $(a+b, c+d)$.

Now, consider two fully informed players bargaining over n distinct goods $\{X_1, \ldots, X_n\}$. Again, each good is modelled as a divisible pie of size 1. We consider that the players set different weights for the goods such that: $w_{i1}/w_{j1} > w_{i2}/w_{j2} > \ldots > w_{in}/w_{jn}$.

Remark 2. The bilateral multi-issue bargaining game of alternating offers with an infinite horizon, in which the players' preferences are as described above, has an equilibrium. The outcome is Pareto optimal:

$$(x_1^*, \ldots, x_{k-1}^*, x_k^*, x_{k+1}^*, \ldots, x_n^*) \;=\; (1, \ldots, 1, s, 0, \ldots, 0)$$

where x_l^*, $l=1,\ldots,n$, denotes the share of ag_i for each divisible pie. The constant s represents the share of ag_i for the X_k pie.

The formal proof is also based on the familiar necessary conditions for equilibrium. In short, consider that $\mathbf{x}_i^* = (x_{i1}^*, \ldots, x_{in}^*)$ and $\mathbf{x}_j^* = (x_{j1}^*, \ldots, x_{jn}^*)$ represent the equilibrium proposals of players ag_i and ag_j, respectively. The problem for ag_i is stated as follows:

maximize:
$$U_i(x_1, \ldots, x_n, t) = \delta_i^{(t-1)} \sum_{l=1}^{n} w_{il} \, x_l$$
subject to:
$$U_j(x_{i1}^*, \ldots, x_{in}^*, t) = U_j(x_{j1}^*, \ldots, x_{jn}^*, t+1)$$
$$0 \le x_{il}^* \le 1, \; 0 \le x_{jl}^* \le 1, \; \text{for} \; l=1,\ldots,n$$

This maximization problem is similar to the continuous knapsack problem and solvable by a greedy approach (see, e.g., Martello and Toth 90). First, ag_i gives away the maximum possible share of the issue with the lowest ratio of weights. If the supply of that issue is exhausted, it gives away the maximum possible share of the issue with the next lowest ratio of weights, and so forth until ag_j gets the utility of $U_j(x_{j1}^*, \ldots, x_{jn}^*, t+1)$. The problem for ag_j is also analogous to the continuous knapsack problem and solvable by a greedy approach.

Concession Strategies and Tactics

Human negotiation practice can provide rules-of-thumb to agent designers. Consider two incompletely informed agents bargaining over n distinct issues $\{is_1, \ldots, is_n\}$. For convenience, each issue is_l is modelled as an interval $[min_l, max_l]$. The agents' preferences are as defined in the subsection "The Negotiation Protocol".

The opening stance (or position) and the pattern of concessions are two central elements of real-world negotiation. Three different opening positions (extreme, reasonable and modest) and three levels of concession magnitude (large, moderate and small) are commonly discussed in the literature (Lewicki et al. 03, Pruitt and Kim 04). They can lead to a number of concession strategies, notably:

- *starting high and conceding slowly* − negotiators adopt an optimistic opening attitude and make successive small concessions;
- *starting reasonable and conceding moderately* − negotiators adopt a realistic opening attitude and make successive moderate concessions.

These strategies are implemented through a variety of tactics. The terms "strategy" and "tactic" differ in scope − a tactic is a short-term move designed to enact a high-level strategy.

Let $p_{j \to i}^{t_{n-1}}$ be the offer that ag_j has proposed to ag_i in period t_{n-1}. Likewise, let $p_{i \to j}^{t_n}$ be the offer that ag_i is ready to propose in the next time period t_n. The formal definition of a generic concession strategy follows.

Definition 1. Let $ag_i \in Ag$ be a negotiating agent. A concession strategy for ag_i is a function that specifies either the tactic to apply at the beginning of negotiation or the tactic that defines the concessions to be made during the course of negotiation:

$$s = \begin{cases} \text{apply } tact_i^{t_1} & \text{if } ag_i\text{'s turn and } t=1 \\ \text{apply } tact_i^{t} & \text{if } ag_i\text{'s turn and } t>1 \\ \text{if } U_i(p_{j \to i}^{t_{n-1}}) \ge U_i(p_{i \to j}^{t_n}) \text{ accept else reject if } ag_j\text{'s turn} \end{cases}$$

where $tact_i^{t_1}$ is an opening negotiation tactic and $tact_i^{t}$ is a concession tactic. ∎

The two concession strategies are defined by considering different tactics. For instance, the "starting reasonable and conceding moderately" strategy is defined by: "$tact_i^{t_1} = starting\_realistic$" and "$tact_i^t = moderate$".

Opening negotiation tactics are functions that specify the initial values for each issue is_l at stake. The following three tactics are commonly used in real-life negotiations (Lewicki et al. 03):

- *starting optimistic* − specifies a value far from the target point;
- *starting realistic* − specifies a value close to the target point;
- *starting pessimistic* − specifies a value close to the limit.

The definition of the tactic "starting realistic" follows (the definition of the other two tactics is essentially identical, and is omitted).

Definition 2. Let $ag_i \in Ag$ be a negotiating agent and $is_l \in Agenda$ a negotiation issue. Let trg_{il} be the target point of ag_i for is_l. The tactic starting realistic for ag_i is a function that takes is_l and trg_{il} as input and returns the initial value $v[is_l]_i^{t_1}$ of is_l:

$$starting\_realistic(is_l, trg_{il}) = v[is_l]_i^{t_1}$$

where $v[is_l]_i^{t_1} \in [\,trg_{il}{-}\epsilon,\ trg_{il}{+}\epsilon\,]$ and $\epsilon > 0$ is small. ∎

Concession tactics are functions that compute new values for each issue is_l. The following five tactics are commonly used (Lewicki et al. 03):

- *stalemate* − models a null concession on is_l;
- *tough* − models a small concession on is_l;
- *moderate* − models a moderate concession;
- *soft* − models a large concession;
- *accommodate* − models a complete concession.

The definition of a generic concession tactic follows (without loss of generality, we consider that ag_i wants to maximize is_l).

Definition 3. Let $ag_i \in Ag$ be a negotiating agent, $is_l \in Agenda$ a negotiation issue, and lim_{il} the limit of is_l. Let $v[is_l]_i^t$ be the value of is_l offered by ag_i at t. A concession tactic for ag_i is a function that takes $v[is_l]_i^t$, lim_{il} and the concession factor $Cf \in [0, 1]$ as input and returns the new value $v[is_l]_i^{t+2}$ of is_l:

$$concession\_tactic(v[is_l]_i^t, lim_{il}, Cf) = v[is_l]_i^{t+2}$$

where $v[is_l]_i^{t+2} = v[is_l]_i^t - Cf(v[is_l]_i^t - lim_{il})$. ∎

The five tactics are defined by considering different values for Cf. In particular, the stalemate tactic by $Cf = 0$, the accommodate tactic by $Cf = 1$, and the other three tactics by different ranges of values for Cf (e.g., the tough tactic by $Cf \in\,]0.00, 0.05]$, the moderate tactic by $Cf \in\,]0.05, 0.10]$, and the soft tactic by $Cf \in\,]0.10, 0.15]$).

RELATED WORK

AI researchers have developed a number of negotiation models incorporating specific protocols and libraries of negotiation strategies (see, e.g., Kraus 01, Lopes et al. 04, Fatima et al. 06, Li et al. 06). However, despite these and other relevant pieces of work, we are aware of no similar efforts to define strategies as functions that specify the tactics to be used at every period of negotiation. Tactics, in turn, are defined as functions that specify the short-term moves to be made throughout negotiation. Also, we are particularly interested in both equilibrium strategies that lead to Pareto optimal outcomes and important strategies studied in the social sciences and frequently used by human negotiators.

CONCLUSION

This article has followed an interdisciplinary approach and has used both game-theoretic techniques and negotiation procedures from the social sciences as a basis to develop autonomous negotiating agents. It has presented a model that handles bilateral multi-issue negotiation, describes equilibrium strategies for the bargaining game of alternating offers, and formalizes a set of human negotiation strategies. Our aim for the future is: (i) to extend the model, (ii) to empirically evaluate the model, and (iii) to use game-theoretic techniques as a basis to develop autonomous agents that are able to negotiate under incomplete information.

REFERENCES

[Fatima et al. 06] Fatima S., M. Wooldridge and N. Jennings (2006), Multi-Issue Negotiation with Deadlines, *Journal of Artificial Intelligence Research*, 27:381–417.

[Jennings et al. 01] Jennings N., P. Faratin, A. Lomuscio, S. Parsons, M. Wooldridge and C. Sierra (2001), Automated Negotiation: Prospects, Methods and Challenges, *Group Decision and Negotiation*, 10:199–215.

[Kraus 01] S. Kraus (2001), *Strategic Negotiation in Multi-Agent Environments*, MIT Press.

[Lewicki et al. 03] Lewicki R., B. Barry, D. Saunders and J. Minton (2003), *Negotiation*, McGraw Hill.

[Li et al. 06] C. Li, J. Giampapa, and K. Sycara (2006), Bilateral Negotiation Decisions with Uncertain Dynamic Outside Options, *IEEE Transactions on Systems, Man, and Cybernetics* (Part-C), 36:31–44.

[Lopes et al. 02] F. Lopes, N. Mamede, A. Q. Novais, and H. Coelho (2002), A Negotiation Model for Autonomous Agents: Formal Description and Empirical Evaluation, *Journal of Intelligent & Fuzzy Systems*, 12:195–212.

[Lopes et al. 04] Lopes F., N. Mamede, A. Q. Novais and H. Coelho (2004), Negotiation Strategies for Autonomous Computational Agents, In: *ECAI-04*, pp. 38–42.

[Martello and Toth 90] Martello S. and P. Toth (1990), *Knapsack Problems: Algorithms and Computer Implementations*, John Wiley & Sons.

[Osborne and Rubinstein 90] Osborne M. and A. Rubinstein (1990), *Bargaining and Markets*, Academic Press.

[Pruitt and Kim 04] D. Pruitt and S. Kim (2004), *Social Conflict*, McGraw Hill.

[Rahwan et al. 04] Rahwan I., S. Ramchurn, N. Jennings, P. McBurney, S. Parsons and L. Sonenberg (2004), Argumentation-based Negotiation, *The Knowledge Engineering Review*, 18:343–375.

AGENT-BASED AND POPULATION-BASED SIMULATION:
A COMPARATIVE CASE STUDY FOR EPIDEMICS

S. Waqar Jaffry, Jan Treur
Vrije Universiteit Amsterdam, Department of Artificial Intelligence
De Boelelaan 1081, 1081 HV Amsterdam, The Netherlands
{swjaffry,treur}@few.vu.nl, http://www.few.vu.nl/~{swjaffry,treur}

KEYWORDS

Simmulation, agent-based, population-based, epidemics.

ABSTRACT

This paper reports a comparative evaluation of population-based simulation in comparison to agent-based simulation for different numbers of agents. Population-based simulation, such as for example in the classical approaches to predator-prey modelling and modelling of epidemics, has computational advantages over agent-based modelling with large numbers of agents. Therefore the latter approaches can be considered useful only when the results are expected to deviate from the results of population-based simulation, and are considered more realistic. However, there is sometimes also a silent assumption that for larger numbers of agents, agent-based simulations approximate population-based simulations, which would indicate that agent-based simulation just can be replaced by population-based simulation. The paper evaluates this assumption by a detailed comparative case study in epidemics.

INTRODUCTION

The classical approaches to simulation of processes in which groups of larger numbers of agents are involved are population-based: a number of groups are distinguished (populations) and each of these populations is represented by a numerical variable indicating their number or density (within a given area) at a certain time point. The simulation model takes the form of a system of difference or differential equations expressing temporal relationships for the dynamics of these variables. Well known classical examples of such population-based models are systems of difference or differential equations for predator-prey dynamics (e.g., Lotka, 1924; Volterra, 1926, 1931; Maynard Smith, 1974; Burghes and Borrie, 1981) and the dynamics of epidemics (e.g., Ross, 1916; Kermack and McKendrick, 1927; Burghes and Borrie, 1981; Anderson and May, 1992; Ellner and Guckenheimer, 2006). Such models can be studied by simulation and by using analysis techniques from mathematics and dynamical systems theory.

From the more recently developed agent system area it is often taken as a presupposition that simulations based on individual agents are a more natural or faithful way of modelling, and thus will provide better results (e.g., Davidsson, Gasser, Logan, and Takadama, 2005; Sichman and Antunes, 2006; Antunes and Takadama,

2007). Although for larger numbers of agents such agent-based modelling approaches are more expensive computationally than population-based modelling approaches, such a presupposition may provide a justification of preferring their use over population-based modelling approaches, in spite of the computational disadvantages. In other words agent-based approaches with larger numbers of agents are justified because the results are expected to deviate from the results of population-based simulation, and are considered more realistic.

However, in contrast there is another silent assumption sometimes made, namely that for larger numbers of agents (in the limit), agent-based simulations approximate population-based simulations. This would indicate that agent-based simulation just can be replaced by population-based simulation, which would weaken the justification for agent-based simulation discussed above. In this paper, by a case study in epidemics, these considerations are explored in more detail. Comparative simulation experiments have been conducted based on different simulation models, both agent-based (for different numbers of agents), and population-based. The results are analysed and related to the presupposition and assumption discussed above.

THE DOMAIN OF EPIDEMICS

Microbes such as viruses, bacteria, fungi and parasites, may have disturbing effects when they enter the human body. Not seldom humans suffer from such infections and in the mean time propagate them to each other. Examples of types of infectious diseases are influenza, chlamydia, HIV, hepatitis, tuberculosis, and many others. The battle against such infections takes place both at the biological level in the body, and at the behavioural and social level. At the behavioural and social level, humans sometimes try to adapt their interaction behaviour to prevent propagation of infections from one human to another one. This paper focuses on the propagation of infections in populations, in relation to the interaction behaviour, in particular the frequency and intensity of contacts that individuals have in the population.

Agents within a population can be in different states: susceptible (not infected yet), infective, or recovered (immune and not infectious). When an agent who is infective, has contact with another, susceptible agent then there is a chance that the other agent will also be infected due to this contact. This chance depends on the intensity of the contact. The overall chance that a susceptible agent is infected, also depends on the

number of contacts with infective agents. A possible pattern, for example, for an easy transmittable infection such as influenza, is that the propagation goes so fast that only in a few weeks time almost the whole population is infected. In such a case the term epidemic is used to indicate the spreading of the infection over the population. For other types of infections, for example HIV or chlamydia, more intensive contacts (which usually occur less frequently) are needed for transmission, and therefore propagation may proceed slower.

An important question, especially for the more harmful infections, in a society is whether by measures at the behavioural and social level, it is possible to keep the number of infected persons in a population limited. And if so, how far should such measures go? It is clear that by avoiding any contact between agents, propagation can be stopped, but that is often not a realistic option. On the other hand, if there are still some contacts between agents, will at the end the infection not be spread (perhaps by very slow propagation) over the whole population? Such questions are addressed in this paper by two types of models: population-based models (with populations of susceptible, infectives and recovered agents, respectively) and agent-based models.

A POPULATION-BASED MODEL

This section describes the population-based model. The analysis of epidemics has a long history, going back, for example, to (Ross, 1916; Kermack and McKendrick, 1927). More recent presentations can be found in (Anderson and May, 1992) and (Ellner and Guckenheimer, 2006, Ch. 6, pp. 183-215). First of all, a distinction is made between the population of *susceptibles* vs. the population of *infectives*, the latter of which are infectious for the former. A third population consists of those that already were infected, but have *recovered* and therefore are immune and not infectious anymore, based on a *recovery rate* indicating the fraction of infectives that recovers per day. Furthermore the *frequency* of *contacts* (per day, the time unit chosen) plays a main role; the chance that in a contact *infection transmission* occurs depends on the *contact intensity*.
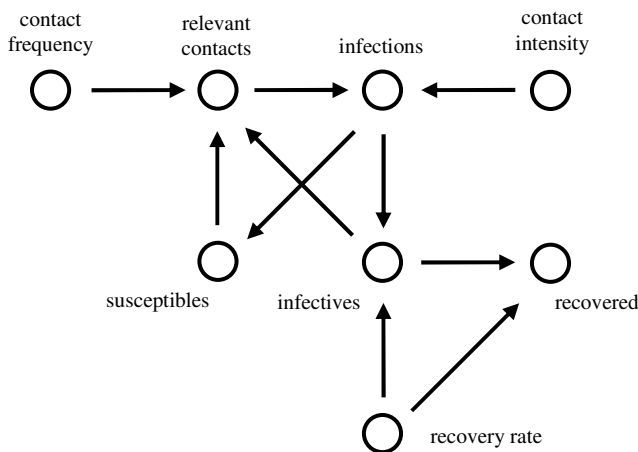
Figure 1: Dynamic Relationships in the Model

The populations can be described by their sizes, but often they are characterised by their densities: size divided by area. As the area is considered fixed, the sizes (numbers) will be used to characterize the populations. The dynamic relationships between these concepts are depicted in Figure 1. For a mathematical formalisation usually the contact frequency times the contact intensity divided by the overall size of the populations together is combined in one parameter, called the *contact rate*. Thus the following variables and parameters are used.

size of the population of susceptible individuals	S
size of the population of infective individuals	I
size of the population of recovered individuals	R
size of all populations together	N
contact rate	β
recovery rate	γ
threshold	ρ

Here $\beta = ContactFrequency*ContactIntensity/N$. Note that for given values of contact frequency and contact intensity, this parameter β depends on the overall population size. The dynamics of these concepts involve temporal relationships, which are analysed in more detail below. Each susceptible person has (per day) a number of contacts indicated by *ContactFrequency*. From these contacts a fraction $I(t)/N$ is with infective individuals, where N is the size of the three populations together (assumed fixed). Therefore the number of relevant contacts per day is: $RelevantContacts(t)= ContactFrequency*S(t)*I(t)/N$. Moreover, in a fraction of the contacts the infection is transmitted. This fraction is indicated by *ContactIntensity*; therefore the number of new infections per day is:

$Infections(t)$
$= ContactIntensity*RelevantContacts(t)$
$= ContactIntensity* ContactFrequency*S(t)*I(t)/N$
$= \beta*S(t)*I(t).$

Given the number of infections *Infections(t)* per time unit, in a time interval between t and $t+\Delta t$ the number of (new) infections is $Infections(t)*\Delta t$. This is subtracted from the susceptible population, and added to the infective population. Furthermore, γ indicates the fraction of the infective population per day that becomes recovered (and not infective anymore): over the interval between t and $t+\Delta t$ a number of $\gamma*I(t)*\Delta t$ is taken from the infective population and added to the recovered population. Therefore the following temporal relationships are used.

$S(t+\Delta t =\ S(t) -\ Infections(t)*\Delta t$
$I(t+\Delta t) =\ I(t) +\ (Infections(t) -\ \gamma* I(t))\ *\Delta t$
$R(t+\Delta t) =\ R(t) +\ \gamma* I(t)\ *\Delta t$

Note that by these relationships the sum of the three populations always remains the same: what adds to the recovered population subtracts from the infective population, and what subtracts from the susceptible population adds to the infective population. In the more usual notation, by replacing *Infections(t)* the equations can be written as:

$S(t+\Delta t) = S(t) - \beta*S(t)*I(t)*\Delta t$
$I(t+\Delta t) = I(t) + (\beta*S(t)*I(t) - \gamma* I(t)) *\Delta t$
$R(t+\Delta t) = R(t) + \gamma* I(t) *\Delta t$

In differential equation form they are represented in the following manner; for example, see also in (Kermack and McKendrick, 1927; Burghes and Borrie, 1981; Anderson and May, 1992; Ellner and Guckenheimer, 2006):

$$\frac{dS(t)}{dt} = - \beta*S(t)*I(t)$$
$$\frac{dI(t)}{dt} = \beta*S(t)*I(t) - \gamma* I(t)$$
$$\frac{dR(t)}{dt} = \gamma* I(t)$$

Note again, that the parameter β in principle depends on the overall population size. This means that to do experiments with different overall population sizes, different values for β may have to be used. Based on these equations the following analysis has been made:

(a) Threshold for increase/decrease of infective population

Increase and decrease of the size of the population of infectives are characterised by

$$\frac{dI(t)}{dt} \geq 0 \quad \Leftrightarrow \quad \beta*S(t)*I(t) - \gamma* I(t) \geq 0$$
$$\frac{dI(t)}{dt} \leq 0 \quad \Leftrightarrow \quad \beta*S(t)*I(t) - \gamma* I(t) \leq 0$$

This can be characterised by the size of the population of susceptibles as follows with $\rho = \gamma/\beta$:

$I(t)$ increasing $\quad \Leftrightarrow \quad S(t) \geq \rho$
$I(t)$ decreasing $\quad \Leftrightarrow \quad S(t) \leq \rho$

This shows that the usual pattern is that the size of the population of infectives will increase until the size of the population of susceptibles has become lower than the threshold ρ, after which it will decrease. In particular, when the initial size $S(0)$ is already less than this threshold ρ, then the number of infectives will decrease right from the start. This is called the *epidemic threshold law* with *threshold ρ*.

(b) Equilibria

An equilibrium occurs if and only if

$$\frac{dS(t)}{dt} = \frac{dI(t)}{dt} = \frac{dR(t)}{dt} = 0$$

which is characterised by

$\beta*S(t)*I(t) = \beta*S(t)*I(t) - \gamma* I(t) = \gamma* I(t) = 0$

This is equivalent to $I(t) = 0$. Notice that by itself this does not put any constraint on $S(t)$ or $R(t)$. Equilibria may depend on initial values as well. However, taken together with a) in the usual pattern in an equilibrium state $S(t)$ will have become below ρ. So, when ρ is rather small (e.g., individuals remain infective for a long time, or contact intensity is high), the number of individuals that never become infected will also be small, or even zero. These observations are illustrated by simulations in the next section.

(c) Relation between equilibria and initial values

From the set of differential equations, in particular the first and third one, it can be derived that

$$\frac{dS(t)}{dt} = - \beta*S(t)*I(t) = -\frac{1}{\rho} S(t) \frac{dR(t)}{dt} \quad \text{or}$$
$$\frac{dR(t)}{dt} = -\frac{\rho}{S(t)} \frac{dS(t)}{dt}$$

By integration, using the natural logarithm, it follows for all t it holds

$R(t) = C - \rho\, log(S(t))$

with C a constant. Assuming $R(0) = 0$, it holds $C = \rho\, log(S(0))$. Therefore

$R(t) = \rho\, log(S(0)) - \rho\, log(S(t)) = \rho\, log(S(0)/S(t)).$

Equivalent forms are:

$e^{R(t)/\rho} = S(0)/S(t)$

$S(t) = S(0)\, e^{-R(t)/\rho}$

$S(0) = S(t)\, e^{R(t)/\rho}$

Now, according to (a) for an equilibrium occurs if and only if $I(t) = 0$, which is equivalent to $S(t) + R(t) = N$. Filled in the above formula this obtains:

$S(0) = (N - R(t))\, e^{R(t)/\rho}$ or

$S(0) = S(t)\, e^{(N-S(t))/\rho}$

This shows a relation between the population sizes in an equilibrium state and the initial values; note that $S(0) = N - I(0)$. Note that conversely, each of these relations also implies that $S(t) + R(t) = N$, hence $I(0) = 0$ and an equilibrium state occurs. So, these relations provide if and only if criteria for an equilibrium to occur.

POPULATION-BASED SIMULATIONS

A number of population-based simulation experiments have been performed using a Spread Sheet (Microsoft® Office Excel® 2007). In Figures 2 and 3 results are shown of one of them, with time scale in days. In the first simulation shown in Figure 2 the whole population gets infected; it used the following parameter settings:

N   100
ContactFrequency   0.8
ContactIntensity   0.5
$\beta = 0.004, \gamma = 0.05, \quad \rho = 12.5$

Initially the size of the infective population is 1. Given the analysis above, in this case it may be expected that the size of the population of susceptibl-es will become below *12.5*. Note that in this and next figures the scales on the vertical axis differ.
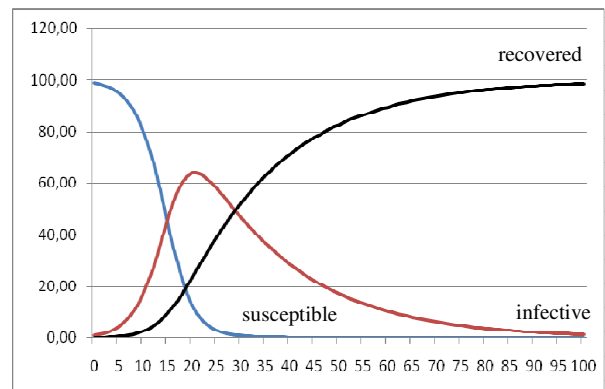

Figure 2: Pattern in which the Whole Population gets Infected

The size of the susceptibles decreases to zero, while the size of the infective population increases until day 20 and decreases after this day. The size of the recovered population shows a logistic growth pattern with the whole population of *100* as limit. Notice that the maximal size of the population of infectives is taken at the time point that the size of the susceptibles population is around *12.5*, which is the value of the threshold ρ. In the second simulation, (see Figure 4), only a part gets infected; parameter settings were:

N	*100*
ContactFrequency	*0.6*
ContactIntensity	*0.2*

β = 0.0012, γ = 0.1, ρ = 83.3

Here initially the size of the infective population is 10. Apparently here the contact frequency and intensity were low enough to let the infection die out: around 50% of the population is never infected. The logistic growth pattern of the (infected and) recovered population has its limit around 50. Nevertheless, the individuals still did not bring their contacts down to zero, or even close to zero.
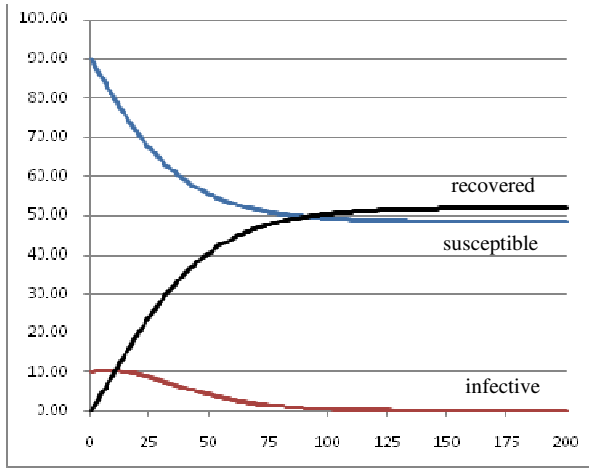

Figure 3: Only Part gets Infected, Starting with 10

This shows that by relatively small differences in behaviour at the individual level, relatively big differences at the collective level can be realised. Notice that for this case the maximal size of the population of infectives is at the time point that the size of the susceptibles population is around *83*, which is the value of the threshold ρ.

AN AGENT-BASED MODEL

To obtain a model at the level of individual agents, *N* distinct agents and *L* distinct locations are introduced. At every time point each agent is at some location, at random. Contacts between agents are modelled as being at the same location. By taking the number *L* of locations (numbered by *1, 2, …, L*) lower or higher, a specific contact frequency is modelled. Each agent is in precisely one of three infection states (susceptible, infective, recovered).

The number of locations has a relationship with the contact frequency in the following manner. If *L* is the

number of locations, and *N* the number of agents, then the average number of agents at one location is *N/L*, so the average number of contacts of one agent at such a location is *N/L – 1*. This is equal to the contact frequency. Therefore *ContactFrequency = N/L - 1* gives the relation between number of locations and contact frequency. To be able to compare this model at the agent level to the model at the population level, it is convenient to have contact frequency as a basic parameter. To this end, the relation between the number of locations and contact frequency shown above is used in an inverse manner to determine the number of locations for a given value of the contact frequency: *L = N/(ContactFrequency + 1)*. For a given contact frequency, this *L* is taken as a bound for the number of locations: they are indexed by the natural numbers *k* with $1 \le k \le L$. At each time point locations of the agents are determined at random, using this bound *L* by taking at random one of the natural numbers between *1* and *L*.

When a suceptible agent *A* is at a certain location, the probability that infection takes place depends on the contact intensity, but also on the number *k* of infective agents present at that location. Although in an agent-based model, contact intensity may be taken as depending on the agent or even on the pair of agents involved in a contact, for reasons of comparability with the population-based model the contact intensities are taken uniform: in any contact between any susceptible agent *A* and any infective agent *B*, the probability that *A* will be infected is *ContactIntensity*. Given this assumption, the probability that agent *A* will not be infected by a specific infective agent at the same location is *1 – ContactIntensity*. Assuming independence of these probabilities, the probability that *A* will not be infected by any of the infective agents present at that location is $(1 - ContactIntensity)^k$. Therefore the probability that *A* will be infected at that location (at that time point) is $1 - (1 - ContactIntensity)^k$. The following relationships describe the changes of the infection state of an agent *A*. Here *r1* and *r2* are two independent random numbers between *0* and *1:* fixed per time point but refreshed at new time points where *Δt=1*. When a susceptible agent is at a location where one or more infective agents are present, the transmission of the infection at that time point has a probability given by the contact intensity. Moreover, for someone who is infective there is a probability of recovery given by the recovery rate. This is modelled by the following relationships:

if   *InfectionState(A, t) = susceptible* and
      there are k infective agents at the same location as A
      and $r1 < 1 - (1 - ContactIntensity)^k$
or   *InfectionState(A, t) = infective* and
      r2 ≥ RecoveryRate
then *InfectionState(A, t+1) = infective*
else if *InfectionState(A, t) = infective* and
      r2 < RecoveryRate
or   *InfectionState(A, t) = recovered*
then *InfectionState(A, t+1) = recovered*
else *InfectionState(A, t+1) = susceptible*

AGENT-BASED SIMULATIONS

Similar simulation experiments as the ones described above have been performed using the model at the level of the individuals.

N	*10*
ContactFrequency	*0.8*
ContactIntensity	*0.5*
$\beta = 0.04$, $\gamma = 0.05$, $\rho = 1.25$	

As this model is based on random choices, the patterns can vary. In Figure 4 two example traces based on the following parameter settings are shown

N	*10*
ContactFrequency	*0.6*
ContactIntensity	*0.2*
$\beta = 0.012$, $\gamma = 0.1$, $\rho = 8.3$	

The infection states are indicated by numbers *0* (susceptible), *1* (infective) and *2* (recovered).
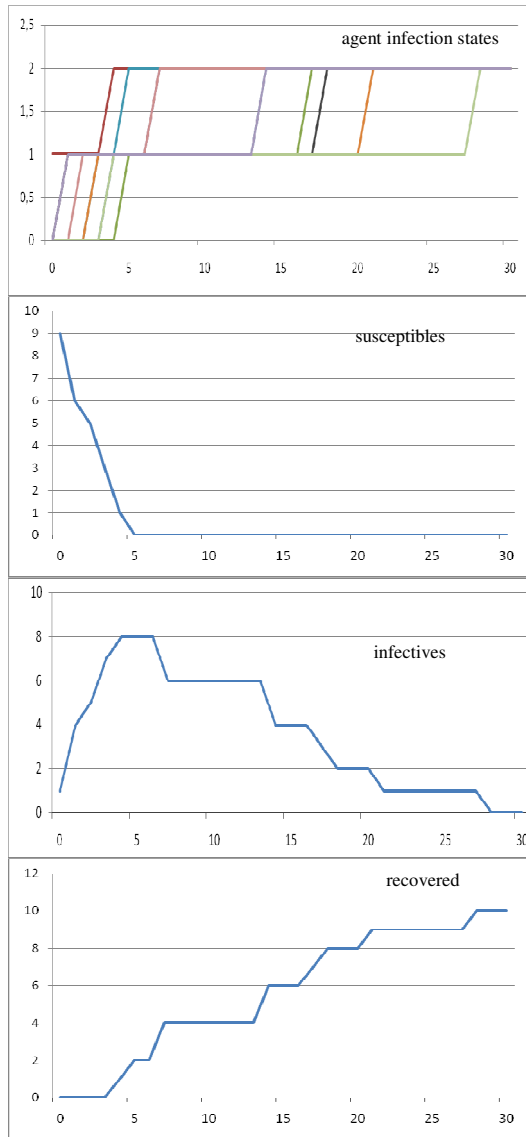
Figure 4: Pattern in which the Whole Population gets Infected

These settings correspond to the ones for the trace shown in Figure 2. Initially one agent is infective. In the upper graph it is shown how individuals get infected one after the other. At the start A2 is infective. Already from the second day on, A4, A7 and A10 get infected. After that A8, A5 and A6 follow, and finally A1, A9 and A3 get infected. In the second graph in Figure 4 the aggregated number of susceptibles is shown, in the third graph the number of infectives, and in the lower graph the number of recovered individuals.

The pattern is similar to the pattern shown in Figure 2. Note that also here the maximal number of infectives is reached at the time point that the number of susceptibles drops under ρ. Initially 1 agent is infective. In the upper graph it is shown how three individuals get infected. At the start A2 is infective. Soon A6 gets infected but recovers already in two days. Since A2 takes longer to recover, A8 is infected on day 5. After 6 days A8 recovers, and in the meantime also A2 recovered. No further infections took place.
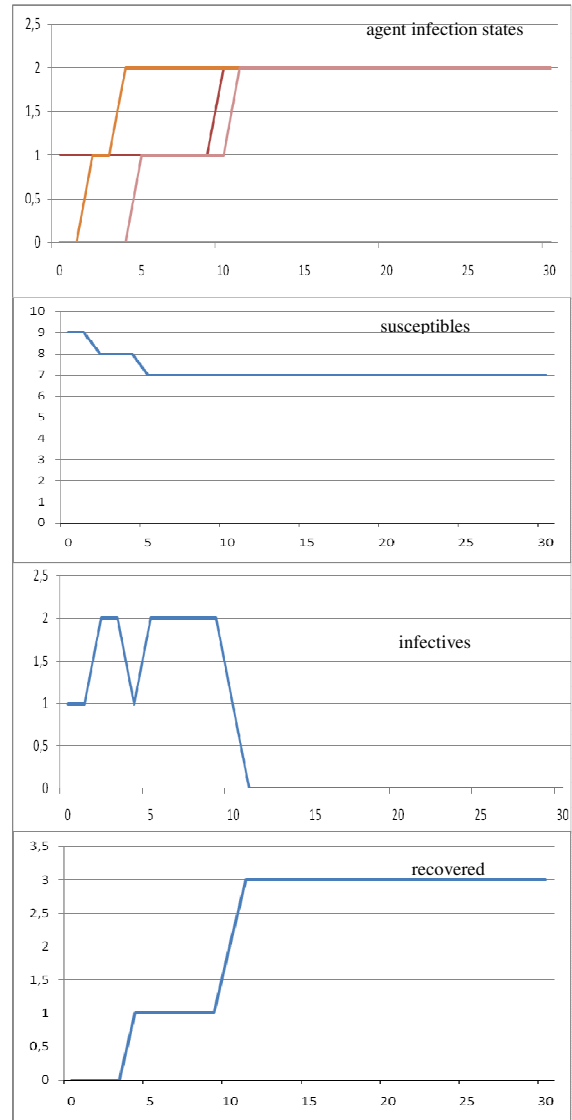
Figure 5: Pattern in which Part of the Population gets Infected

The pattern is similar to the pattern shown in Figure 4. Note that also here the maximal number of infectives coincides with the number of susceptibles dropping below the threshold ρ.

From other simulations it was found out that this example trace is a bit exceptional for this setting. Most traces of the individual model show either only one or two infectives, after which the epidemic dies out, or (almost) all individuals become infected. See Figure 6 for an overview of 100, resp. 1000 experiments with the model for 10 agents. The average number of recovered agents for this sample is 5.71. Note that this means that the model at the collective level shows a kind of average pattern that for the model at the individual level for 10 agents almost never occurs.
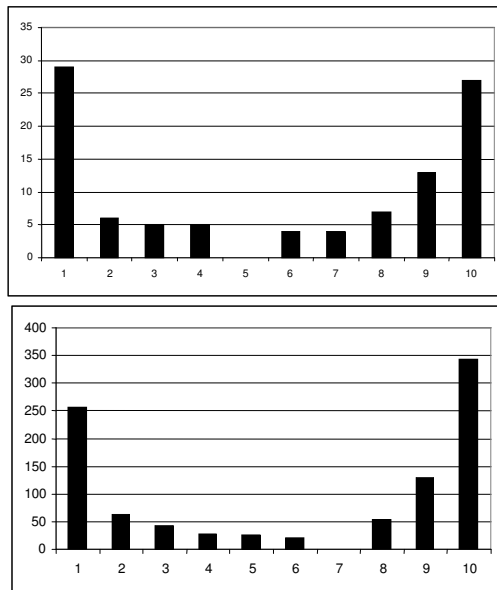
Figure 6: Average Numbers Recovered for 100 and 1000 runs for 10 agents with 10% initially infected resp.

Under similar experimental configurations simulations for larger population has been conducted through simulation software developed in the language C++. Considering the simplicity of the model, a procedural approach without multithreading was used in the software design. Figure 7 shows the results of 1000 simulations conducted for both populations of 100 and 1000 agents carrying 10 percent initially infected. These simulations have taken 0.02105 and 0.90953 seconds per simulation for 100 and 1000 agents respectively on a standard desktop computer. In these simulations the agent-based model shows a different pattern. Rather than an average pattern as for the case of 10 agents; see Figure 6, it shows single peak towards the higher number of recovered agents with an average of (approximately) 93 percent recovered agents in both cases. Variation in number of recovered agents for 1000 samples in case of population count 100 and 1000 was 31 and 9 percent respectively, which is much lower then 90 percent variation in all samples observed in population count 10. Moreover, the average on all simulations were also close to the peak that differs a lot

from both the outcome of agent based model at low population as 10 agents; See Table 1, and the population-based model.
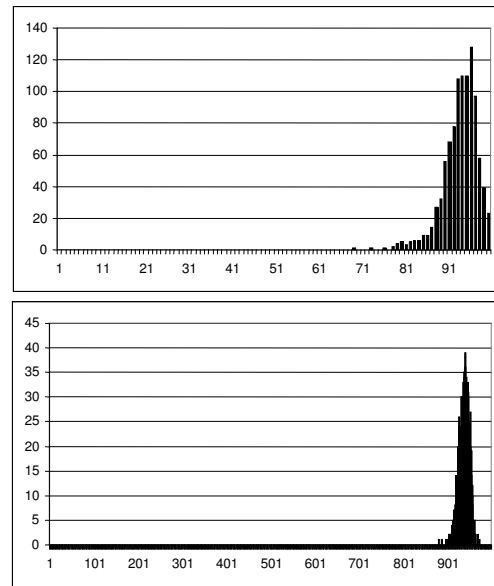
Figure 7: Average Numbers Recovered for 1000 runs for 100, and 1000 agents with 10% initially infected

Table 1: Percentage of Min, Max and Average Recovered in the Sample of 1000 Simulations for 10, 100 and 1000 Agents resp. with 10% Initially Infected

Total Population	10	100	1000
Initial Susceptible	9	90	900
Initial Infected	1	10	100
Average Recovered	61.62	93.50	93.83
Min Recovered	10.00	69.00	88.20
Max Recovered	100.00	100.00	97.30
Variation in Samples	90.00	31.00	9.10
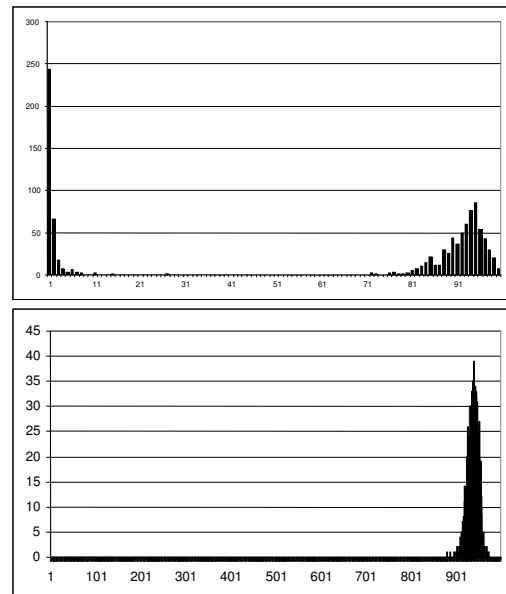
Figure 8: Average Numbers of Recovered Agents for a Sample of 1000 Simulations for 100 and 1000 Agents resp. with 1% Initially Infected

In Fig. 8 the results for similar populations have been shown with 1% initially infected. In these simulations for a population of overall size 100 an average of 60.5% recovered and 99% variation in all samples was observed which is somewhat similar in behaviour as of population count 10; see Fig. 6 that also shows two peaks with nearly average pattern. But the population of overall size 1000 has shown a graph almost similar to an average of 92.98% and 8.40% variation in all samples as that has been seen for 10% initially infected for the same population count; see Fig. 7. To further investigate this behaviour change as seen in a population of overall size 100 with a change of percentage of initially infected, simulations were performed for 0.1% initially infected for a population count of 1000. These simulations have confirmed the graph change pattern observed in case of population count 100; see Fig. 9.
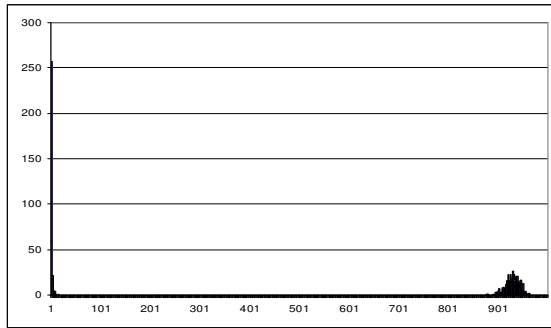
Figure 9: Numbers of Recovered Agents for a Sample of 1000 Simulations for 1000 Agents resp. with 0.1 Percent [1 Agent] Initially Infected

From the above simulations it is evident that in agent-based simulations for epidemics the percentage of the initially infected population is not the factor to be taken same for similar experimental configuration for different population sizes but it is the number of initially infected agents that should be taken same; see also Table 2.

Table 2: Average Percentage of Min, Max and Average Recovered in the Sample of 1000 Simulations for 10, 100 and 1000 Agents with 1 Agent Initially Infected

Total Population	10	100	1000
Initial Susceptible	9	99	999
Initial Infected	1	1	1
Average Recovered	61.62	60.4	58.5
Min Recovered	10.00	1.00	0.10
Max Recovered	100.00	100.00	96.40
Variation in Samples	90.00	99.00	96.30

From above it is found that in population-based modelling of epidemics a similar percentage of initially infected population yields in a similar percentage of recovered population for all total population sizes; see Table 3. But in case of agent-based modelling the initial count of the infected population the (approximately)

yields a similar percentage of recovered population for all total population sizes; see Table 4.

Table 3: Percentage of Recovered Population in Total Population Count [N] 10, 100 and 1000 Agents resp. with 10, 1 and 0.1 Percent Initially Infected [I(0)] in Population Based Simulation.

N/I(0)	10	100	1000
10%	55.620	55.620	55.620
1%	34.910	34.910	34.910
0.10%	27.070	27.070	27.070

Table 4: Average Percentage of Recovered in the Sample of 1000 Simulations for Total Population Count [N] 10, 100 and 1000 Agents resp. with 1, 10 and 100 Initially Infected Agents [I(0)] in Agent Based Simulation.

N/I(0)	10	100	1000
1	61.620	60.485	58.502
10	100.00	93.506	92.982
100	-----	100.00	93.830

Taking this number of initially infected agents rather then percentage of initially infected population as a parameter for similar experimental configuration is yet another subtle difference between population-based and agent-based simulation results of epidemics; see Table 3 and Table 4.

DISCUSSION

Papers addressing agent-based simulation of epidemics usually do not make a comparison between population-based models and agent-based models; see for example (Emrich, Suslov, Judex, 2007). Although in (Bagni, Berchi, and Cariello, 2002), a number of different types of models are briefly discussed, these models have not been compared by applying them to certain scenarios.

The comparative exploration of population-based simulation and agent-based simulation reported in this paper shows different phenomena that were not directly easy to predict. For the settings that were used, a large number of agent-based simulations based on only 10 agents provided an average of infected and recovered persons around 5 that is not far from the results of the population-based model with the same settings. However, the variation was very wide. It was very rare that the simulation came up with a result that is close to the average. On the contrary, almost half of them ended up in 1, and almost the other half in the maximal number of 10 recovered persons. For higher numbers of agents (100 and 1000) the outcome is completely different. For these cases the outcomes concentrate on the maximal number of infected and recovered persons; the variation is very low for these cases. Furthermore, the averages are also close to the maximal number of persons and therefore deviate a lot (around 100%) from the outcome of the population-based model with the same settings with average around 5. Based on the results the following can be noted:

- *Average outcomes do not match well*

The assumption that a population-based model shows the same results as the average of agent-based models only holds for the smaller number of agents, not for the larger numbers; this is the opposite as is sometimes assumed: that for larger numbers of agents the averages will approximate the outcome of a population-based simulation. This assumption is refuted by the simulation experiments.

- *Variation low for large numbers of agents*

However, for small numbers of agents the variation is so high that the average number gets less meaning.

- *Agent-based simulations more faithful?*

The answer on this question can be yes or don't know. Yes because a real difference is shown, so probably the agent-based model will be closer to reality. The answer can also be 'don't know' because it is not clear at forehand which of the two different outcomes is closer to reality. Possibly reality is even in between the two outcomes. To verify this, detailed empirical data have to be analysed, specific plans for this have not been made (yet) for this first explorative study.

- *The threshold law*

The threshold law shows in both the agent-based and population-based simulations.

Further work would be to take empirical data and compare the two types of models with these data. Moreover, the relationship between equilibria and the initial values for susceptibles and infectives can be explored further in the context of empirical data.

REFERENCES

Anderson, R.A., and May, R.M. (1992). *Infectious Diseases of Humans: Dynamics and Control*. Oxford University Press, Oxford, UK.

Antunes, L. and Takadama, K. (eds.) (2007). *Multi-Agent-Based Simulation VII, Proceedings of the Seventh International Workshop on Multi-Agent-Based Simulation, MABS'06*. Lecture Notes in Artificial Intelligence, vol. 4442, Springer Verlag, 2007

Bagni, R., Berchi, R., Cariello, P. (2002). A comparison of simulation models applied to epidemics. *Journal of Artificial Societies and Social Simulation*, vol. 5, no. 3, 2002.

Burghes, D.N., Borrie, M.S. (1981). *Modelling with Differential Equations*. John Wiley and Sons.

Davidsson, P., Gasser, L., Logan, B., and Takadama, K., (eds.) (2005). *Multi-Agent and Multi-Agent-Based Simulation, Proceedings of the Joint Workshop on Multi-Agent and Multi-Agent-Based Simulation, MABS'04*. Lecture Notes in Artificial Intelligence, vol. 3415, Springer Verlag, 2005.

Ellner, S.P., and Guckenheimer, J., (2006). *Dynamic Models in Biology*. Princeton University Press.

Emrich, S., Suslov, S., Judex, F. (2007). Fully Agent-Based Modelling of Epidemic Spread Using AnyLogic. *Proc. of the 6th EUROSIM Modelling and Simulation Conference, EUROSIM'07*, 2007.

Iwami, S., Takeuchi, Y., and Liu, X. (2007). Avian-human influenza epidemic model. *Journal of Mathematical Biosciences,* vol. 207 (2007), pp. 1-25.

Kermack, W.O., and McKendrick, W.G. (1927). A contribution to the mathematical theory of epidemics. *Proceedings of the Royal Society of London*, Series A 115, pp. 700-721.

Lotka A.J. (1924) *Elements of Physical Biology*. Reprinted by Dover in 1956 as *Elements of Mathematical Biology*.

Maynard Smith, J. (1974). *Models in Ecology*. Cambridge University Press, Cambridge.

Ross, R. (1916). An application of the theory of probabilities to the study of pathometry. Part I. *Proceedings of the Royal Society of London*, Series A 92, pp. 204-230.

Sichman, J.S., and Antunes, L. (eds.) (2006). *Multi-Agent-Based Simulation VI, Proceedings of the Sixth International Workshop on Multi-Agent-Based Simulation, MABS'05*. Lecture Notes in Artificial Intelligence, vol. 3891, Springer Verlag, 2006

Volterra, V. (1926). Fluctuations in the abundance of a species considered mathematically. *Nature* 118, 558-560.

Volterra, V. (1931). Variations and fluctuations of the number of individuals in animal species living together. In *Animal Ecology*. McGraw-Hill, 1931. Translated from 1928 edition by R. N. Chapman

Widmaier, E.P., Raff, H., and Strang, K.T. (2006). *Vander's Human Physiology: the Mechanisms of Body Functioning*, MacGraw Hill, 10th edition.

Xiao, D., and Ruan, S. (2007). Global analysis of an epidemic model with nonmonotonic incidence rate. *Journal of Mathematical Biosciences*, vol. 208 (2007), pp. 419-429.

AUTHOR BIOGRAPHIES

S. WAQAR JAFFRY was born in Lahore, Pakistan and went to the University of The Punjab Lahore, where he studied Computer Scinces and obtained his degree in 2001. He worked for a couple of years for the University of The Punjab Lahore before moving in 2007 to VU University Amsterdam where he is doing his PhD research in Artificial Intelligence. His e-mail address is swjaffry@few.vu.nl and his Web-page is at http://www.few.vu.nl/~swjaffry.

JAN TREUR is a full professor of Artificial Intelligence at VU University Amsterdam. He received his Ph.D. in Mathematics and Logic in 1976 from Utrecht University. Since 1986 he works in Artificial Intelligence, from 1990 as a full professor and head of the Department of Artificial Intelligence at the Vrije Universiteit Amsterdam. From the 1990s he headed a research programme on component-based design of knowledge-based and agent systems. In the last seven years the research programme focussed on modelling dynamics of agent systems in practical application areas, and related to other disciplines such as Biology and Cognitive Science. A current focus is on modelling human aspects in Ambient Intelligence applications. His e-mail address is treur@few.vu.nl; at http://www.few.vu.nl/~treur his Web-page can be found.

SIMULATION ENVIRONMENT FOR ARTIFICIAL LIFE AGENTS

Pavel Nahodil and Karel Kohout
Department of Cybernetics
Czech Technical University in Prague
Technická 2, 166 27 Prague, Czech Republic
E-mail: { nahodil, kohoutk }@fel.cvut.cz

KEYWORDS

Anticipation, Simulation, Hybrid Architecture, Animate, Behavior, Artificial Life, Artificial Creatures.

ABSTRACT

Our research is focused on simulation of agents - animates. The architecture of these agents is mainly inspired by nature; therefore they are sometimes called artificial creatures. The main contribution of this paper is the description of designed simulation environment architecture for the Artificial Life (ALife) domain. It was named the World of Artificial Life (WAL). Our platform incorporates also results of research in domain of hybrid agent architectures. Based on these results it focuses to the proposal and implementation of the simulation environment for them. First, we present the alternatives for our simulation environment which we evaluated and considered to use. Since we did not selected any of them further in the text we will formulate the problem we were solving and its important goals, followed by presentation of our proposal for such simulator and overview of achieved results.

STATE OF THE ART

This chapter is separated into two parts. First part describes the *Artificial Life* (ALife) field in general and discusses the current trends and streams. The second part is focused on the simulation environments for Artificial Life simulation.

Artificial Life

The trend in artificial intelligence leans towards communities of robots-agents. These structures appears in nature in all types of complexity starting from genes, cells, multi-cell structures, through plants, animals, groups of animals up to their societies. Similarly to the Nature also in robotics it is obvious that one super-intelligent robot (and therefore expensive) is with its abilities far behind a swarm of small, simple and less intelligent but also the less expensive robots. It is believed, that the power lies in quantity and simplicity. Also the range of possible types of tasks implemented on community of mutual cooperative robots is much wider than in case of one super-robot. Therefore approaches from *Artificial Life* are more often applied for control of community of robots. ALife approach (biological model) of implementation of intelligent

behavior is inspired mostly by nature phenomena, instead of classical artificial intelligence (rational approach) which is more concerned about logic, rationality and just partially on algorithms inspired by nature. Another significant difference between AI and ALife approach is in the object of interest. Artificial intelligence is focused on complex thinking for example chess playing, text understanding, disease diagnostic etc. ALife if on the other hand focused on basic elements of natural behavior with strong stress on survival in environment. The most of existing ALife approaches are based on algorithms, which enables robots as artificially created creatures, to evolve and adapt [Kadlecek, 2001]. Biological systems have inspired the development of a large number of artificial intelligence techniques such as neural network architectures, genetic and evolutionary programming, robotic and multi-agent systems. Each biological system brings a large amount of evolutionary baggage unnecessary to support intelligent behavior, but focusing not on content but on principles, the study of animal behavior can provide a lot of models that can be successfully implemented within a robotic or agent system. Due to the inherent complexity of these systems, a multi-level analysis approach supported with a lot of experiments and simulations is required. Animals, in contrast to the majority of agent applications, in which agents are highly specialized in terms of behavior, deployment environment, learning capabilities etc., incorporate broad set of behaviors and high level of adaptability, mobility, social capabilities, proactivity, reactivity, and are employing various learning methods in one system [Wooldridge and Jennings, 1995]. They have to provide wide set of behaviors such as predator avoidance, nesting, fighting, eating, exploring, sleeping, reproduction and others. With computational power of computers we have reached the point when we can imitate Mother Nature. ALife tries to model "*the world as it is*" or "*the world as it could be*". ALife gives us chance to test our comprehension of intelligence, adaptation for living conditions and evolution [Kadlecek and Nahodil, 2001].

The philosophy of classical MAS has been reversed in Artificial Life. ALife draws inspiration from many science disciplines such as biology, ethology, sociology, psychology, mathematics (grammars) and physics. ALife is often being connected with emergence. By mutual local cooperation of primitives a new phenomenon on global level arises. This is called

emergence and is achieved without any central control. The principle of superposition of primitives is not valid – nonlinear behavior of elemental primitives. Tools for evolution towards more complex and more perfect structures are self-reproduction, mutation and selection.

Simulation Environments

There is a lot of a freely available application or simulators for ALife domain available on the Internet. The issue with these is that almost all of them are focused on one phenomenon or one particular problem. There are not much general frameworks that would give wide range of possibilities. We are searching for a simulation environment that would give us enough freedom to program mind of the agent while the virtual world (including not only the graphical representation but also the functionality in terms of provision of sensoric input) are provided by the application. There are just few such applications available. We will mention here two of them that in our opinion were close enough to be used. First of them is StarLogoTNG which is being developed at Media Laboratory, MIT, Cambridge, Massachusetts [StarLogoTNG, 2008]. Creators describe this project as programmable modeling environment for exploring the workings of decentralized systems. With StarLogoTNG, you can model many real-life phenomena, such as bird flocks, traffic jams, ant colonies, and market economies. This application is the closest one for our requirements hence we will go in the detail of description. It is fairly new it was published only last year. The main goal of this application is to lower the barrier to entry for programming by making programming easier. This means that it provides graphical programming language (block building) and also powerful and editable 3D environment. This gives wide area of possibilities but for our purposes it was not wide enough. We would need to use some programming constructions that are not part of the provided set.

Another simulation environment widely recognized is Swarm [Swarm, 2008]. It is a software package for multi-agent simulation of complex systems, originally developed at the Santa Fe Institute. The basic architecture is the simulation of collection of concurrently interacting agents. It allows the researcher to describe agent behaviors one by one, agent by agent, context by context, all while keeping an exact notion of time and concurrency in the world. Swarm also makes it possible to compose or decompose hierarchies of agents. Swarm Code is Object-Oriented. The swarm libraries are written in a computer language called "Objective-C", a superset of the C language. Objective-C adds the ability to create software "classes" from which individual instances can be created. These instances are self-contained entities, and the terminology of object-oriented programming turns out to be very well suited to discussions of agent-based models. Most swarm applications have a structure that roughly goes like this. First, a top level often called the "observer swarm" is

created. That layer creates screen displays and it also creates the level below it, which is called the "model swarm". The model swarm in turn creates the individual agents, schedules their activities, collects information about them and relays that information when the observer swarm needs it. This terminology is not required by Swarm, but its use does facilitate it. Swarm libraries provide a number of convenient pieces of code that will facilitate the design of an agent-based model. These tools facilitate the management of memory, the maintenance of lists, scheduling of actions, and many other chores.

REQUIREMENTS AND GOALS

The main goal of our effort was to develop a simulator on a high modularity level and simple enough to be usable by anyone interested in the ALife research. Our simulator helped us to focus on the studied topic while abstracting from implementation details of the environment itself. Special care has been applied to possibilities of analysis, either during the simulation or after the simulation from saved data. Visualization modules are covering not only displaying the simulated agent world in 3D, but are targeted on efficient analysis of agents' behavior. Visualization can provide both simplified and attractive view in order to present the simulation either to broader or non-technical audience. Together with scientific views with various statistics and value details of the agent world, in or to satisfy needs of scientists for detailed analysis of behavior of the simulated system.

DESIGNED ABSTRACT ARCHITECTURE

In order to implement the set task a general abstract architecture was proposed and named WAL Abstract Architecture, WALA[2] in short. This should provide general guide or instructions, how an application for simulation of Artificial Life should be defined and implemented with care for high interoperability and modularity between various implementations. This design was not work of one person, but result of tight cooperation and many discussions. While designing this abstract architecture we kept in mind that our implementation can be superseded in future by better ones, but if it will stick with the philosophy and recommendations of the abstract architecture, agents will be transferable with none or minor reprogramming and redesign. Another goal of WALA[2] is to assure that agents can also run and compete on other implementations of same architecture. The reason for this is evaluation of results when different agent's architectures and approaches can be evaluated in base environment, or to simply test agents' behavior in different environment than he was designed for. We can observe if he can adapt and to new circumstances and survive. WALA[2] itself defines modular block architecture of platform as it is shown on Figure 1.
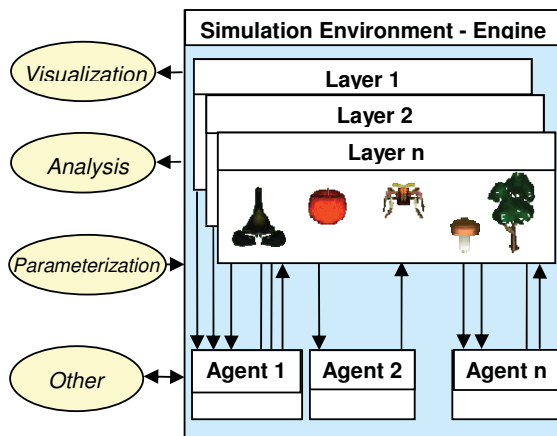
Figure 1: Block Scheme of WAL Abstract Architecture

Platform - Engine

The core part of the simulation environment will be referred to as the engine or platform. It is the base unit and it controls run of the simulation on a program level. This means it synchronizes the whole application – gives impulses on start and end of each step. It contains an interface for modules and it contains and maintains its parts. There are two components of the environment the layers and agents. In one simulation step engine asks all layers to evaluate actions of all agents and environmental changes according to these actions. The distribution of evaluation to layers means distribution of simulation control so that each layer can run in different computation thread (on multiprocessor unit they also might run on different processors). The main data structure where parameters of all layers and agents are stored is maintained by engine. This data can be view or modified by agent actions or even by external modules. It is important to distinguish between the control part of the engine, which interacts mostly with the operating system (graphical interface, loading and saving configuration, user interaction etc.) and the part providing and simulating the virtual world for agents. The first is done by above described engine. The second function is described further in the text and is handled by layers.

Engine Interfaces

Interface between simulation environment and its program surrounding (as for example visualization, analysis tool or parameterization) is important part of application. This is what makes WAL modular and distributable. From speed of data processing point of view it is suitable to exchange information in binary format. It is also possible to use text based formats such as XML. The textual format is in principle highly redundant (but descriptive) and its processing could be slow. Still, it can be used for offline analysis. Engine contains all its data, the data of layers and agents in inner tree-based data structure. It can provide all of

these data or just part of it to external modules. Each connected external module can ask for data and use them for its purposes. The inner data representation is not defined in abstract architecture. It can be implemented in various ways and it does not matter as long as the interface for exchanging of this data remains the same. This interface should work in both directions for exporting the data to be read by external modules as well as for receiving updates of the data structure from modules. The running simulation can also be stopped at any moment and even traced back to certain point and run again to observe if any change of behavior will occur in exactly the same situation. Change of simulation parameters should be available while the simulation is running. As an agent we understand any object in simulation either virtually alive (creature, predator) or virtually non-living (trees, food, water, rocks). Sensors and effectors of the agent are their interface with virtual world and therefore they are part of environment and layers. Besides that agent mind and control are not part of the environment and can be also remote. The binary format is good for both, as it can be parsed very quickly.

Simulated World in Layers

This is the part of the application directly interacting with the agent via his sensors and effectors. Layers define the virtual or simulated world in which agents live. It serves for logical and computational separation of operations which has to be controlled by engine. The layer is a logically separable part of the environment which can be used standalone and which when combined with others defines the environment as whole. All agents having influence in a particular layer or being influenced by this layer must be registered in it. This means that the layer has the full information for computing the next step. The layer will evaluate and execute agents' actions in each step. According to the executed actions of the agents the layer will modify its own values and then will provide a new sensory data to agents. The layer must have the ability to register and deregister the agent and also fill the sensors of those registered agents. This means, that it must have interface for communicating with sensors. For example the thermal layer should have the ability to provide information for sensors of temperature. In each step the layer must read agents executed actions, evaluate them (whether they are possible or executable), modify the environment and provide new sensory data. Basically there were two types of layers defined. The point layer in which can the value be evaluated directly or can be obtained immediately from layers data. The gradient layer where the value in the point of interest cannot be evaluated just from the actual information but also the history of the value must be taken in account. The physical body of the agent and his sensors and effectors are also part of the environment, so it is necessary to interpret them in it. The layer must have the

information how much space the body, sensors and effectors takes. This could enable building of more complex agent from basic blocks. The potential of the layers was then used in several works, when the implementations had up to seven different layers [Foltyn, 2005]. The advantage of layers is that they can solve communication between agents in sense of "physical" distribution of the signal. We can implement the acoustic layer, and propagate the sound based on the physical laws. This solves the problem of transporting the message, not the understanding and context of the message. To sum up, layers in one simple sentence, they implement various physical laws. Complete physical description is almost unachievable. Layers can bring us closer to it.

Human Interface and Analysis

All described above is just an algorithm with no human interface. Visualization of designed world can be both attractive and useful tool. For this purpose external visualization module or internal (default) can be used. Internal visualization is meant to debug and observe simulation by creator (Figure 2).
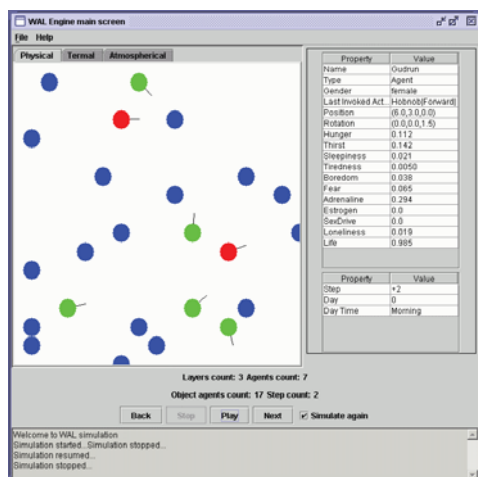
Figure 2: Example of Internal Visualization

The external module can be the exact opposite. It can be used to present this simulation to wider audience than the science community (Figure 3). On-line or off-line tools for parameters flow analysis are also supported. Proposed environment is compatible with highly effective 3D visualization tool called *Visualization Analytical Tool* (VAT in short). It can be used to observe agents parameters at any time of simulation. I will not go into much detail about the parameter visualization problem. Information about it can be found in [Kadlecek et al. 03]. By using the third dimension for data visualization the analysis is more comprehensive and computer graphics knows various methods to visualize even more than thee dimensions. That is why the 3D analytical tools are strongly

supported. This leads to many advantages, because the value of the parameter can be also mapped to shape, height, width or length or many others. Another advantage is possibility to use sensitivity analysis. Analytical tools provide offline or online evaluation of simulation together with fast orientation in complex situations. It also enables the possibility of backward analysis of interesting simulation. It can be also used to observe relations between sensory inputs and executed actions (i.e. what action was triggered under when there was a specific sensory input and vice versa).

Figure 3: Example of External Visualization

Agent of WAL

WALA[2] separates the body of agent (physical representation of agent) from the mind (behaviour control mechanism). The agent's body is part of the environment and therefore it is covered in environment architecture. The mind of an agent on the other hand communicates with the body through data from sensors. Please note that even agents own state has to be observed by sensors. This covers the state of agents sensors and effectors (some of them can be damaged or partial malfunctioning) and the Vegetative System Block (capturing the agent body hormone levels). After agents sensors are filled with data, the body sends them to the mind of agent, where they are processed. The way in which the data are being processed is not a subject of this abstract architecture. There are a large number of approaches for agent mind design (architecture). Examples of such architectures, widely recognized and published can be found in following works [Kadlecek and Nahodil 2001], [Foltyn 2005], [Mach 2005]. The mind evaluates the situation and selects the action or actions for execution (parallel actions are considered). The body tries to perform these actions using its effectors. The layer mentioned above will then evaluate the cause of actions by setting new sensory data. Agent's body is part of the environment and as such it has physical properties such as position, shape or temperature. The decomposition is shown on the next Figure 4.
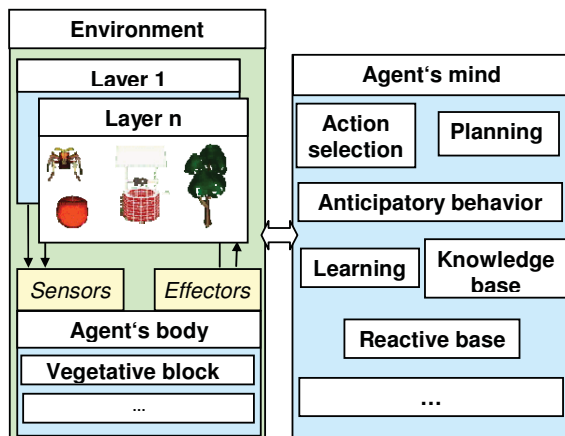
Figure 4: WAL Agent Decomposition

Agent's Sensors and Effectors

Sensors are the agent's senses which they use to perceive the surroundings and also its state. The richness of information about the surrounding world the agent can obtain depends just on the type and number of the sensors. Layers should know all types of possible sensors to be able to fill them with data. This means that creating a new layer necessarily requires also the creation of adequate sensors or altering the layer so it can work with current sensors and vice versa. When adding a new sensor it is also necessary to alter the layers so they know this sensor and are able to fill it. Effectors serves agent for interaction with its surroundings. We use simplified effectors. For example we use effectors of motion which can move agents in certain direction with certain speed. Of cause we could go deeper in detail and implement effectors such as leg or wheel, but this would distract us (by solving inverse and forward kinematics tasks etc...) from observing the behavior. We do not require this level of detail, but we are not running away from it, and the possibility to implement it is still open.

Communication

Communication between the agent body and the layers is internal communication so there is no need for explicit data sending. This communication can be done via the internal data structure, which is in this case the shared medium. The communication between body and mind is in general done via messages. It can be done even remotely via various media (for example over TCP/IP see Figure 5). Communication on the agent level means sending a message from one agent to another or group of agents. Here we would like to let layers decide who receives the information and who not. We are trying to reflect the real world behavior where information is carried via different media and can be received by various entities based on their sensor capabilities. When one agent wants to send a

message ("say something") to another it will use its effectors and certain media (acoustic wave). The information can be then received not only by the addressee but also by another agent who is in the range of the signal (even if it did not request this information). It can disregard it, or abuse it for its own purposes.
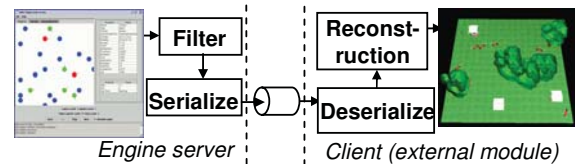
Figure 5: Block Scheme of External Communication

VISUALIZATION CASE STUDY

We would like to proof usefulness of external modules for simulation analysis on a case study performed while redesigning the agent to WAL environment. The simulation scenario concludes a single agent which was intended to move an object between two places. The agent had enough food and water to satisfy its needs. Figure 6 shows the visualized data from this simulation. On the left there is a 3D mesh; on the right is a detail of values in the 60th step. Even a very first look at the 3D mesh could advise that there is something wrong with the simulation. Almost all of parameters are zero (the mesh is flat) [Řehor, 2003].
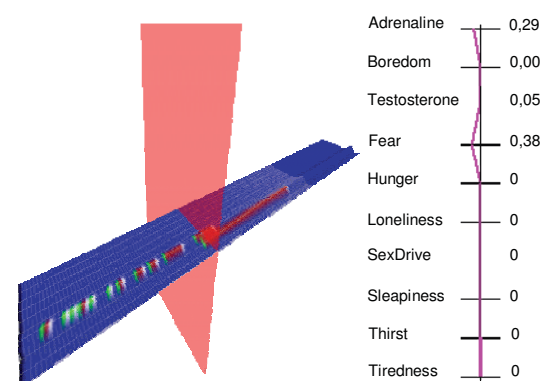
Figure 6: Visualized Data of Simulation -Using VAT

This means that the agent is neither hungry nor thirsty; it is neither tired nor sleepy. The reason for this could be a data export failure, a mistake in implementation of the inner agent vegetative block or a bad initial configuration of an agent. Because we run the simulation previously and export of data and the vegetative block were working properly, there is no problem with implementation itself. Brief check of the configuration showed that - there was to high value set to the time function for the chemical increasing/decreasing. Figure 7 shows the mesh after the configuration mistake correction.
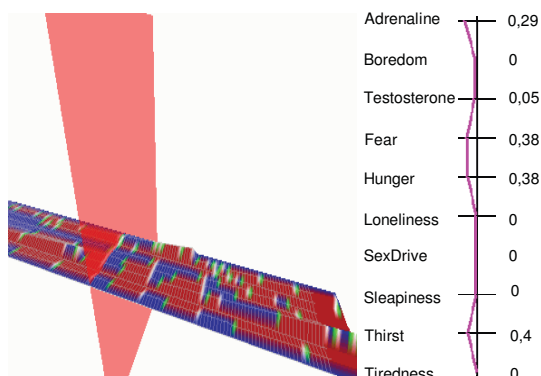
Figure 7: Use Case: Correct Setup – Using VAT

RESULTS DISCUSSION

It is very difficult to provide comparison of simulation environments. However the usefulness and applicability of the platform needs to be measured somehow in order to evaluate which parameters are better in our proposition from other works. The main difficulty is that there are two components to run the simulation. First of them is the hereby described environment, second is the control of the agent itself (agent mind). Comparing just the environments without agents would limit us just to naming the programming tools, used 3D engines and maybe reaction time of the application itself. This would not be enough to compare from the ALife simulation point of view. If we place an agent into such environment that it is again hard to distinguish between the qualities of an agent and qualities of the environment. The scientific way to solve this situation would be to create a reference agent and place it into multiple environments and then compare. This approach unfortunately faces technical difficulties in terms of modularity of compared applications, different implementation languages etc. This would lead to the abstracted comparison where we would need to define just qualities of the agent and implement these agents separately. Another objection might be raised here for objectivity of such approach. Different implementations would skew or results. Each programming language used for that has its strengths and weaknesses. To avoid these we would really need a very simple abstract agent. Would this agent test the qualities of the simulation environment? We believe not. This whole discussion leads us to conclusion that on the field of simulators especially from the Artificial life point of view we still do not have a comparison method. Also this topic will be in the scope of our further research. We would like to briefly introduce two architectures which are powered by WAL simulation environment and has been published lately.

Architectures Using WAL

One of the first architectures using the WAL environment was anticipatory agent architecture named Lemming [Foltyn, 2005]. It uses the algorithms known from Artificial Intelligence for agent's learning and offers an alternative to genetic programming and the artificial neural networks commonly used in ALife domain. This model ensures agent's survival in unknown dynamically changing environment. As you can se on the Figure 8 the visualization of the world changed but the engine is powered by the WAL architecture.
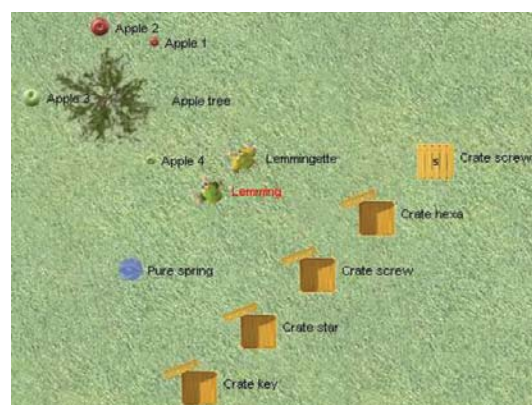
Figure 8: Architecture Lemming

Architecture is ACS proposed by M. Mach [Mach, 2005]. It uses successfully the Hidden Markov models combined by a reinforcement learning method.

REFERENCES

Kohout, K. *Simulation of Animates Behavior.* Diploma thesis. Prague: Czech Technical University in Prague, Faculty of Electrical Engineering, Department of Cybernetics, 2004.

Kadleček, D., Nahodil, P. *New Hybrid Architecture in Artificial Life Simulation.* In: Lecture Notes in AI 2159, Berlin Heidelberg: Springer Verlag, 2001. pp. 143-146.

Foltýn, L. *Realization of Intelligent Agents Architecture for Artificial Life Domain.* Diploma thesis. Prague: Czech Technical University in Prague, Faculty of Electrical Engineering, Department of Cybernetics, 2005.

Mach, M. *Data Mining Knowledge Mechanism of Environment Based on Behavior and Functionality of it's Partial Objects.* PhD thesis. Prague: Czech Technical University in Prague, FEE, Dept. of Cybernetics, 2005.

StarLogo on the Web [online]. Last revision Jan 2008 [cit. 2008-04-10]. < *http://education.mit.edu/starlogo-tng/*>.

Řehoř, D., Kadleček, D., Slavik, P., Nahodil, P. *VAT - A New Approach for Multi-Agent Visualization.* In: 3rd Int. Conference on Visualization, Imaging and Image Processing, 2003, Spain: ACME Press, pp. 849 – 854.

The Swarm Development Group [online]. Last revision Feb 2008 [cit. 2008-04-10]. < http://www.swarm.org/>.

Wooldridge, M. J. - Jennings, N. R.: *Intelligent Agents.* Lecture Notes in Computer Science Vol. 890, Berlin Heidelberg: Springer-Verlag, 1995, pp. 194 – 199.

Simulation of Complex Systems

SYNTHESIS OF SUSTAINABLE DEVELOPMENT SCENARIOS OF SOCIAL ECONOMIC SYSTEMS

Dmitry A. Kononov and Vladimir V. Kul'ba
Institute of Control Sciences
of Russian Academy of Sciences
65, Profsoyuznaya str. Moscow, Russia, 117806,
E-mail: kulba@ipu.rssi.ru

Jana Bikovska
Riga Technical University
1 Kalku Street
Riga, Latvia, LV-1658
E-mail: jana@itl.rtu.lv

KEYWORDS

Scenario, control, sustainable development.

ABSTRACT

This paper presents basic principles of a scenario approach application to the problem of optimal development of social economic systems. The formalized description of scenario of system sustainable development is introduced and synergetic and attractive scenarios are examined. The modes providing stable and safe development of social economic system are investigated for synergistic scenarios, however for attractive scenarios it is the control purpose. Scenario design example based on the use of the social economic system development model represented by signed the graph language are given. Formal methods of such scenarios development, the "Object-subject" method and the method of base scenarios, are proposed.

INTRODUCTION

The analysis of the behavior of dynamic processes, appearing in socioeconomic systems under the influence of disturbances of different nature, is an important step of formal decision-making procedures for a wide range of control problems in the systems for providing predetermined security level. Typical tasks of this kind are tasks that appear under investigation of not quite well-structured and non-structured problems with low quality of the initial data and with qualitative description of dependences. Getting a numerical solution with the precise normative models in this case is unreasonable. It is applied in full to the problem of working out of a strategy of regional, national and international crisis-free development. Under such conditions the role of the analysis methods that allow by information available about structural features of an investigated system to judge about appropriate development directions of a dynamic processes, their sustainability and other features, becomes very important. Such a task can be solved by synthesis of scenario of a social economic system (SES) development (Kononov et al. 1999).

Recently, the signed graphs have become a popular tool for analyzing SES functioning and control. They allow working with quantitative and qualitative data and can be effectively applied for a system behavior scenario

synthesis. The signed graph model extension to the directed operator graph model allows one to formally develop synergetic and attractive scenarios. They are hypothetic discrete movement trajectories of the modeled system in a phase space of variables (factors), based on the information about system structure and control actions performed (Kononov et al. 2001). The proposed method consists of an approximation of SES development trends by trajectory fragments, defined by impulse processes on the directed graph. Here the basic tendencies of the system development are defined without external influences (synergetic scenarios) or with predetermined control influences (attractive scenarios - direct control task).

If the observed parameter characteristics are negative, it is possible to solve a reverse task – to find effective control decisions and their application to the model (generation of attractive scenarios). As a possible option, static or dynamic control approaches could be applied. The static control approach does not change relations between structural elements; however, the dynamic control approach changes the model structure of SES development.

THE CONCEPT OF A SUSTAINABLE DEVELOPMENT SCENARIO

The problem of complex systems safety control appears in different fields and it relates to the development of system object of different levels. There are many different kinds of complex systems that could be classified by fields of activity, applied research methods, possibility of application and consequences of one or another control influence, ranging from a particular individual to the objects of global scale. A major feature of SES is absence of an emergency situation during its functioning and development process.

Safety as an aggregation of system functioning properties, including stability, adaptability, vitality, etc., can be formally described by the presence of the following conditions:
- possibility of determination and retention of some system parameters in the defined area of space, i.e. *observability and controllability*;

- exclusion of crisis situation appearance in SES that could destroy its elements and/or relations between them;
- possibility of guaranteeing the movement of an object within the phase space in the set direction, for example, in the direction of the strong and balanced growth, i.e. the problem statement and its effective solution about termination, sequencing or continuation of a system behavior scenario along the defined trend.

From the scenario-based research point of view, the safety can be formalized in terms of events and situation.

Expected event of a system behavior is defined as $\Im = \left(\mathbf{x}(t), \mathbf{y}(t), t\right)$, where t - a time instant, $\mathbf{z}(t) = (\mathbf{x}(t), \mathbf{y}(t))$ – expected realization of the environmental parameters and the phase trajectory.

Situation $S(t)$ at the time instant t is defined as a set of events which occurred before the moment t :

$$S(t) = \{\Im^{(i)}\left(\mathbf{x}^{(i)}(t_i), \mathbf{y}^{(i)}(t_i), t_i\right)$$

$$0 \le t_i \le t, \ i = 0, 1, ..., s; \ t_0 = 0\}$$

Condition $I(t)$ at time instant t is defined as $(S(t), \Theta(t))$, where $S(t)$ - situation at time instant t , $\Theta(t)$ - quasi-informational hypothesis of a decision-making person.

Scenario $\Re$ ***of system behavior*** is defined as a pair $\left(I(t_i), t_i\right)$, composed according to $A^{(t)}$ choice law:

$$\Re = \Re\{\left(I(t_i), t_i\right)\} \ i = 0, 1, ..., N; t_0 = 0.$$

The value N is called ***scenario depth***, and $T = t_N$ - ***scenario horizon***.

To apply the scenario-based research SES it is important to select the events significant to an expert which determine the features of a model of fixing of events in the set subject domain on the base of the special knowledge of the experts (Kononov and Kul'ba 1996).

Definition 1. *Situation* $S(t)$ *is called crisis-free, if*

$$\mathbf{z}(t) \in \text{int}\,\widehat{Q} \qquad (1)$$

for all events from $S(t)$ significant to expert, i.e. the generated sequence of the states is within the set of $\widehat{Q}$ possible states.

The definition of a crisis-free situation dependent on the way of determining $\widehat{Q}$ could be different.

Let a scenario $\Re$ with horizon T be defined.

Definition 2. *Scenario* $\Re$ *is called the scenario of sustainable development, if for any* $\tau = \overline{0, T}$ *situation* $S(\tau)$ *is crisis-free.*

The scenarios of sustainable development are characterized by the most desirable methods of SES development, saving its homeostasis.

SCENARIO-BASED RESEARCH OF SUSTAINABLE DEVELOPMENT IN THE MODELS OF SES BEHAVIOR WITH SIGNED DIGRAPH

Let us consider a scenario research of SES when its behavior model is introduced as an operator graph. A dynamic of the system behavior is set on the graph $G(X, E)$ where in each node $x_i \in X$ ($i \in \widehat{N} = \overline{1, n}$) the parameters' column $\mathbf{v}^{(i)} \in E^P$ ($i \in \widehat{N}$), which determines a current state of the system, is defined.

At time instant $t = 0$ parameters have initial value $\mathbf{v}^{(i)}(0)$. Graph $G(X, E)$ and a set $\widehat{F} = \{\overline{F}^{(p)} \ 1 \le p \le P\}$ of adjacency matrix $\overline{F}^{(p)}$ define structure of the transient, which can be observed as a sequence of transformations of a condition of system parameters by the rule:

$$\mathbf{v}_p(t) = \zeta^{(p)}\left(\widehat{F}, \mathcal{G}, \mathbf{v}_p(t-1)\right) \text{where } 1 \le p \le P, \quad (2)$$

where $\mathbf{v}_p(t) = \{v_p^{(1)}(t), v_p^{(2)}(t), ..., v_p^{(n)}(t)\}$ is a vector - line of values of the parameter p , observed in nodes of the directed graph at time instant t; $\mathcal{G}$ - detected or applied control influence on SES elements.

An operator digraph, in which intermatrix relations $\overline{F}^{(p)}$ in the operator $\zeta^{(p)}$ are missing, is called operator digraph with independent components (IC-operational graph). IC-operator graph describes structure and interaction of base elements of SES, as a rule, over the range of one stratum fields of the description and control.

The sequence of change of parameter $\mathcal{G}$ over time leads to a sequence of changes of a condition of the system parameters $\mathbf{v}^{(i)}$ that form the determined step-to-step scenario $\Re$.

The multitude of scenarios of SES behavior on operator graph is determined by the strategy of its formation. Two types of such strategy are considered: operationally-tactical - k-step sequence of impulses $\mathbf{Im}(t,k) = \{OI_0^{(i)}(\tau)_{i \in \hat{N}}\}_{(\tau = \overline{t,t+k})}$ and strategic - k-step sequence of SES structural transformations $\Phi(t,k) = \{A(\tau)\}_{(\tau = \overline{t,t+k})}$, that starts at time instant t.

To each of the chosen sequences there corresponds the linear time step-to-step scenario as a time sequence of step-to-step events significant to expert.

For the IC-digraph $G(X, A, \zeta)$ on the horizon T the sequence $\Phi(0,T)$ is constant, i.e. $A(\tau) = A$ for all $\tau = \overline{0,T} = \hat{T}$, and T-step impulse process is defined as follows:

$$\mathbf{Im}(0,T) = \{\mathbf{OI}_0(\tau) = (OI_0^{(i)}(\tau))_{i \in \hat{N}}\}_{(\tau \in \hat{T})}. \quad (3)$$

It is by virtue of a dynamic model

$$\mathbf{OI}(\tau) = \mathbf{OI}(\tau - 1) A + \mathbf{OI}_0(\tau) \ (\tau = 0,1,\ldots), \quad (4)$$

$$\Delta \mathbf{v}(\tau) = \mathbf{v}(\tau) - \mathbf{v}(\tau - 1) = \mathbf{OI}(\tau - 1) A + \mathbf{OI}_0(\tau), \quad (5)$$

where $\mathbf{OI}(\tau)$ is the impulse, which has been cumulated up to the time instant τ, corresponds to the determined time step-to-step scenario with the beginning in the expert significant event $\mathbf{v}(0)$:

$$\Re(0, T, \mathbf{v}(0)) = V(\hat{T}) = \{\mathbf{v}(\tau) = (v^{(i)}(\tau))_{i \in \hat{N}}\}_{(\tau \in \hat{T})} \ (6)$$

By a threat the presence of a hypothetical opportunity of realization of the undesirable phenomena could be called. Describing SES as a formal system object we will define a threat as a set of system and environment factors that could lead to deterioration of a condition of its system parameters from the decision-maker point of view. There are internal and external threats. For impulse processes on operator graph the following definitions are given:

The set of available impulses is given $\Omega \mathbf{Im} \in E^n$.

Definition 3. *T-step impulse process* $\mathbf{Im}(0,T)$ *is called admissible, if*

$$\mathbf{OI}_0(\tau) \in \Omega \mathbf{Im} \ \text{for all} \ \tau \in \hat{T}. \quad (7)$$

Definition 4. *At realization of admissible impulse process* $\mathbf{Im}(0,T)$ *conditions* $\mathbf{v}(0)$ *contain* T-step *threat, if*

$$\mathbf{v}(T+1) \notin \text{int} \, \hat{Q}(\tau + 1). \quad (8)$$

Definition 5. *Condition* $\mathbf{v}(0)$ *is called* T - *critical if it contains* T-*step threat at realization of any admissible impulse process*

$$\mathbf{v}(T+1) \notin \text{int} \, \hat{Q}(\tau + 1). \quad (9)$$

A 1-critical conditions is called emergency.

The scenario-based research of safe development of a system (3)-(6) can be carried out on the base of various models depending on the sets of allowable conditions $\hat{Q}(\tau)$ and strategy of behavior, which are permissible impulse processes.

THE MODEL OF KYOTO PROTOCOL

As an example of the signed digraph methodology application, the mechanism of ecological risk management based on the market distribution of quotas of environment pollution among countries-pollutants is considered.

According to the international contract, there is a mechanism of strict regulations of cumulative emission of harmful substances in an environment from the side of all industrial enterprises of the countries-participants of the contract (the Kyoto Protocol). It means that each participant can get in the world market a quota on a manufacturing pollution in a certain amount. In abstract case of the model, it is considered that if the participant for some reasons exceeds the quota limits provided his work is stopped. As the quotas distribution is a market mechanism, quotas can also be objects of sale and purchase. A competitive struggle can be developed around quotas. It is essential that the level that limits cumulative pollution issue is not defined by a demand for quotas in the market - international agreements determine them.

As the base processes that describe the considered mechanism, two subsystems are considered for their qualitative analysis:
1) subsystem "Socium":
   - change of an environment quality;
   - change of a life quality;
   - change of risk of an occurrence of an extreme ecological situation.
2) subsystem "Industry":
   - industry growth;
   - change of the level of competitive struggle for pollution quotas.

The following parameters describing phase states of base processes are considered:
   Q - population life quality;
   PO - number of inhabitants;
   W - number of workplaces;

I – pollution level;

H - risk of occurrence of an extreme ecological situation;

N – enterprise number;

C - level of competitive struggle;

P - the price on generalized resources - quotas;

E - efficiency of use of the generalized resource;

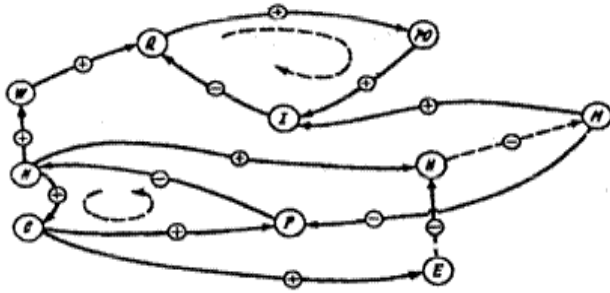M - allowed marginal level of issue of harmful substances.

Figure 1: Graph of "Kyoto" Model

The system model is constructed as a signed digraph (see Figure 1), where nodes are the base processes and parameters that characterize them. The interference of processes determined by the socioeconomic and natural laws is represented by realization of edges with the appropriate signs. At construction of the model it is necessary to take into account time parities (ratio) between the basic and caused impulse processes. In the model, under consideration an influence transfer time between base processes is accepted to be equal to the length of the appropriate way between them in the signed digraph.

For the constructed model, sufficient conditions of impulse stability are unsatisfied. It means that at occurrence of an impulse in any node of (N, C, P) cycle there is a linear increase in the module of an impulse in nodes of a cycle (Q, PO, I), i.e. a linear resonance. This phenomenon can have the following interpretation: fluctuations in amount of the enterprises conduct the fluctuations in an occupation level of the population that "shakes" processes of migration. Increasing fluctuations on amplitude in a population attract growth of environmental contamination due to household pollution. Computing experiment shows the increase in fluctuations amplitude.

Assume that socium cannot remain indifferent to changes of risk of extreme ecological situation. Within the framework of the offered formalized model it is possible to use various alternatives of operationally-tactical and strategic management. As strategic management it could mean drawing an edge or further edges from the node H. The solution when the negative edge (H, M) is added to the model could be considered as reasonable in case of risk increase of emergency solution.

In this case it also does not solve the problem: a resonance takes place again. Practically it means that in any possible way it is reasonable to complicate influence of rival elements on market mechanisms of ecological risk management.

Investigation of direct impulse and structural control approaches enables one to get typical modes of system behavior with simulation model.

CONSTRUCTION OF THE SCENARIOS OF SUSTAINABLE DEVELOPMENT WITH THE "OBJECT-SUBJECT" METHOD

This section discusses the method of optimal control of the distribution process of indignations by allocation of nodes-objects and nodes-subjects. The node-object corresponds to a parameter, whose value has to be controlled. The node-subject corresponds to a parameter, by influencing which this controlled change outstripping in time is carried out. Control is accomplished by means of preventive inclusion into nodes-subjects a sequence of external impulses that form the attractive adaptive scenario of safe development $\mathfrak{R}^{(sec)}\left(T^{(sec)}\right)$ on the horizon $t \in [t_0, t_1] = T^{(sec)}$. At construction of the safe development scenario $\mathfrak{R}^{(sec)}\left(T^{(sec)}\right)$ it is necessary to specify:

- *adaptive strategy construction factors:*
  - a set $\mathbf{g} = \{ g_j \in \widehat{N}, j = \overline{1, g} \}$ of nodes-objects, so $V_{ob} = \{ v^{(l)} \mid l \in \mathbf{g} \} \in R^g$ - set of values of their parameters;
  - allowable areas $Q_{ob}\left(t^{(l)}\right)$ ($l \in \mathbf{g}$) of values of the given characteristic $\mathbf{h}\left(V_{ob}, t\right) \in Q_{ob}\left(t^{(l)}\right)$ nodes-objects V_{ob} at time instant t;
  - a set of time moments $T^{(l)} \subseteq T^{(sec)}$ ($l \in \mathbf{g}$) in which restrictions for values of parameters in tops - objects are entered;

- *control subjects:*
  - set $\mathbf{q} = \{ q_j \in \widehat{N}, j = \overline{1, q} \}$ of nodes-subjects, so $V_u = \{ v^{(u)} \mid u \in \mathbf{q} \} \in R^q$ - set of values of their parameters;
  - set of time moments $T^{(u)} \subseteq T^{(sec)}$ ($u \in \mathbf{q}$) influences on nodes-subjects;
  - sequence of operational and tactical impulses in node $u \in \mathbf{q}$

$$\mathbf{Im}^{(u)}\left(T^{(u)}\right) = \{OI^{(u,0)}\left(\tau\right)\}_{(\tau \in T^{(u)})} \in Q^{(u)}\left(\mathbf{Im}\left(T^{(u)}\right)\right), \quad (10)$$

concentrated in the set of allowable impulses $Q^{(u)}\left(\mathbf{Im}\left(T^{(u)}\right)\right)$;

- *control purpose:*

deduction of values of the parameters V_{ob} appropriate to nodes-objects, in the allowed area

$$h^{(l)}(V_{ob},t) \in Q_{ob}(t^{(l)}) \text{ at } t^{(l)} \in T^{(l)} \qquad (11)$$

for all $l \in \mathbf{g}$.

Thus, the control task is formulated as follows: the initial expert significant event $\mathbf{v}(t_0)$ is defined. It is necessary to construct the attractive T-step-to-step scenario $\Re^{(\text{sec})}(T^{(\text{sec})})$ of safe development, i.e. to carry out synthesis of optimum control (10) as a sequence of T-step-by-step expert significant events

$$\Re^{(\text{sec})}(T^{(\text{sec})}) = \Re^{(\text{sec})} \{(I^{(\text{sec})}(t),t)\} \quad t \in T^{(\text{sec})}, \quad (12)$$

satisfying conditions (11). This task is a reverse control task.

Let's consider solving the task under the following conditions:

$$\mathbf{h}(V_{ob},t) = \mathbf{v}(t) - \mathbf{v}(t-1) \quad t \in T^{(\text{sec})}, \qquad (13)$$

$$Q_{ob}(t^{(l)}) = \{ v^{(l)}(t) \mid h^{(l)}(V_{ob},t) \le 0 \qquad (14)$$

at $t^{(l)} \in T^{(l)}$ for all $l \in \mathbf{g}$ },

that describes preservation (not increase) of the parameter values in the given area for nodes-objects at certain time moments $T^{(l)}$;

$$Q^{(u)}(\mathbf{Im}(t_0,t_1)) = \mathbf{L_q}(\mathbf{Im}(t_0^{(u)},t_1^{(u)})) \qquad (15)$$

$u \in \mathbf{q}$, $t^{(u)} \in T^{(u)}$,

i.e. control tools are any impulse processes begun in nodes-subjects $\mathbf{q}$ on the interval $T^{(u)}$.

It is possible to show that for processes (3)-(6) the following holds:

$$\mathbf{v}(t) - \mathbf{v}(t-1) = \sum_{\tau=0}^{t} \mathbf{OI}_0(\tau) A^{t-\tau} \text{ for} \qquad (16)$$

$t = 1,2...$, or

$$v^{(l)}(t^{(l)}) - v^{(l)}(t^{(l)}-1) = \sum_{u=1}^{n} \sum_{\tau=0}^{t^{(l)}} OI_0^{(u)}(\tau) a_{ul}^{t^{(l)}-\tau},$$

where $a_{jl}^{t-\tau}$ is the (j,l)-th element of the matrix $A^{t-\tau}$. Hence, conditions (13)-(14) will be transformed to

$$\sum_{\tau=0}^{t^{(l)}} \sum_{u=1}^{n} OI_0^{(u)}(\tau) a_{ul}^{t^{(l)}-\tau} \le 0 \text{ for all } l \in \mathbf{g} \quad (17)$$

or

$$-(\sum_{u \in \mathbf{q}} \sum_{\substack{0 \le \tau \le t^{(l)} \\ \tau \notin T^{(u)}}} OI_0^{(u)}(\tau) a_{ul}^{t^{(l)}-\tau} +$$

$$\sum_{u \notin \mathbf{q}} \sum_{\tau=0}^{t^{(l)}} OI_0^{(u)}(\tau) a_{ul}^{t^{(l)}-\tau}) = b^{(l)} \quad l \in \mathbf{g}$$

Thus, the solving of the stated reverse task is reduced to solving of system of linear inequalities with regard to variables $OI_0^{(u)}(\tau)$ ($u \in \mathbf{q}$, $t \in T^{(u)}$):

$$\sum_{u \in \mathbf{q}} \sum_{\substack{0 \le \tau \le t^{(l)} \\ \tau \in T^{(u)}}} OI_0^{(u)}(\tau) a_{ul}^{t^{(l)}-\tau} \le b^{(l)} \quad l \in \mathbf{g}. \qquad (18)$$

SYNTHESIS OF SCENARIOS OF SUSTAINABLE DEVELOPMENT BY THE METOD OF BASE SCENARIOS

The method of optimum control of indignation distribution process by distinguishing nodes-objects and nodes-subjects assumes that the area of operating influences is a linear space, i.e. is calculated on use of infinite admissible pulse processes, which in real life is hardly possible. Besides, the method of its decision has an essential shortcoming: the received SES sequence, though the scenario of T-safe development, but for any δ>0 contains δ-step threat. At the same time on the base of the results described in (Kononov et al. 2007) the following problem can be solved.

The initial expert significant event $\mathbf{v}_0(t)$ is set. To construct an attractive scenario of δ-sustainable T-safe development $\Re^{(\text{sec})}(T^{(\text{sec})},\delta)$, i.e. to carry out synthesis of the optimum scenario in the form of sequence step-to-step events significant to experts

$$\Re^{(\text{sec})}(T^{(\text{sec})},\delta) = \Re^{(\text{sec})} \{(I^{(\text{sec})}(t),t)\} \quad t \in T^{(\text{sec})}, \quad (19)$$

satisfying condition (9).

The mathematical reason to use the method consists in the following theorems.

Theorem 1 (on the splitting of the expanded phase space). On directed graph of the dimension n for each matrix A there exists a finite canonical structure $\Xi(A)$, representing expert-significant partition of the expanded phase space Z

$$\Xi(A) = \{ Z^{(k)} \quad k = \overline{1,s} \}, Z^{(k)} \cap Z^{(l)} = \varnothing \qquad (20)$$

with $k \ne l$, $Z = \sum_{k=1}^{s} Z^{(k)}$.

Each of these elementary partitions $Z^{(k)}$ is absorbent for any independent pulse process that was started in it (Kul'ba et al. 2004).

Thus, for each matrix A it is possible to specify in advance the modes of the SES concentration generated by the pulse processes, i.e. *to compute* (rather than to

indicate by expert) expert- significant partition $Z^{(k)}$ ($k = \overline{1,s}$) the expanded phase space Z .

The parameters grouping k -root subspace, are connected. The specified groups set SES stratification.

Theorem 2 (About basic modes of pulse scenarios of SES development). There are the following types of basic scenarios: resonant development scenario (RD-scenario), compressive development scenario (CD-scenario), fluctuation development scenario (FD-scenario), resonance-fluctuation development scenario (PFD-scenario of type "a untwisted spiral"), compressing-fluctuation development (CFD-scenario of type "a twisted spiral"). It is possible to extract also their extreme cases: the stationary scenario and the reflection scenario.

Theorem 3 (About modes of impulse scenarios of SES development). The expanded phase space can be presented as follows:

$$Z = \bigcup_{k=1}^{s} Z^{(k)} , \ Z^{(k)} \cap Z^{(l)} = \varnothing \text{ where } k \neq l , \quad (21)$$

i.e. s non-crossed subspaces, in each of which basic modes of development SES are defined.

Any step-to-step expert significant event can be represented in the form of

$$\mathbf{v}(t) = \sum_{j=1}^{s} \mathbf{v}^{(j)}(t), < \mathbf{v}^{(j)}(t), \mathbf{v}^{(k)}(t) >= 0, (22)$$

$$j \neq k , \ j, k = \overline{1, s} ,$$

where $\mathbf{v}^{(j)}(t)$ is concentrated in j -th EIE $Z^{(j)}$.

Theorem 4. Any step-to-step determined scenario corresponding to the set of impulse process on constant signed digraph, can be represented as a linear combination of base scenarios of SES development.

The presented theorems allow one to completely solve problems of the reverse scenario control on signed digraphs and to build for their decision effective computing schemes, in particular, for restrictions (11) with various functions $h^{(l)}(V_{ob}, t)$, and also at the defined sets of allowable impulses $\Omega \mathbf{Im}$.

CONCLUSIONS

Nowadays sustainable development of social economic system is a primary task of its existence. In the paper is described scenario-based approach to a research of social economic systems behaviour that allows synthesizing synergistic and attractive scenarios of a system development in order to prevent and control possible emergency situations. As the example, a model of the Kyoto Protocol is developed with signed directed graph and possible control actions and their consequences are discovered.

REFERENCES

Kononov D.A., Kosyachenko S.A., and Kul'ba V.V. 1999. "Analysis of scenarios of development of socioeconomic systems in emergency control systems: models and methods". Avtom. And Rem. Cont., Vol.90. Part 2. No. 9, pp. 1303-1320.

Kononov D.A., Kosyachenko S.A., and Kul'ba V.V. 2007. Design and analysis of development scenarios of social economic systems with the application of the operator graph apparatus, *Automation and Remote Control, 2007, vol. 68, No. 1. pp. 109-123.*

Kononov D.A. and Kul'ba V.V. 1996. Ecological management: Object development scenarios and control of ecological environment. *Inzh. Ekol.,* No 6**.**

Kononov D.A., Kul'ba V.V., and Shubin A.N. 2001. *Stability of socioeconomic systems: scenario investigation methodology*, Preprints Vol., pp.91-96. 8th IFAC Conference of Social Stability: The Challenge of Technological Development; SWISS'01 27-29 Sept. 2001; Vienna, Austria.

Kul'ba V.V., Kononov D.A., Kosyachenko S.A., and Shubin A.N. 2004. Methods of formation of scenarios of development of social and economic systems [in Russian]. *AV.A. Trapeznikov Institute of Control Sciences, Russian Academy of Sciences,* Moscow.

AUTHOR BIOGRAPHIES

VLADIMIR V. KUL'BA, Academician, Dr.habil.sc., Professor, Trapeznikov Institute of Control Sciences Russian Academy of Sciences. He has made significant contribution in areas of control theory, optimal modular data processing, local and distributed data bases, network computing systems, and others.

DMITRY A. KONONOV, Dr.Math., Professor, Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, Moscow, Russia. His research area is development of mathematical models and methods for analysis and synthesis of complex systems' scenario.

JANA BIKOVSKA, Mg.sc.ing, Ph.D. student, teaching assistant at the Department of Modelling and Simulation, Riga Technical University, Latvia. Her professional interests are in the field of supply chain management, simulation based business games, scenario approach in complex systems with application to simulation games.

SIMULATION OF THE IMPACT OF THE SOIL FRICTION ON THE DRAFT RESISTANCE OF THE PLOUGH BODY

Arvids Vilde and Adolfs Rucins
Latvia University of Agriculture, Research Institute of Agricultural Machinery
1 Instituta Street, Ulbroka, LV-2130, Latvia
e-mail: vilde@delfi.lv; arucins@delfi.lv

KEYWORDS

Plough body's draft resistance, friction 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 determine the draft resistance of the plough depending on the parameters of its body, as well as to evaluate the impact of the physical and mechanical properties of soil, such as friction upon it. The greatest influence upon the draft resistance is exerted by soil hardness, density and slip resistance along the surfaces of the operating parts. The latter is also affected by soil adhesion, which particularly manifests itself in wet clay soils at lower temperatures. It has been clarified that the friction resistance constitutes 46 – 62 % of the total draft resistance of the plough body. The main ways of lowering the friction resistance and the total draft resistance of the plough are the introduction of a more rational design of its body having optimum parameters, decreasing the resistance of the share-mouldboard surface and the values of reactions of the supporting surfaces, as well as the application of antifriction materials and better modes of joining with tractors.

INTRODUCTION

It follows from our previous investigations (Vilde 1999, 2003, 2004; Rucins and Vilde 2003, 2004, 2006) that the draft resistance of ploughs depends on the body parameters, the physical and mechanical properties of soil and the working modes. However, there were not enough investigations for analytic assessment of the impact of the body parameters and the variability of soil properties, such as friction, on the variations in the ploughing resistance. This encumbers the calculation of the proper solution of the plough body design and raising the ploughing efficiency.

The purpose of these investigations was analytical assessment of the impact of the variability of the plough body parameters and the soil friction properties on the variations in the ploughing resistance in order to determine the optimal body design under particular soil conditions to improve the ploughing efficiency.

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 and the working modes. On the basis of previous investigations, a computer algorithm has been worked out 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).
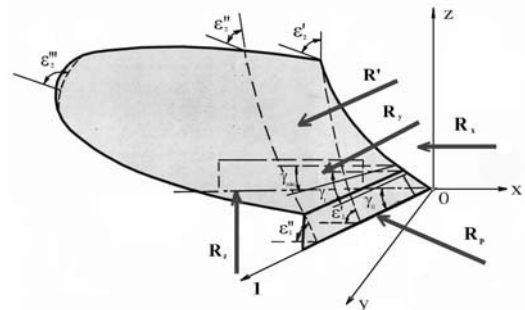
Figure 1. Scheme of the plough body, its parameters and acting forces.

According to our previous investigations (Vilde 1999) the draft resistance R_x of the plough body is determined by the share of the cutting resistance R_{Px}, the resistance caused by the gravity (weight) R_{Gx} of the soil slice 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).

$$R_x = \sum R_{ix} = R_{Px} + R_{Gx} + R_{Jx} + R_{Ax} + R_{Qx} \qquad (1)$$

The vertical reaction R_z and the lateral reaction R_y of the operating part are defined by corresponding partial reactions:

$$R_z = \sum R_{iz}; \qquad R_y = \sum R_{iy} \qquad (2; 3)$$

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:

$$R_x = R'_x + R''_x = \sum R'_{ix} + f_0\left(\sum R_{iz} + \sum R_{iy} + p_{Axy}S_{xy} + p_{Axz}S_{xz}\right) \qquad (4)$$

where: f_0 is the coefficient of the soil friction along the working and supporting surfaces of the plough body;
p_{Axy} and p_{Axz} – the specific adhesion force applied, respectively, to the lower and the lateral supporting surfaces of the body;

S_{xy} and S_{xz} − the surface area, respectively, of the lower and the lateral supporting surfaces of the body.

The friction resistance F_x is a constituent part of these reactions and their components (Rucins et al. 2003), and, by analogy, we can write that

$$F'_x = \sum F'_{ix} = F'_{Px} + F'_{Gx} + F'_{Jx} + F'_{Ax} + F'_{Qx} = R'_x - R'_{xo} \tag{5}$$

$$F''_x = f_0(R_z + R_y + p_{Axy} S_{xy} + p_{Axz} S_{xz}) = R''_x \tag{6}$$

$$F_x = F'_x + F''_x \tag{7}$$

The friction resistance of the share-mouldboard surface is defined as the difference between the total resistance (general value of the partial resistance) and resistance R_{xo} in operation without friction ($f_0 = 0$).

$$F_{ix} = R_{ix} - R_{ixo}; \quad F_x = R_x - R_{xo} \tag{8; 9}$$

Ratio λ_F of the friction resistance in the partial and total resistance (reaction) is determined from their correlations:

$$\lambda_{F_{ix}} = F_{ix} R_{ix}^{-1}, \quad \lambda_{F_x} = F_x R_x^{-1} \tag{10; 11}$$

Ratio λ_R of the supporting reactions in the partial and total draft resistance is determined from correlation:

$$\lambda_{R_i} = R_i R_{ix}^{-1} \tag{12}$$

The cutting resistance R'_{Px} is proportional to soil hardness ρ_0 and the share edge surface area ω:

$$R'_{Px} = k_p \rho_0 \omega = k_p \rho_0 ib, \tag{13}$$

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 friction of soil along the edge does not influence the cutting resistance of the edge.
At a sharp ploughshare (the rear bevel is absent)

$$R_{Pz} = 0 \tag{14}$$

At a blunt (threadbare) ploughshare having a rear bevel the vertical reaction R_{Pz} on the hard soils can reach a summary value of vertical reactions, that arise from other forces acting on the share-mouldboard surface (soil gravity and inertia) and the weight Q of the body.
At an inclined ploughshare a lateral reaction R_{Py} arises, its value being affected by the friction reaction.

$$R_{Py} = k_p \rho_0 ib \, ctg \, (\gamma_0 + \varphi_0) \tag{15}$$

where: γ_0 is the inclination angle of the edge towards the direction of the movement (the wall of the furrow);
φ_0 − the angle of friction.

When friction is absent, $f_0 = 0$, $\varphi_0 = 0$ and

$$R_{Pyo} = k_p \rho_0 ib \, ctg \, \gamma_0 \tag{16}$$

The 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

$$R''_{Px} = k_p \rho_0 ib f_0 \, ctg \, (\gamma_0 + \varphi_0) = F''_{Px} \tag{17}$$

The total cutting resistance is:

$$R_{Px} = k_p \rho_0 ib \, [1 + f_0 \, ctg \, (\gamma_0 + \varphi_0)] \tag{18}$$

The lateral cutting resistance of the knife is determined by formulae, similar to those for the cutting resistance of the share. 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 gravity of the lifting soil slice:

$$R'_{Gx} \approx q \, \delta g k_y r \sin^{-1} \gamma \, *$$
$$* \{ [(\sin \gamma \cos \varepsilon_1 + \cos^2 \gamma \sin^{-1} \gamma) e^{f_0 \sin \gamma (\varepsilon_2 - \varepsilon_1)} -$$
$$- (\sin \gamma \cos \varepsilon_2 + \cos^2 \gamma \sin^{-1} \gamma)] \cos \varepsilon_1 +$$
$$+ (\cos \varepsilon_1 e^{f_0 \sin \gamma (\varepsilon_2 - \varepsilon_1)} - \cos \varepsilon_2)(\cos \varepsilon_1 -$$
$$- f_0 \sin \varepsilon_1 \sin \gamma)^{-1} \sin \varepsilon_1 \, *$$
$$* \left[\sin \varepsilon_1 \sin \gamma + f_0 (\sin^2 \gamma \cos \varepsilon_1 + \cos^2 \gamma) \right] \} \tag{19}$$

$$R_{G_y} \approx q \delta g \, r \sin^{-1} \gamma (\varepsilon_2 - \varepsilon_1)(\varepsilon_1 + 0.52) \, ctg \gamma \tag{20}$$

$$R_{G_z} \approx q \delta g \, r \sin^{-1} \gamma \, (\varepsilon_2 - \varepsilon_1) \tag{21}$$

$$R''_{Gx} = f_0 (R_{Gz} + R_{Gy}) = F''_{Gx} \tag{22}$$

Forces caused by the soil inertia:

$$R'_{Jx} = q \, \delta v^2 k_y^{-1} \sin \gamma \{ (\sin \gamma \cos \varepsilon_1 + \cos^2 \gamma \sin^{-1} \gamma) *$$
$$* e^{f_0 \sin \gamma (\varepsilon_1 - \varepsilon_2)} - (\sin \gamma \cos \varepsilon_2 + \cos^2 \gamma \sin^{-1} \gamma) +$$
$$+ (\cos \varepsilon_1 - f_0 \sin \varepsilon_1 \sin \gamma)^{-1} e^{f_0 \sin \gamma (\varepsilon_2 - \varepsilon_1)} *$$
$$* \sin \varepsilon_1 \left[\sin \varepsilon_1 \sin \gamma + f_0 (\sin^2 \gamma \cos \varepsilon_1 + \cos^2 \gamma) \right] \} \tag{23}$$

$$R_{Jz} = q \, \delta \, v^2 k_y^{-1} \sin \gamma \sin \varepsilon_2 \, e^{f_0 \sin \gamma (\varepsilon_2 - \varepsilon_1)} \tag{24}$$

$$R_{Jy} \approx q \, \delta \, v^2 k_y^{-1} \sin \gamma \cos \gamma (1 - \cos \varepsilon_2) \tag{25}$$

$$R''_{Jx} = f_0 (R_{Jz} + R_{Jy}) = F''_{Jx} \tag{26}$$

Forces caused by soil adhesion:

$$R'_{Ax} = p_A br \sin^{-1} \gamma (e^{f_0 \sin \gamma (\varepsilon_2 - \varepsilon_1)} - 1) *$$
$$* \{ \sin \gamma \cos \varepsilon_1 + \cos^2 \gamma \sin^{-1} \gamma + (\cos \varepsilon_1 - f_0 \sin \varepsilon_1 \sin \gamma)^{-1} *$$
$$* \sin \varepsilon_1 \left[\sin \varepsilon_1 \sin \gamma + f_0 (\sin^2 \gamma \cos \varepsilon_1 + \cos^2 \gamma) \right] \} \tag{27}$$

$$R_{Az} = 0; \qquad R_{Ay} \approx 0; \tag{28), (29}$$

$$R''_{Ax} = f_0 (p_{Axy} S_{xy} + p_{Axz} S_{xz}) = F''_{Ax}. \tag{30}$$

where: q − the cross section area of the soil slice;
δ – 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;
b – the surface width of the soil slice;
ε_1 and ε_2 are correspondingly the initial and the final angles of the lifting (share- mouldboard) surface;
g – acceleration caused by gravity ($g = 9.81$).

The draft resistance caused by the body's weight:

$$R''_{Qx} = Q f_0 \qquad (31)$$

where: Q – the weight of the plough body (including a part of the weight of the plough).

The soil friction coefficient and the specific force of soil adhesion are not constant values. Their values decrease with the increase in speed (Rucins at al., 2003). 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 AND DISCUSSION

The presented work discusses, as an example, the simulation results of the impact of the body parameters, the physical and mechanical properties of soil, such as the soil friction and the working modes on the draft resistance of the plough body at various initial lifting angles ε_1, and at various angles γ of the horizontal generatrices, and at various working widths depending on the speed of operation when ploughing, for example, loamy soils, which predominate in Latvia.

Calculations were carried out with a computer according to the foregoing formulae.

The following values of the basic factors were taken into consideration affecting the resistance of the share–mouldboard surface and the plough body.

The parameters of the plough body:

The thickness of the share blade and knife	$i = 0.004$ m
The initial angle of the lifted strip of soil	$\varepsilon_1 = 30^0$
The final angle of the lifted strip of soil	$\varepsilon_2 = 100^0$
The inclination angle of the horizontal generatrix	$\gamma = 35^0\ldots45^0$
The radius of the curvature of the lifting surface	$r = 0.5$ m
The area of the lower supporting surface	$S_{xy} = 0.0157\,\text{m}^2$
The area of the lateral supporting surface	$S_{xz} = 0.068$ m^2
The weight of the plough body	$Q = 200$ kg

The physical and mechanical properties of soil:

The hardness of soil	$\rho = 4.1$ mPa
The density of soil	$\delta = 1600$ kg m^{-3}
The coefficient of soil friction	$f_0 = 0\ldots0.6$
The adhesion force	$p_{A0} = 2.5$ kPa

The mode and status of work:

The ploughing depth	$a = 0.20$ m
The working width of the body	$b = 0.35$ m
The cross section area of the soil slice	$q = 0.07$ m^2
The soil compaction coefficient in front of the share-mouldboard surface	$k_y = 1.1$
The working speed	$v = 1\ldots5$m s^{-1}

The inclination angle γ of the horizontal generatrix of the real share-mouldboard surfaces of the plough bodies lies between $26^0\ldots50^0$. Steeper surfaces $(\gamma > 50^0)$ refer to the slanting blades of bulldozers.

As an example, the calculation results of the impact of the soil friction coefficient f_0 upon the draft resistance of the plough body share-mouldboard (lifting) surface, as well as reacting forces on the supporting surfaces, the draft resistance and the total draft resistance of the entire plough body at the inclination angle $\varepsilon_1 = 30°$ of the share (initial soil slice lifting angle), at the inclination angle $\gamma = 35°\ldots45°$ of the horizontal generatrix and at various speeds v are presented in the following graphs.

The draft resistance of the lifting (share-mouldboard) surface and its components are presented in Figs. 2 – 5, the reacting forces on the supporting surfaces – in Figs. 6 – 10, the draft resistances of the supporting surfaces – in Fig. 11, and the total draft resistance of the plough body – in Fig. 12.
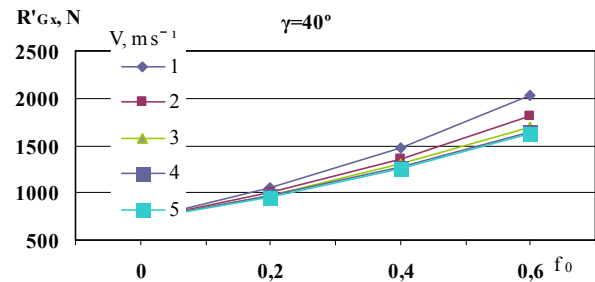
Fig.2. Impact of the soil friction coefficient f_0 upon the draft resistance of the plough body share–mouldboard surface caused by the gravity of the soil slice.
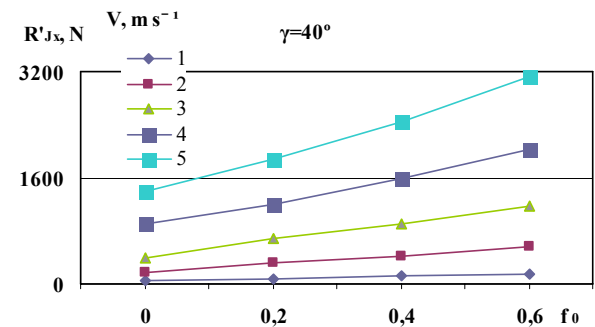
Fig.3. Impact of the soil friction coefficient f_0 upon the draft resistance of the plough body share–mouldboard surface caused by the soil inertia forces.
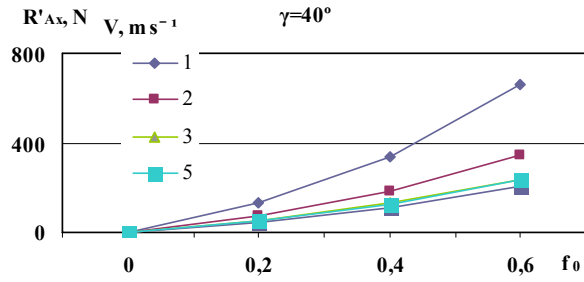
Fig.4. Impact of the soil friction coefficient f_0 upon the draft resistance of the plough body share–mouldboard surface caused by adhesion.

a

b

c
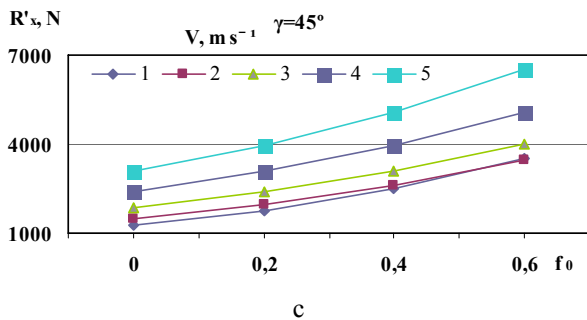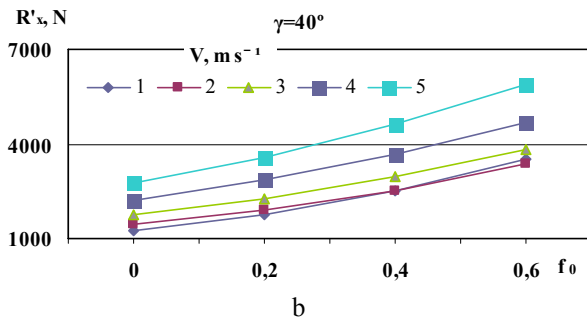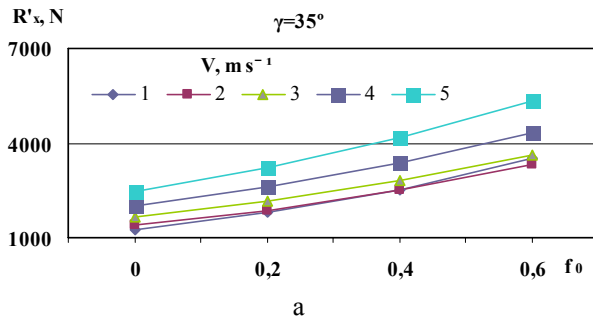
Fig.5. Impact of the soil friction coefficient f_0 upon the total draft resistance of the plough body share–mouldboard surface caused by the soil gravity, the inertia forces, adhesion and soil cutting resistance at the inclination angle γ of the horizontal generatrix: a – γ =35°; b – γ = 40°; c – γ = 45°

From the graphs above (Figs, 2–5) it follows that at the soil friction coefficient f_0 = 0.3...0.4 and at the speed v=2…3 m s^{-1}, presently predominating in ploughing, the draft resistance caused by the soil friction takes 36…42% of the total draft resistance of the share–mouldboard surface.
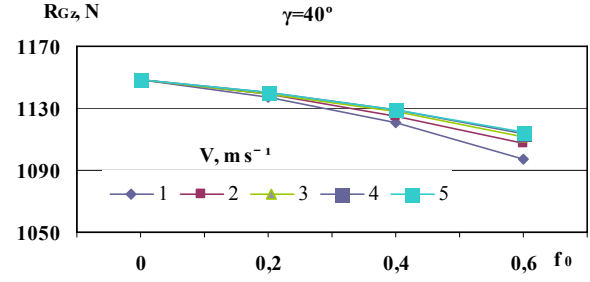
Fig.6. Impact of the soil friction coefficient f_0 upon the vertical reaction of the plough body caused by the gravity of the soil slice.
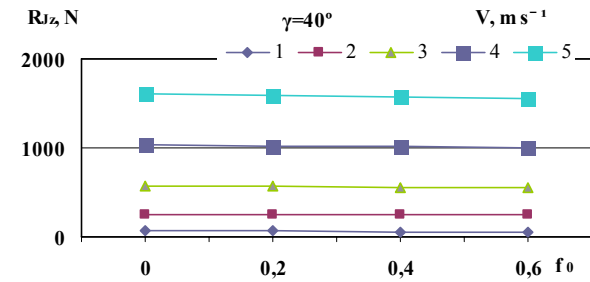
Fig.7. Impact of the soil friction coefficient f_0 upon the vertical reaction of the plough body caused by the inertia forces of the soil slice.
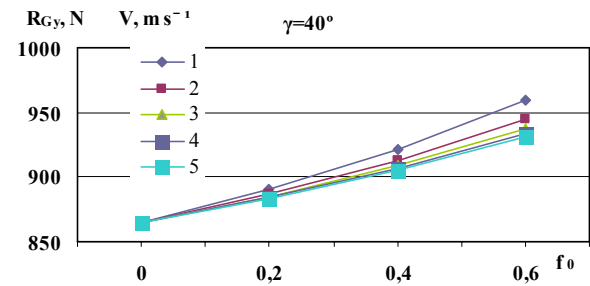
Fig.8. Impact of the soil friction coefficient f_0 upon the lateral reaction of the plough body caused by the gravity of the soil slice.
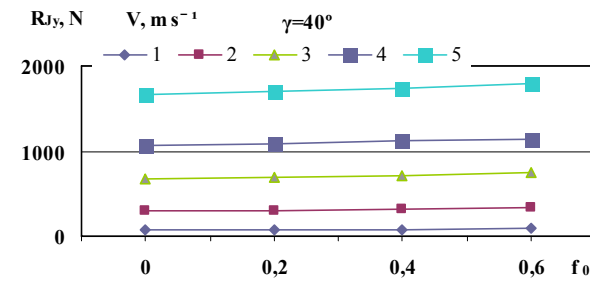
Fig.9. Impact of the soil friction coefficient f_0 upon the lateral reaction of the plough body caused by the inertia forces of the soil slice.

The calculations and graphs above (Figs. 6–9) show that the value of the soil friction caused at the soil slice gravity, inertia forces and adhesion has only a little influence on the reactions of the supporting surfaces.
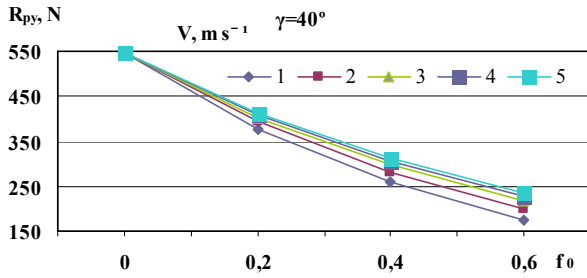
Fig.10. Impact of the soil friction coefficient f_0 upon the lateral reaction caused by the soil cutting with the plough share at the inclination angle of the cutting edge $\gamma_0 = 40°$.

The graph above (Fig. 10) shows that at the values of the friction coefficient $f_0 = 0.3...0.4$ the lateral reaction caused by the soil cutting decreases on 36…55%.

a

b

c
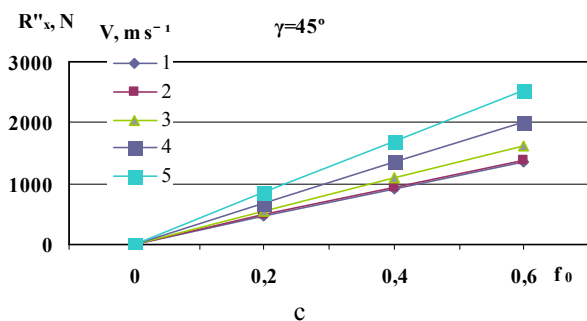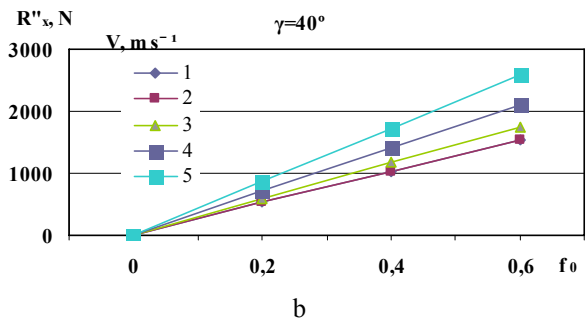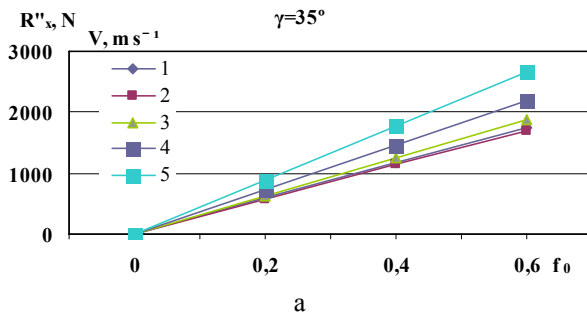
Fig.11. Impact of the soil friction coefficient f_0 upon the draft resistance of the supporting surfaces of the plough body: a $- \gamma =35°$; b $- \gamma = 40°$; c $- \gamma = 45°$.

It follows from the graphs above (Fig. 11) that the increase in speed increases the draft resistance of the supporting surfaces caused by soil friction. The value of the inclination angle of the horizontal genetratrix γ at the interval $\gamma = 35°…45°$ has only a little influence on the draft resistance of the supporting surfaces.
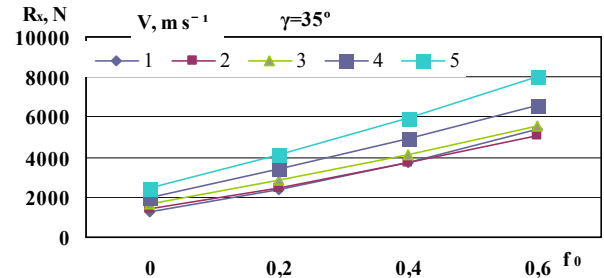
a
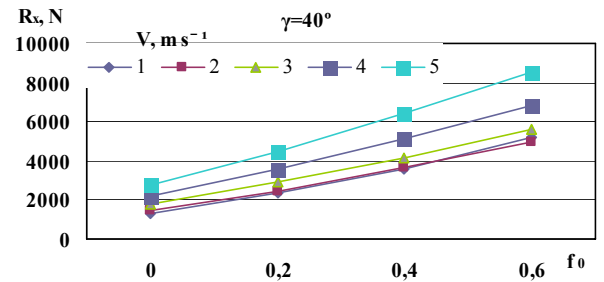
b
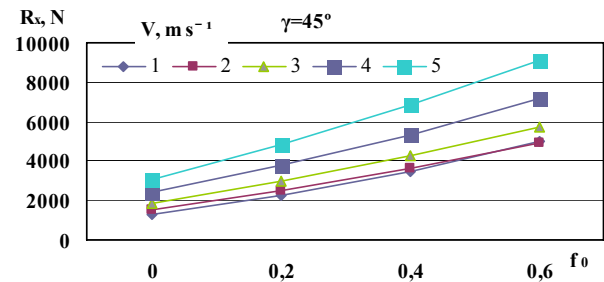
c

Fig.12. Impact of the soil friction coefficient f_0 upon the total draft resistance of the plough body:
a $- \gamma =35°$; b $- \gamma = 40°$; c $- \gamma = 45°$

It follows from the graphs above (Fig. 12) that at the values of the friction coefficient $f_0 = 0.3...0.4$ the draft resistance caused by the soil friction takes 46…62% of the total draft resistance of the plough body. It follows that the total draft resistance is approximately proportional to the friction coefficient. Increasing the speed decreases the share (ratio λ_F) of the friction resistance in the total draft resistance of the plough body. This phenomenon can be explained by the decreasing value of the friction coefficient when the speed is increasing (Vilde et al. 2007).

From the graphs (Fig. 12) it is evident too that at the values of the friction coefficient $f_0 = 0.3...0.4$ increasing the inclination angle of the horizontal genetratrix γ in the interval $\gamma = 35°…45°$ increases the draft resistance of the plough body to 6…10 %. This phenomenon is in agreement with the previous conclusions that the optimal values for the inclination angle of the horizontal genetratrix γ on the initial part of the share–mouldboard surface are 34…38[0] (Rucins and Vilde 2007).

It follows from formulas (17) – (30), too, that increasing the initial lifting angle ε_1 increases the draft resistance of the share-mouldboard surface, including the resistance of the soil friction (Rucins at al. 2007), but increasing the

working width of the body decreases the specific draft resistance of ploughing (Rucins and Vilde 2005). It was established from them that the optimal values of the initial lifting angle are $\varepsilon_1 = 28...32°$ and the optimal working width of the plough body – $b = 45...50$ cm.

From the presented example it is evident that the draft resistance of the supporting surfaces is considerable. It can reach 25…30% of the total plough body draft resistance, or 36…44% of its share–mouldboard draft resistance (Figs. 5, 11, 12).

Therefore it is very important for the reduction of the energy consumption of ploughing to reduce the draft resistance of the supporting surfaces. It may be obtained by using a contemporary hang–up device with the tractors, for example, power regulation allowing the transfer of the vertical reactions of the plough to the body of the tractor (Vilde et al. 2004). It may decrease the draft resistance of the ploughs to 6…10 %.

In the sources provided by other authors there are no materials about the application of the simulation methods in order to study the impact of the plough body parameters, as well as the soil friction properties on the draft resistance of the plough bodies. In order to obtain a better design of the plough body, a series of different bodies were built and tested (Larsen 1968, Burchenko et al. 1976, Burchenko 2001, Nikiforov and Ivanov 1973). Yet it is bound with a great loss of resources, labour and time, so the best solution of the compared variants may not always be the optimum ones.

The materials of our investigations carried out by using the correlations indicated above present the values and regularity of the changes in the forces, the soil friction, 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 the working conditions depending on the body parameters, the soil friction coefficient and the working speed. In such a way it is possible to discover the draft resistance structure of the body, to assess the ratio of each element in the total resistance, to search and find possibilities how to reduce the tillage energy requirement.

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, as well as the impact of the soil friction properties.

2. Presentation of the plough body draft resistance as the sum of its components – the cutting resistance of the soil slice, the resistance caused by its gravity, the soil inertia forces and adhesion including the soil friction resistance - allows analysing the forces acting upon the share-mouldboard and the supporting surfaces, finding out the character of their changes depending on the soil

properties, parameters of the surfaces, of working speed, assessment of their ratio in the total resistance and determination of the optimal parameters of the body.

3. Increase in the inclination of the horizontal generatrix leads to a decrease in the draft resistance caused by the weight, adhesion and friction of the soil but it increases the resistance caused by the 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 soil slice.

4. The impact of the soil–metal friction upon the draft resistance of the plough body is significant. It may reach 46…62 % of the total draft resistance including the resistances of the supporting surfaces (25…30%). Therefore measures will be taken to diminish it, for example, by improving the body design, improving the mode of aggregation (joining) with tractors, using antifriction materials. The relief of the lower supporting surface may diminish the draft resistance of the body caused from the soil friction to 6…10 %.

5. The optimal values of the main parameters of the body for contemporary ploughs, working at the speeds of 2…2.5 m s^{-1} are: the inclination angle of the share towards the furrow bottom – $28...32^0$; the inclination angle of the horizontal generatrix towards the furrow wall on the initial part of the share-mouldboard surface – $34...38^0$, on the top – not less than 48^0; the working width of the bottom – 45…50 cm.

6. The use of bodies having optimal parameters allows obtaining a good ploughing quality, reduction of the draft resistance by 12…20% and a corresponding rise in the efficiency, saving fuel and financial means.

7. Further, in such a way it is possible to carry out the simulation of the impact of the soil moisture, as well as of the other soil properties, on the work and the draft resistance of the ploughs with aim to find out the favourable soil conditions for efficient ploughing.

REFERENCES

Vilde A. 1999. "Dynamics of the soil tillage machine operating parts and their elements". *Proceedings of the Latvia University of Agriculture, Vol.1 (295).* Jelgava, Latvia, 36–44.

Vilde A. 2004. "Mechanical and mathematical foundations for modelling the dynamics of soil tillage machine operating parts". *TEKA Commission of Motorization and Power Industry in Agriculture, Volume IV.* Polish Academy of Sciences Branch in Lublin. Lublin, Poland, 228–236.

Rucins, A. and A. Vilde. 2004. "Mathematical modelling of the operation of plough bodies to determine their draft resistance and optimum parameters". *TEKA Commission of Motorization and Power Industry in Agriculture, Volume IV.* Polish Academy of Sciences Branch in Lublin. Lublin, Poland, 177–184.

Rucins, A.; A. Vilde and W. Tanas'. 2006. "Forces acting on a plough body". *TEKA Commision of Motorization and Power Industry in Agriculture, Volume VI.* Polish Academy of Sciences Branch in Lublin. Lublin, Poland, 135–145.

Rucins, A. and A. Vilde. 2005. "Modelling forces acting on the plough body". *Simulation in Wider Europe. 19th*

European Conference on modelling and Simulation ECMS 2005 June 1-4, 2005 Riga, Latvia. Riga, 414–419.

Rucins, A.; A. Vilde and J. Nowak. 2007. "Impact of the share inclination angle on the ploughing resistance". *TEKA Commission of Motorization and Power Industry in Agriculture, Volume VII.* Polish Academy of Sciences Branch in Lublin, Lublin, Poland, 199 – 209.

Rucins, A. and A. Vilde. 2005. "Plough body working width optimization". *Recent results and future challenges in soil tillage research. International scientific seminar Reports.* Akademija, Lithuania, 69–75.

Vilde A. 2003. "The impact of soil moisture and composition on its properties and energy consumption of tillage". *TEKA Commission of Motorization and Power Industry in Agriculture, Volume III.* Polish Academy of Sciences Branch in Lublin. Lublin, Poland, 249 – 255.

Vilde A. 2001. "Physical and mechanical properties of soil affecting energy capacity of its tillage". *Proceedings of the 1st 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., and A. Vilde. 2003. "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.

Vilde, A.; S. Cesnieks and A. Rucins. 2004. "Minimisation of soil tillage". *TEKA Commission of Motorization and Power Industry in Agriculture, Volume IV.* Polish Academy of Sciences Branch in Lublin. Lublin, Poland, 237–242.

Rucins, A. and A. Vilde. 2007. "Impact of the Plough Body Parameters on the Ploughing Efficiency". *Ecology and Agricultural Machinery. Proceedings of the 5th International Scientific and Practical Conference, May, 15–16, 2007, Vol. 2.* Saint Petersburg–Pavlovsk, 2007, 282–290.

Vilde, A.; A. Rucins and G. Sevostjanovs. 2007. "Impact of Speed on the Soil Sliding Resistance". *International Conference "Technical and Technological Progress in Agriculture", No 12, 20–21 September 2007.* Raudondvaris, Lithuania, 34–38.

Larsen, L.W.; W.G. Lovely and C.W. Bockup. 1968. "Predicting draught forces using model mouldboard ploughs in agricultural soils". *Trans. American Society of Agricultural Engineers*, 11, 665–668.

Burchenko, P.N.; A.N. Ivanov; B.A. Kashajev; V.G. Kirjuhin and A.I Miltsev. 1976. "Resultaty issledovanija rabochih organov skorostnyh plugov." In *Povyshenie rabochih skorostjei mashinno-traktornyh agregatov.* Moscow, Kolos, 215–218. (In Russian). "Results of the investigation of the working parts of high-speed ploughs".

Burchenko, P.N. 2001. "K teoryi razvertivajusheisja lemeshno-otvaljnoi poverhnosti korpusa pluga". In *mashinnye tehnologii i tehnika dlja proizvodstva zernovih, maslichnih i zernobobovih kultur. Tom 3, chastj 1.* Moscow, VIM, 38-51. (In Russian). "Belonging to the theory of the roll out share-mouldboard surface of the plough body".

Nikiforov, P.E. and A.N. Ivanov. 1973. "Issledovanije rabochih organov plugov dlja raboty so skorostjami 10–15 km/h". In *Povyshenie rabochih ckorostjei mashinno–traktornyh agregatov.* Moscow, CINTIAM, 197–203. (In Russian). "Investigation of the working parts of ploughs for working at the speeds of 10–15 km/h".

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 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 Achievements 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 a Latvian State Emeritus Scientist 2001. He is an expert of promotion councils and a publicist who has written more than 800 publications including eighteen monographs. He enjoys orchards and stenography.

His e–mail address is vilde@delfi.lv

ADOLFS A. RUCINS was born in 1962 in Riga, Latvia. After finishing the Latvian Industrial Polytechnical School he entered the Latvia Technical University, where he studied Machinery and obtained his engineer degree in 1989. Subsequently he obtained the Degree of a Doctor sc. ing. in 2008. He worked for 3 years as a lecturer at the Latvian Technical University. Since 1992, up to this time he is a researcher at the Latvian Research Institute of Agricultural Machinery.

The research branches: technology, machines and units for soil tillage and grain sowing. Now he takes part in the research works in Precision Agriculture using the GPS, as well as simulation of the forces acting on the plough body, in order to determine its draft resistance and optimal parameters. He has also written 67 publications.

His e–mail address is arucins@delfi.lv

MODELLING A PEROXIDASE-BASED FLUORESCENT BIOSENSOR

Evelina Gaidamauskaitė
Department of Software Engineering
Vilnius University
Naugarduko str. 24, LT-03225, Vilnius, Lithuania
E-mail: evelina.gaidamauskaite@mif.vu.lt

Romas Baronas
Institute of Mathematics and Informatics
Vilnius University
Akademijos str. 4, LT-08663, Vilnius, Lithuania
E-mail: romas.baronas@mif.vu.lt

KEYWORDS

Modelling, reaction-diffusion, simulation, fluorescent biosensor.

ABSTRACT

The paper presents a one-dimensional-in-space mathematical model of a peroxidase-based fluorescent biosensor. The mathematical model is based on a system of non-linear reaction diffusion equations. The problem was solved numerically using finite difference technique. The model was used to study the effect of biosensor parameters, namely the thickness of the enzyme layer and the outer substrate concentration, on the response of the biosensor and it's sensitivity. The performed calculations showed a complex influence of the substrate and hydrogen peroxide concentrations.

INTRODUCTION

Biosensors are analytical devices converting a biochemical recognition reaction into a measurable effect (Scheller and Schubert, 1992; Turner et al., 1987). They are applied widely to monitor chemical substances in the medicine, food technology and the environmental industry (Wollenberger et al., 1997). Optical biosensors based on the fluorescence detection generate a photoluminescent signal indicative of target analyte binding (Ligler and Taitt, 2002). Fluorescence emission is then detected and converted into an analytical signal (Knopf and Bassi, 2007). They have been used for the analysis of many chemical and biological substances, especially in cases limited by low analyte concentrations (Atwood and Steed, 2004).

The mathematical model of optical biosensors has been very recently developed (Baronas et al., 2007a). However, the model was based upon absorbance measurements. In this report, a mathematical model for fluorescent biosensors is described. The model is used to predict the response of biosensor with varying physical and chemical characteristics.

MATHEMATICAL MODEL

The biochemical reaction of substrate (S) with enzyme peroxidase (E) in the presence of hydrogen peroxide (H_2O_2) to form the product (P) may be expressed as

$$E + H_2O_2 \xrightarrow{k_1} I + H_2O, \qquad (1)$$

$$I + S \xrightarrow{k_2} E + P, \qquad (2)$$
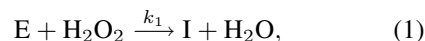
where I and H_2O are the compound and water, respectively, k_1 and k_2 are the reaction rates.

The product (P) emits light and therefore the response of the biosensor increases during the reaction as the product forms. The fluorescence of the product (P) is proportional to the concentration of the analyte (S) being measured. (Vo-Dinh, 2003).

Assuming the symmetrical geometry of the biosensor and homogeneous distribution of immobilized enzyme in the membrane, the mass transport and the reaction kinetics in the enzyme layer for the reaction scheme (1), (2) can be described by the following system of the reaction-diffusion equations ($0 < x < d$, $t > 0$):

$$\frac{\partial S}{\partial t} = D_S \frac{\partial^2 S}{\partial x^2} - k_2 CS, \qquad (3)$$

$$\frac{\partial P}{\partial t} = D_P \frac{\partial^2 P}{\partial x^2} + k_2 CS, \qquad (4)$$

$$\frac{\partial H}{\partial t} = D_H \frac{\partial^2 H}{\partial x^2} - k_1 EH, \qquad (5)$$

$$\frac{\partial E}{\partial t} = -k_1 EH + k_2 CS, \qquad (6)$$

$$\frac{\partial C}{\partial t} = k_1 EH - k_2 CS, \qquad (7)$$

where x and t stand for space and time, $S(x,t)$, $P(x,t)$, $H(x,t)$, $E(x,t)$, $C(x,t)$ are the substrate, product, hydrogen peroxide, enzyme and compound concentrations in the enzyme layer, d is the thickness of the enzyme layer, D_S, D_P, D_H are the diffusion coefficients. The enzyme and the formed compound are immobilized in the membrane and therefore there are no diffusion terms in the equations for the enzyme and compound.

Let $x = 0$ represents the plate surface, while $x = d$ the boundary between the enzyme membrane and the bulk solution. The biosensor is initially immersed in a hydrogen peroxide solution. The operation of the biosensor starts when some substrate appears on the outer surface of the enzyme layer. This is used in the initial

conditions ($t = 0$),

$$P(x, 0) = C(x, 0) = 0, \quad 0 \leq x \leq d,$$
$$H(x, 0) = H_0, \quad 0 \leq x \leq d,$$
$$E(x, 0) = E_0, \quad 0 \leq x \leq d, \quad \quad (8)$$
$$S(x, 0) = 0, \quad 0 \leq x < d,$$
$$S(d, 0) = S_0,$$

where E_0 stands for the initial concentration of the enzyme in the enzyme membrane, H_0 is the hydrogen peroxide concentration in the bulk solution as well as the initial concentration in the enzyme membrane and S_0 is the substrate concentration in the bulk solution.

If the bulk solution is well-stirred and in powerful motion, then the diffusion layer ($0 < x < d$) remains at a constant thickness. In the bulk solution the concentration of the substrate, product and hydrogen peroxide remains constant ($t > 0$),

$$S(d, t) = S_0, \; P(d, t) = 0, \; H(d, t) = H_0. \quad (9)$$

Assuming the impenetrable and unreactive plate surface, the mass flux of the species must vanish at this boundary,

$$\frac{\partial S}{\partial x}\Big|_{x=0} = \frac{\partial P}{\partial x}\Big|_{x=0} = \frac{\partial H}{\partial x}\Big|_{x=0} = 0. \quad (10)$$

By an extension of the Beer-Lambert law, the response of a fluorescent biosensor describing the intensity of fluorescence $F_I(t)$ at time t may be obtained as follows (Baldini et al., 2006; Knopf and Bassi, 2007):

$$F_I(t) = I_0 \varphi \left(1 - 10^{-\varepsilon_P l_{ef} \bar{P}}\right), \quad l_{ef} = d, \quad (11)$$

where I_0 stands for the intensity of excitation light, φ is a quantum yield of fluorophore, ε_P is the molar extinction coefficient of the product, l_{ef} - effective thickness of the diffusion layer, $\bar{P}$ - the concentration of the product averaged through the entire diffusion layer. The quantum yield of the product fluorescence φ, defined as the ratio of the number of photons emitted to the number of photons absorbed, practically varies between 0.001 and 1. For organic compounds ε_P varies between 10^4 and 10^2 m^{2}mol^{-1}.

The fluorescence $F_I(t)$ is almost linearly proportional to the averaged concentration of the product $\bar{P}$,

$$F_I(t) \approx 2.303 I_0 \varphi \varepsilon_P l_{ef} \bar{P}, \quad (12)$$

when the absorbance expressed as $\varepsilon_P l_{ef} \bar{P}$ is less than 0.1 (Vo-Dinh, 2003). Since the fluorescence intensity $F_I(t)$ is directly proportional to the intensity I_0 of the excitation light, the measured fluorescence $F(t)$ is usually reported in relative fluorescent units (RFU),

$$F(t) = F_I(t)/I_0. \quad (13)$$

We assume that the system (3)-(10) approaches a steady state as $t \to \infty$,

$$F_\infty = \lim_{t \to \infty} F(t), \quad (14)$$

where F_∞ is the steady state fluorescence.

The high sensitivity of the biosensor is especially important for the rapid detection of the analyte (Scheller and Schubert, 1992; Turner et al., 1987). The sensitivity is defined as a gradient of the steady state fluorescence with respect to the substrate concentration. The fluorescence varies in orders of magnitude with the concentration of the substrate to be analyzed. Therefore we consider dimensionless expression of the sensitivity,

$$B_S(S_0) = \frac{S_0}{F_\infty(S_0)} \times \frac{dF_\infty(S_0)}{dS_0}, \quad (15)$$

where B_S stands for the dimensionless sensitivity of the biosensor, $F_\infty(S_0)$ is the steady state fluorescence calculated at the substrate concentration S_0 in the bulk solution.

DIGITAL SIMULATION

No analytical solution for the problem is known, therefore a numerical approximation method must be used. We applied the implicit finite difference technique to solve the boundary value problem (3)-(10) (Britz, 2005; Samarskii, 2001). A uniform discrete grid was constructed in both directions x and t. The resulting systems of linear algebraic equations were solved efficiently because of the tridiagonality of their matrices. The computational model was implemented in Java programming language (Moreira et al., 2000).

We define the steady state fluorescence F_R of the biosensor as the fluorescence intensity calculated at the response time T_R,

$$F_R = F(T_R) \approx F_\infty,$$
$$T_R = \min_{j > 0, \, F_j > 0} \left\{ \tau_j : \frac{F_j - F_{j-1}}{F_j \tau} < \varepsilon \right\}, \quad (16)$$

where τ is the size of time step. We used $\varepsilon = 10^{-5}$ for the calculations.

Unless otherwise stated, we used the following values of the parameters:

$$D_S = D_P = D_H = 300 \; \mu\text{m}^2\text{s}^{-1},$$
$$k_1 = 7.1 \times 10^6 \; \text{s}^{-1}\text{M}^{-1},$$
$$k_2 = 2 \times 10^7 \; \text{s}^{-1}\text{M}^{-1}, \quad (17)$$
$$\varphi = 0.01, \quad d = 1 \; \mu\text{m},$$
$$\varepsilon_P = 10^3 \; \text{m}^2\text{mol}^{-1}.$$

The dynamics of the biosensor response having four different membrane thicknesses d is presented in Fig. 1. The steady state fluorescence appears with some delay corresponding to different thicknesses of the enzyme layers. The influence of the enzyme membrane thickness on the biosensor response was explored in greater detail (Baronas et al., 2003).
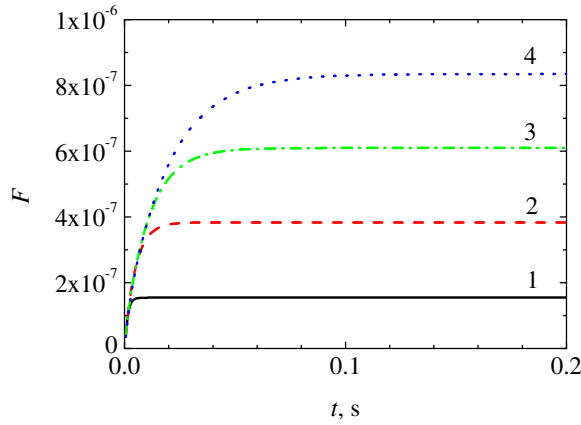
Figure 1: The dynamics of the fluorescence F at four enzyme layer thicknesses d: 1 (1), 2 (2), 3 (3), 4 (4) μm, $S_0 = 100$ μM, $E_0 = 1$ nM, $H_0 = 1$ mM

Figure 2: Dependence of the dimensionless steady state fluorescence F_R on the thickness d of the enzyme membrane at three substrate concentrations S_0: 10 (3), 100 (1, 4, 5, 6, 7), 1000 (2) μM, three enzyme concentrations E_0: 0.1 (5), 1 (1, 2, 3, 6, 7), 10 (4) nM and three hydrogen peroxide concentrations H_0: 0.1 (7), 1 (1, 2, 3, 4, 5), 10 (6) mM

RESULTS AND DISCUSSION

Distributions of the concentrations of the substrate, product, hydrogen peroxide, complex and enzyme peroxidase in the enzyme membrane for an optical biosensor were given in a previous paper (Baronas et al., 2007a). By changing the input parameters we analyze the behavior of the response and sensitivity of the fluorescent biosensor.

The Effect of the Thickness of the Enzyme Layer

The effect of the thickness of the enzyme layer on the response of amperometric as well as potentiometric biosensors has been studied recently (Baronas et al., 2003, 2007b; Scheller and Schubert, 1992; Turner et al., 1987). Fig. 2 shows the dependence of the dimensionless steady state fluorescence F_R on the thickness of enzyme membrane.

In the case of fluorescent biosensors, thickening the enzyme membrane causes the increase of the product concentration and therefore light is emitted to a greater extent. The response of the biosensor mostly depends on the outer concentration S_0 of the substrate (see curves 2 and 3 in Fig. 2). When the enzyme layer is relatively thick ($d > 1\mu$m), the used variations of enzyme and hydrogen peroxide have similar effect on the fluorescence (see curves 1, 4, 5, 6 and 7). This can be explained by production of the same amount of the compound I, limited by the equation (1).

The effect of the enzyme layer thickness on the biosensor dimensionless sensitivity B_S is shown in Fig. 3. The sensitivity of the biosensor increases extending the enzyme layer. With exception of curves 2 and 7, the high values of sensitivity ($B_S > 0.8$) are observed within specified interval. At higher outer substrate concentration S_0 the enzyme becomes saturated and cannot respond effectively to the change of the substrate concentration (curve 2). The almost similar results were obtained at lower
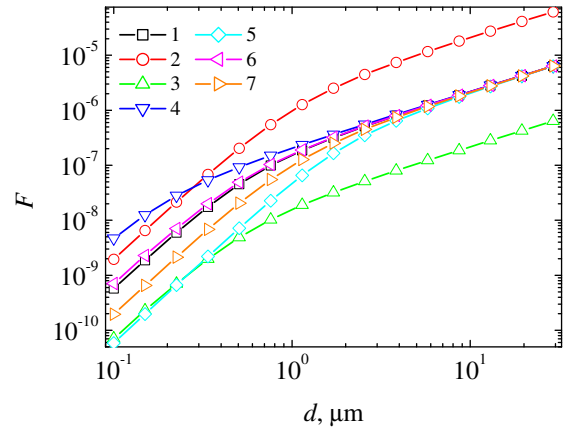
concentration of hydrogen peroxide H_0 (curve 7). The sensitivity is significantly reduced because of the lower concentration of the formed compound I in comparison with the concentration of the substrate. In both cases, the sensitivity B_S of the biosensor can be greatly enhanced by increasing the thickness of the enzyme layer.
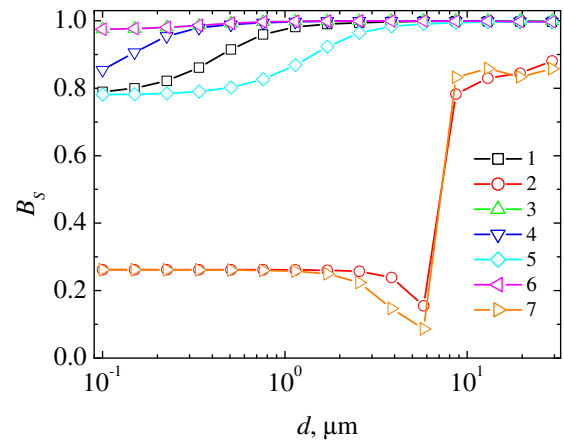
Figure 3: Dependence of the dimensionless sensitivity B_S on the thickness d of the enzyme membrane. The parameters and notation are the same as in Fig. 2

The Effect of the Outer Substrate Concentration

The effect of outer substrate concentration S_0 on the dimensionless steady state fluorescence F_R is depicted in Fig. 4. The approximately linear curves of the fluorescence F_R increase asymptotically approaching the steady

states. A variation in the initial concentration E_0 of the enzyme affects the fluorescence, but has no effect on the limit of linearity of the calibration curve (see curves 2 and 3). However, the initial concentration H_0 of hydrogen peroxide have an effect upon the dynamic range of the calibration curves (curves 4 and 5). An increase of the concentration H_0 expands the limit of linearity and wise versa.
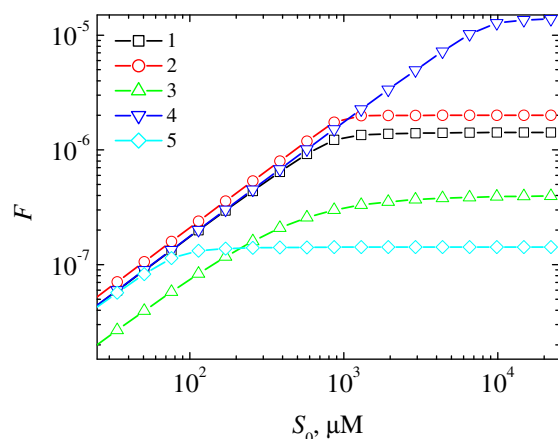
Figure 4: Dependence of the dimensionless steady state fluorescence F_R on the substrate concentration in the bulk solution S_0 at three concentrations E_0 of the enzyme: 0.1 (3), 1 (1, 4, 5), 10 (2) nM and three concentrations H_0 of the hydrogen peroxide: 0.1 (5), 1 (1, 2, 3), 10 (4) mM

The dimensionless sensitivity B_S is markedly reduced as the concentration of the substrate S_0 increases (see Fig. 5). The change of the initial enzyme concentration E_0 has slight effect on the biosensor sensitivity (curves 2 and 3). However, the lower concentration of enzyme (curve 3) causes an unacceptable sensitivity at $S_0 \approx > 1.1 \times 10^2 \mu M$. The high values of the sensitivity up to $S_0 \approx 10^4 \mu M$ can be observed with the increased initial concentration H_0 of hydrogen peroxide (curve 4).

CONCLUSIONS

The developed mathematical model (3)-(10) of a peroxidase-based fluorescent biosensor can be used for a digital simulation and an investigation of the biosensor response.

By providing a relatively thick enzyme layer a higher sensitivity and fluorescence values can be obtained. However, a great care must be taken in cases of high outer substrate and low initial hydrogen peroxide concentrations (Fig. 3).

The higher sensitivity of the fluorescent biosensor can be achieved by increasing the concentration of the hydrogen peroxide (Fig. 5).
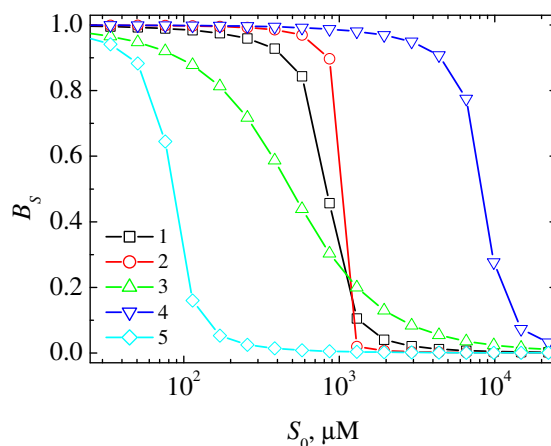
Figure 5: Dependence of the dimensionless sensitivity B_S on the substrate concentration S_0 in the bulk solution. The parameters and notation are the same as in Fig. 4

References

Atwood, J. L. and J. W. Steed. (2004). *Encyclopedia of Supramolecular Chemistry*. Marcel Dekker, New York.

Baldini, F.; A. N. Chester; J. Homola; and S. Martellucci. (2006). *Optical Chemical Sensors*. Springer, Amsterdam.

Baronas, R.; E. Gaidamauskaitė; and J. Kulys. (2007). "Modelling a Peroxidase-based Optical Biosensor". *Sensors*, 7(11):2723–2740.

Baronas, R.; F. Ivanauskas; and J. Kulys. (2003). "The influence of the enzyme membrane thickness on the response of amperometric biosensors". *Sensors*, 3(7):248–262.

Baronas, R.; F. Ivanauskas; and J. Kulys. (2007). "Computational modelling of the behaviour of potentiometric membrane biosensors". *Journal of Mathematical Chemistry*, 42(3):321–336.

Britz, D. (2005). *Digital Simulation in Electrochemistry*. Springer-Verlag, Berlin.

Knopf, G. K. and A. S. Bassi. (2007). *Smart Biosensor Technology*. CRC Press, New York.

Ligler, F. S. and C. R. Taitt. (2002). *Optical Biosensors: Present and Future*. Elsevier Science, Amsterdam.

Moreira, J. E.; S. P. Midkiff; M. Gupta; P. V. Artigas; M. Snir; and R. D. Lawrence. (2000). "Java programming for high performance numerical computing". *IBM Systems J.*, 39:21–56.

Samarskii, A. A. (2001). *The Theory of Difference Schemes*. Marcel Dekker, New York-Basel.

Scheller, F. and F. Schubert. (1992). *Biosensors*. Elsevier, Amsterdam.

Turner, A. P. F.; I. Karube; and G. S. Wilson. (1987). *Biosensors: Fundamentals and Applications*. Oxford University Press, Oxford.

Vo-Dinh, T. (2003). *Biomedical Photonics Handbook*. CRC Press, New York.

Wollenberger, U.; F. Lisdat; and F. W. Scheller. (1997). *Frontiers in Biosensorics 2: Practical Applications*. Birkhauser Verlag, Basel.

AUTHOR BIOGRAPHIES

EVELINA GAIDAMAUSKAITĖ was born in Vilnius, Lithuania. She is currently a PhD student at Vilnius University. She received her BSc and MSc degrees in Computer Science in 2004 and 2006, respectively from Vilnius University. Her research interests are in computational modelling of biosensors. Her e-mail is: `evelina.gaidamauskaite@mif.vu.lt`.

ROMAS BARONAS was born in 1959 in Kybartai, Lithuania. He is a professor and serves as chair of the Department of Software Engineering at Vilnius University. Prof. Baronas received his MSc degree in Applied Mathematics in 1982 and then obtained his PhD degree in Computer Science in 2000 from the Vilnius University. He also joined the Software Engineering Department at the Institute of Mathematics and Informatics. His teaching and research interests lie in the areas of database systems and computational modelling of biochemical processes. His e-mail address is: `romas.baronas@mif.vu.lt` and his personal Web-page can be found at `http://www.mif.vu.lt/~baronas`.

MODELLING AND SIMULATION OF FOOD NETWORK

Tibor Kmet
Department of Computer Science
Faculty of Natural Sciences
Constantine the Philosopher University
Tr. A. Hlinku 1, 949 74 Nitra
Slovak Republic
Email: tkmet@ukf.sk

KEYWORDS

Simple food web, partial functional differential equations, spatially constant equilibrium, stability analysis

ABSTRACT

The analysis of networks of ecological throphic transfers is a useful complement to simulation modelling in the quest for understanding whole-ecosystem dynamics. Throphic networks can be studied in quantitive and systematic fashion at several levels. The purpose of this paper is to investigate a system of parabolic equations with discrete time delays describing a simple throphic food network which consists of the $n+1$ type living organisms zooplankton, phytoplankton and microorganisms), the n type dissolved organic and inorganic nutrients and detritus. When the delays are relatively small, our predictions are also identical to the predictions given by the corresponding PDE. The system of parabolic equations is discretized by the finite difference method which yields a coupled system of nonlinear algebraic equations. Stability analysis of equilibria and some numerical examples are given.

INTRODUCTION

In ecosystem modelling, food network design is a central task. Ecological networks are quantitative, graph-based descriptions of ecosystems, consisting of compartments (trophospecies and nutrient pools) that exchange fluxes of nutrient or energy. One approach that has been used to account for the lack of empirically derived data is the development of simple algorithms to construct hypothetical, but ecologically realistic networks. However, there have been two distinct approaches marked by the initial assumptions one makes. The first group, based on population/community ecology, focuses strictly on who eats whom, producing structures involving primary producers, grazers, and predators, but explicitly lacks decomposers and detritus. As a result, these networks do not typically contain cycling. The second group, based on ecosystem ecology focuses on energy flowing the system and includes all functional groups including detritus (dead organic material) and decomposers (Fath at al., 2007). The interactions along the detritus feedback cycle are likely to occur at a different time scale than the conventional predation. This problem could be addressed by describing interactions with time-delayed differential equations (Halnes at al., 2007). The focus in most food web analyses has been on how they are arranged, affects different statistical and stability properties of model food webs. The dynamical theory of food webs has been based typically on local and global stability analysis, i.e. an existence and stability of equilibrum.
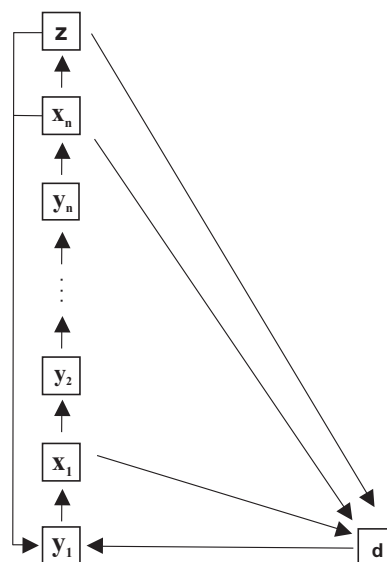

Figure 1. Diagram of the compartmental system modelled by (1): x_i, microorganisms; x_n, phytoplankton; y_i, nutrients; d, detritus; z, zooplankton.

The model that we treat has $2n + 2$ components consisting of the $n + 1$ type living organisms (zooplankton, phytoplankton and microorganisms), the n type dissolved organic and inorganic nutrients and detritus as shown in Figure 1. The nutrient y_i, which is a metabolic product of the microorganisms x_{i-1}, is assumed to be the one primarily responsible for limiting the x_i production for $i = 2, ..., n - 1$. The dissolved organic nutrient concentration y_1 is a result of the partial decomposition of the dead organisms. The phytoplankton x_n, which assimilates the metabolic product y_n, excretes the dissolved organic nutrient and limits zooplankton growth (z). Zooplankton excretes dissolved organic matter, too. The $n + 1$ living organisms levels, detritus and the n nutrients are modelled in terms of their nitrogen content, N. We may assume that the material stream from the component i to j by the Lotka-Volterra hypothesis varies as the product of the interacting components. This quadratic assumption will cause the resulting differential equations to be nonlinear. The dynamics of a simple food network is described by the following functional reaction-diffusion equations:

$$\frac{\partial d}{\partial t} = d_d \frac{\partial^2 d}{\partial p^2} + \sum_{i=1}^{n} x_i M_i(y_i) + \qquad (1)$$
$$z M_z(x_n) - K_1 d(t - r_{k1}, p)$$

$$\frac{\partial x_i}{\partial t} = d_{xi} \frac{\partial^2 x_i}{\partial p^2} + \quad x_i(t - r_{ui}, p) U_i(y_i(t - r_{ui}, p)) -$$
$$x_i(t - r_{ei}, p) E_i(y_i(t - r_{ei}, p)) -$$
$$x_i(t, p) M_i(y_i(t, p))$$

$$\frac{\partial x_n}{\partial t} = d_{xn} \frac{\partial^2 x_n}{\partial p^2} + \quad x_n(t - r_{un}, p) U_n(y_n(t - r_{un}, p) -$$
$$x_n(t - r_{en}, p) E_n(y_n(t - r_{en}, p)) -$$
$$x_n(t, p) M_n(y_n(t, p)) -$$
$$z(t - r_{uz}, p) U_z(x_n(t - r_{uz}, p))$$

$$\frac{\partial z}{\partial t} = d_z \frac{\partial^2 z}{\partial p^2} + \quad z(t - r_{uz}, p) U_z(x_n(t - r_{uz}, p)) -$$
$$z(t - r_{ez}, p) E_z(x_n(t - r_{ez}, p)) -$$
$$z(t, p) M_z(x_n(t, p))$$

$$\frac{\partial y_1}{\partial t} = d_{y1} \frac{\partial^2 y_1}{\partial p^2} + \quad z(t - r_{ez}, p) E_z(x_n(t - r_{ez}, p)) +$$
$$x_n(t - r_{en}, p) E_n(y_n(t - r_{en}, p)) +$$
$$K_1 d(t - r_{k1}, p) -$$
$$x_1(t - r_{u1}, p) U_1(y_1(t - r_{u1}, p))$$

$$\frac{\partial y_i}{\partial t} = d_{yi} \frac{\partial^2 y_i}{\partial p^2} + \quad x_{i-1}(t - r_{ei-1}, p) *$$
$$E_{i-1}(y_{i-1}(t - r_{ei-1}, p))$$
$$- x_i(t - r_{ui}, p) U_i(y_i(t - r_{ui}, p))$$

with Neumann boundary condition

$$\frac{\partial x_i}{\partial p}(t, 0) = \frac{\partial x_i}{\partial p}(t, 1) = 0$$

$$\frac{\partial y_i}{\partial p}(t, 0) = \frac{\partial y_i}{\partial p}(t, 1) = 0 \qquad (2)$$

$$\frac{\partial d}{\partial p}(t, 0) = \frac{\partial d}{\partial p}(t, 1) = 0$$

$$\frac{\partial z}{\partial p}(t, 0) = \frac{\partial z}{\partial p}(t, 1) = 0$$

and initial conditions

$$\begin{align} x_i(t, p) &= \phi_i(p) \geq 0 \\ y_i(t, p) &= \phi_{n+i}(p) \geq 0 \qquad (3) \\ d(t, p) &= \phi_{2n+1}(p) \geq 0 \\ z(t, p) &= \phi_{2n+2}(p) \geq 0, \ 0 \leq p \leq 1, \ t \in \langle -r, 0 \rangle, \end{align}$$

where $x_i(t, p), y_i(t, p), i = 1, ..., n, d(t, p)$ and $z(t, p)$ are the concentration of the recycling matter in microorganisms, the available nutrients, detritus and zooplankton, respectively. The constants r_{**} stand for the discrete time delays in uptake and excretion of nutrient and decomposition of detritus, $r = max \{r_{**}\}$ and $0 < p < 1$.

From the biological viewpoint the functions F_i, U_i, E_i and M_i describe the growth, uptake, excretion and mortality rate of the living organisms, respectively and there exists $\bar{y}_i > 0$ such that $F_i(\bar{y}_i) = 0$ and $F_i(y_i)(y_i - \bar{y}_i) > 0$ for $i = 1, ..., n$ and similarly there exists $\bar{x}_n > 0$ such that $F_z(\bar{x}_n) = 0$ and $F_z(x_n)(x_n - \bar{x}_n) > 0$.

For detail explanation of function see (Kmet, 1996) and (Kmet, 2003).

In the previous work (Kmet, 1996) and (Kmet, 2003) we give a mathematical analysis of simple food web describing by ordinary differential equations and partial differential equations, respectively.

The global and local asymptotic behavior of the similar models of n species of microorganisms competing exploitatively for a single growth-limiting nutrient was studied in (Busenberg at al., 1990), (Candaten at al., 2003), (Ellermeyer, 1994), (Nisbet at al., 1983) and (Wolkowicz, 1997).

Note 1 *By Whu (1996) p. 37 Theorem 1.1 there exists unique mild solution of (1)-(3). Through the remainder of this paper, all solutions are mild solutions.*

Note 2 *Let us denote $X = (x_1, ..., x_{2n+2}) =: (x_1, ..., x_n, y_1, ..., y_n, d, z)$. For each continuous $\Phi = (\phi_1, ..., \phi_{2n+2}) : \langle -r, 0 \rangle X \langle 0, 1 \rangle \rightarrow R_+^{2n+2}$, there is a unique function $X(\Phi, .) = (x_1(\phi, .), ..., x_{2n+2}(\phi, .))$ called the solution of (1)-(3) through Φ. The nonnegative cone of C_+^{2n+2} defined by*

$$C_+^{2n+2} = \left\{ \begin{array}{c} \Phi \in C^{2n+2} | \phi_i(t,p) \geq 0, \; t \in \langle -r, 0 \rangle \\ and \; p \in \langle 0,1 \rangle \; and \; i = 1, \ldots, 2n+2 \end{array} \right\}.$$

PRELIMINARY RESULTS

The following basic proposition holds

Proposition 1 *The solution* $X(.,\Phi) = X(t,p) = (x_1(t,p), \ldots, x_{2n+2}(t,p))$ *of (1)-(3) exists for all* $\Phi \in C_+^{2n+2}$, *remains nonnegative and for all* $t > 0$

$$\int_0^1 (\sum_{i=1}^{2n+2} x_i(t,p)) dp = c,$$

where $c = \int_0^1 (\sum_{i=1}^{2n+2} \phi_i(p,0)) dp$.

Proof 1 *Local existence is standard (Wu, 1996). System of equations* (1) *describing the dynamic of recycling matter can be written in the form:*

$$\frac{d}{dt} x_i = d_{xi} \frac{\partial^2 x_i}{\partial p^2} + G_i(x) \; for \; i = 1, \ldots, 2n+2.$$

Let us denote

$R_+^{2n+2} = \{(x_1, \ldots, x_{2n+2}) | x_i \geq 0 \; for \; i = 1, \ldots, 2n+2\}$.

By straightforward calculation we get that $G_i(x_1, \ldots, x_{i-1}, 0, x_{i+1}, \ldots, x_{2n+2}) \geq 0$ *for* $i = 1, \ldots, 2n+2$ *and* $x_j \geq 0$ *for* $j = 1, \ldots, i-1, i+1, \ldots, 2n+2$. *We obtain that the region* R_+^{2n+2} *is positively invariant for our system. By adding up of right-hand side of* (1) *we get* $\sum_{i=1}^{2n+2} G_i(x) = 0$. *Integrate over the interval* $[0,1]$ *to obtain an equation for the derivative of total mass of recycling matter*

$$\frac{d}{dt}(\int_0^1 (\sum_{i=1}^{2n+2} x_i(t,p)) dp) =$$

$\int_0^1 \frac{d}{dt}(\sum_{i=1}^{2n+2} x_i(t,p)) dp = \int_0^1 \sum_{i=1}^{2n+2} (\frac{d}{dt} x_i(t,p)) dp =$

$$\sum_{i=1}^{2n+2} \int_0^1 d_{xi} \frac{\partial^2 x_i(t,p)}{\partial z^2} dp =$$

$$\sum_{i=1}^{2n+2} d_{xi}(\frac{\partial x_i}{\partial z}(t,1) - \frac{\partial x_i}{\partial z}(t,0)) = 0.$$

It follows easily that $\int_0^1 (\sum_{i=1}^{2n+2} x_i(t,p)) dp = c$

and $c = \int_0^1 (\sum_{i=1}^{2n+2} \phi_i(0,p)) dp$.

Note 3 *The nonnegative orthant is positively invariant for the model presented, and the initial value problem is well-posed in the sense that unique solutions exist for all* $t > -r$ *where* $r = max\{r_{**}\}$ *and depend continuously on the initial data and parameters.*

Let us denote

$$S_c^{2n+2} = \left\{ X \in C_+^{2n+2} + /x_i \geq 0, \int_0^1 \left(\sum_{i=1}^{2n+2} x_i \right) dp = c \right\}$$

and

$$S_c = \left\{ (0,y,0,0) \in S_c^{2n+2} / y_i \; are \; spatially \; constant \right\}.$$

Note that S_c is a trivial spatially constant equilibrium set.

Proposition 2 S_c^{2n+2} *is a positively invariant set of system* (1).

Proof 2 *It follows from Proposition 1.*

In this section we study the system $(1-3)$. We show that for $z = 0$ there exists a strictly positive equilibrium point lying on S_c^{2n+1}.

In the case of $z = 0$ and $r_{**} = 0$ the system (1) has the form :

$$\begin{aligned}
\dot{x}_i &= d_{xi} \frac{\partial^2 x_i}{\partial p^2} + x_i F_i(y_i) \\
\dot{y}_1 &= d_{y1} \frac{\partial^2 y_1}{\partial p^2} + x_n E_n(y_n) + \quad (4) \\
&\quad K_1 d - x_1 U_1(y_1) \\
\dot{y}_i &= d_{yi} \frac{\partial^2 y_i}{\partial p^2} + x_{i-1} E_{i-1}(y_{i-1}) - x_i U_i(y_i) \\
\dot{d} &= d_d \frac{\partial^2 d}{\partial p^2} - K_1 d + \sum_{i=1}^{n} x_i M_i(y_i)
\end{aligned}$$

where $K_i > 0$, $F_i = U_i - E_i - M_i$, $F_i(0) < 0$, $F_i'(y_i) > 0$, $E_i \geq \epsilon_i U_i$, $U_i \geq E_i$ and there exists $\bar{y}_i$ such that

$$F_i(\bar{y}_i) = 0 \; and \; F_i(y_i)(y_i - \bar{y}_i) > 0. \quad (5)$$

Let us denote $c_1 = \sum_{i=1}^{n} \bar{y}_i$.

Proposition 3 *For* $z = 0$ *and* $c > c_1$ *there exists a strictly positive spatially constant fixed point of (1)* $\alpha^* = (x^*, y^*, d^*)$ *in* S_c^{2n+1}.

Proof 3 *As follows from (5)* y^* *must satisfy* $y_i^* = \bar{y}_i$
for $i = 1,...,n$. *The other components of the fixed point*
α^* *in the interior of* S_c^{2n+1} *are the strictly positive*
solutions of the following $n+1$ *linear equations.*

$$
\begin{aligned}
-U_1(y_1^*)x_1 + E_n(y_n^*)x_n + K_1 d &= 0 \\
E_1(y_1^*)x_1 - U_2(y_2^*)x_2 &= 0 \\
E_{i-1}(y_{i-1}^*)x_{i-1} - U_i(y_i^*)x_i &= 0 \quad (6) \\
E_{n-1}(y_{n-1}^*)x_{n-1} - U_n(y_n^*)x_n &= 0 \\
M_1(y_1^*)x_1 + ... + M_n(y_n^*)x_n - K_1 d &= 0.
\end{aligned}
$$

Let us consider the matrix $P = J + \rho I$, *where*
$\rho > -max \; J_{ii}$ *and* J *is a matrix of linear equa-*
tions (6). P *is an irreducibile nonnegative matrix and*
so the machinery of the Frobenius theory of nonnegative
matrix applies. Since $l\,J = 0$, *also*

$$
l\,P = \rho l,
$$

where l *denotes the vector all of whose entries are 1. By*
the Perron-Frobenius theory (Gantmacher, 1959) there
is a unique positive right eigenvector s *associated with*
the eigenvalue ρ, *i.e.* $P\,s = \rho\,s$.

We can normalize to get

$$
x_i^* = (c - c_1)s_i / \sum_{i=1}^{n+1} s_i \quad (7)
$$

$$
d^* = (c - c_1)s_{n+1} / \sum_{i=1}^{n+1} s_i. \quad (8)
$$

There holds $(J + \rho I)(x^*, d^*) = \rho(x^*, d^*)$, *so*
$J(x,d) = 0$ *and* $x_i^* > 0$, $d^* > 0$ *for* $c > c_1$ *and*
$i = 1,...,n$. *It is evident that*

$$
x_1^* + ... + x_n^* + d^* = c - c_1.
$$

This proves the result.

STABILITY ANALYSIS

In this section we investigate the of spatially constant
equilibria for (1). The system (1) always has a trivial
equilibrium set S_c. If $c > c_1$ there exists a unique equi-
librium α^*. The stability of $\bar{x} \in S_c$ and α^* is determined
by the following eigenvalue problem.

As the Laplace operator has eigenvalues $-k^2, (k = 0, 1, \dots)$ with corresponding eigenfunction $cos(k\pi x)$, λ
is a characteristic value of (1) if and only if for some
$k = 0, 1, \dots$ satisfies the characteristic equation

$$
H(\lambda) \equiv \lambda I - k^2 D - J(\bar{x}) = 0.
$$

where D is a diagonal matrix with diffusion coefficients
and $J(\bar{x})$ is described in APPENDIX I.

Let us denote

$$
S_0 = \left\{ \begin{array}{c} (0, y, 0, 0) \in S_c^{2n+2}/y_i \text{ are spatially constant} \\ \text{and } y_i < \bar{y}_i \text{ for } i = 1, \dots, n-1 \end{array} \right\}.
$$

and suppose that $r_{ui} = r_{ei} =: r_i \text{ for } i = 1, \dots, n, 2n + 2$. Let us suppose that $F_n(y_n) < 0$ for all $y_n \geq 0$. Ap-
plying $Theorem$ 1 to the eigenvalue problem we obtain

Proposition 4 *If* $K_1 < min\left\{k^2\pi^2 d_{x2n+1}, \frac{\pi}{2r_{k1}}\right\}$ *then*
0 *is characteristic value of the linearization of* (1) *at* $\bar{x} \in S_0$ *with multiplicity* n *and all other characteristic values*
have negative real parts.

Proof 4 *Let* $\bar{x} \in S_0$. *With the translation* $y = x - \bar{x}$
we can write the system (1) as

$$
\dot{y} = D\frac{\partial^2 y}{\partial p^2} + DG(\bar{x})y + N(y), \quad (9)
$$

$N(y)$ *involves only the higher terms in* $y_1, \dots y_{2n+2}$.
The perturbation system of (1) and its associated eigen-
value problem are, respectively, written as follows:

$$
\dot{y} = D\frac{\partial^2 y}{\partial p^2} + DG(\bar{x})y \quad (10)
$$

$$
\lambda v = (-k^2\pi^2 D + J)v, \; k = 0, 1, \dots \quad (11)
$$

The eigenvalues of $-k^2\pi^2 D + J(\bar{x})$ *at* $\bar{x} \in S_0$ *are as*
follows

$$
l\lambda_{i,k} = e^{-\lambda_{i,k}r_i}(u_i - e_i) - m_i - d_{xi}k^2\pi^2
$$
$$
for \; i = 1, \dots, n
$$
$$
\lambda_{i,k} = -d_{xi}k^2\pi^2
$$
$$
for \; i = n+1, \dots, 2n.
$$
$$
\lambda_{2n+1,k} = -K_1 e^{-\lambda_{2n+1,k}r_{k1}} - d_{x2n+1}k^2\pi^2
$$
$$
\lambda_{2n+2,k} = e^{-\lambda_{2n+2,k}r_{2n+2}}(u_{2n+2} - e_{2n+2}) - m_{2n+2} -
$$
$$
-d_{x2n+2}k^2\pi^2
$$

Upon simplification, the characteristic equation be-
comes

$$
H(\lambda) = H_1(\lambda)H_2(\lambda)H_3(\lambda),
$$

where $H_1(\lambda) = \prod_{i=1}^{n,2n+2}(\lambda_{i,k} - e^{-\lambda_{i,k}r_i}(u_i - e_i) + m_i + d_{xi}k^2\pi^2)$,

$H_2(\lambda) = \prod_{i=n+1}^{2n+1} (\lambda_{i,k} + d_{xi} k^2 \pi^2)$,

$H_3(\lambda) = (\lambda_{2n+1,k} + K_1 e^{-\lambda_{2n+1,k} r_{k1}} + d_{x2n+1} k^2 \pi^2)$.

Denoted the three factor of $H(\lambda)$ by $H_1(\lambda)$, $H_2(\lambda)$ and $H_3(\lambda)$, respectively. The location of the roots of the quasi-polynomials $H_1(\lambda)$ and $H_3(\lambda)$ is accomplished by using the Pontryagins theorem. (Hale at al., 1993) contains a brief introduction of the Pontryagin's results, together with a proof of $Theorem$ 1, (see APPENDIX II $Theorem$ 1).

To apply the $Theorem$ 1 to the quasi-polynomial $H_3(\lambda)$, we first let $\omega = r_{k1} \lambda$ so that ω and λ have real parts of the same sign. Multiplying both side of $H_3(\lambda)$ by $r_{k1} e^{\lambda_{2n+1,k} r_{k1}}$ yields

$$\left(\omega + d_{x2n+1} k^2 \pi^2 r_{k1}\right) e^\omega + r_{k1} K_1 = 0$$

All roots of the equation

$$\left(\omega + d_{x2n+1} k^2 \pi^2 r_{k1}\right) e^\omega + r_{k1} K_1 = 0$$

have negative real parts if and only if

$$d_{x2n+1} k^2 \pi^2 r_{k1} > -1,$$
$$d_{x2n+1} k^2 \pi^2 r_{k1} + r_5 K_1 > 0$$

and

$$r_{k1} K_1 < \zeta \sin(\zeta) - d_{x2n+1} k^2 \pi^2 r_{k1} \cos(\zeta)$$

where ζ is the root of $\zeta = -d_{x2n+1} k^2 \pi^2 r_{k1} \tan\zeta$, $0 < \zeta < \pi$, if $d_{x2n+1} k^2 \pi^2 r \neq 0$ and $\zeta = \frac{\pi}{2}$ if $d_{x2n+1} k^2 \pi^2 r_{k1} = 0$.

If $K_1 < min\left\{ k^2 \pi^2 d_{x2n+1}, \frac{\pi}{2 r_{k1}} \right\}$ inequalities are satisfied and $H_3(\lambda)$ has roots with negative real parts.

If $\bar{x} \in S_0$ then $u_i - e_i - m_i < 0$ and by the similar way it can be shown, that all the roots of the following transcendental equations given by $H_1(\lambda)$ have negative real parts for all $i = 1, \ldots, n, 2n+2$.

The equation $H_1(\lambda) = 0$ is treated similarly by multiplying both side by $r_i e^{\lambda_{i,k} r_i}$ to obtain

$$\lambda_{i,k} - e^{-\lambda_{i,k} r_i} (u_i - e_i) + m_i + d_{xi} k^2 \pi^2 = 0,$$

for $i = 1, \ldots, n, 2n+2$

$$\left(\omega + \left(d_{xi} k^2 \pi^2 + m_i\right) r_i\right) e^\omega - r_i (u_i - e_i) = 0.$$

We can apply $Theorem$ 1. All roots of the equation

$$\left(\omega + \left(d_{xi} k^2 \pi^2 + m_i\right) r_i\right) e^\omega - r_i (u_i - e_i) = 0.$$

have negative real parts if and only if

$$\left(d_{xi} k^2 \pi^2 + m_i\right) r_i > -1,$$

$$\left(d_{xi} k^2 \pi^2 + m_i\right) r_i - r_i (u_i - e_i) > 0$$

and

$$-r_i (u_i - e_i) < \zeta \sin(\zeta) - \left(d_{xi} k^2 \pi^2 + m_i\right) r_i \cos(\zeta)$$

where ζ is the root of $\zeta = -\left(d_{xi} k^2 \pi^2 + m_i\right) r_i \tan\zeta$, $0 < \zeta < \pi$, if $\left(d_{xi} k^2 \pi^2 + m_i\right) r_i \neq 0$ and $\zeta = \frac{\pi}{2}$ if $\left(d_{xi} k^2 \pi^2 + m_i\right) r_i = 0$. The condition of $Theorem$ 1 are satisfied that means all roots of $H_1(\lambda) = 0$ have negative real parts. This prove the result.

NUMERICAL EXAMPLE

If $k = 0$ then $\zeta = \frac{\pi}{2}$ and

$$r_{k1} K_1 < \frac{\pi}{2}.$$

i.e. for

$$K_1 > \frac{\pi}{2 \, r_{k1}}$$

all spatially constant $\bar{x} \in S_0$ are unstable.

Let us consider $H_3(\lambda)$ as a complex function of two complex variables

$$\mathcal{F}(\omega, K_1) = \left(\omega + d_{x2n+1} k^2 \pi^2 r_{k1}\right) e^\omega + r_{k1} K_1$$

If we let $\omega = \alpha + i\beta$ and setting $\alpha = 0$. By $Theorem$ 1 we get that there exists

$$K_1^* = \frac{\pi}{2 \, r_{k1}}$$

for which $H_3(\lambda)$ has root with zero real part. $\omega_0(K_1^*)$ is a simple eigenvalue, therefore $\frac{\partial \mathcal{F}(\omega_0, K_1^*)}{\partial \omega} \neq 0$, by the implicit function theorem we get that

$$R_e \left(\frac{\partial \omega(K_1^*)}{\partial K_1} \right) = R_e \left(-\frac{\frac{\partial \mathcal{F}(\omega_0, K_1^*)}{\partial K_1}}{\frac{\partial \mathcal{F}(\omega_0, K_1^*)}{\partial \omega}} \right) =$$

$$\frac{r_{k1} * (\pi/2 - \cos(\omega_0))}{\left\| e^{i\beta} + i\beta e^{i\beta} \right\|^2} > 0.$$

Real part of eigenvalue $\omega = \alpha + i\beta$ crosses the imaginary axis transversely at K_1^*. For $k = 0$ 0 is the eigenvalue with multiplicity n which is a dimension of the family of equilibria set S_0. According to $Proposition$ 4 the rest eigenvalues have been found to have real parts negative.

Our numerical calculations are based on $Proposition$ 4 and iterative schemes for numerical solution of system of nonlinear parabolic equations with discrete time delays presented by (Pao, 1996), (Pao,

1999). Finite difference solution was constructed. To simulate iterative scheme of finite difference solution a routine using software Matlab was created.

The vector function $F = (F_1, \ldots, F_{2n+2})$ given by right-hand side of (1) is mixed quasimonotone and the Jacobi iterative scheme, which is unconditionally stable with respect to the mesh sizes can be applied for numerical solution of (1).

Mathematical model of nitrogen transformation cycle, which consists of nine partial functional differential equations with discrete time delays was used, (Kmet, 2002) to numerical calculation.

Decomposition rate of detritus K_1 and discrete time delay rate r_{k1} play a role of bifurcating parameters. For

$$K_1 > \frac{\pi}{2\,r_{k1}}$$

all spatially constant $\bar{x} \in S_0$ are unstable. Numerical calculations show that for K_1 near to K_1^* periodic solution occurs (Figure 2).
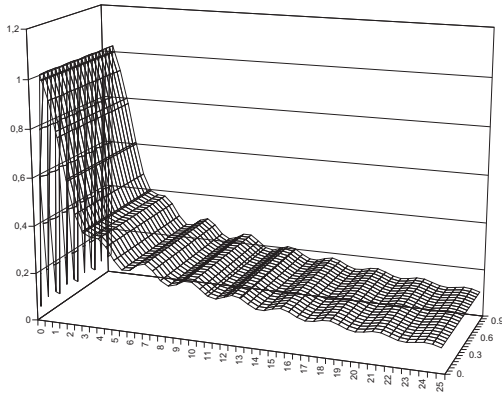
Figure 2. Numerical solution of (1) for $K_1 = 2.0$ and $r_{k1} = 0.8$ ($x_1(t,p)$).

If

$$K_1\,r_{k1} < \frac{\pi}{2}$$

all solutions converge to the set S_0 (Figure 3). If the time delay r_{k1} or the decomposition rate k_1 are relatively small ($K_1\,r_{k1} < \frac{\pi}{2}$), our predictions are identical to the predictions given by the corresponding PDE (Kmet, 2002).
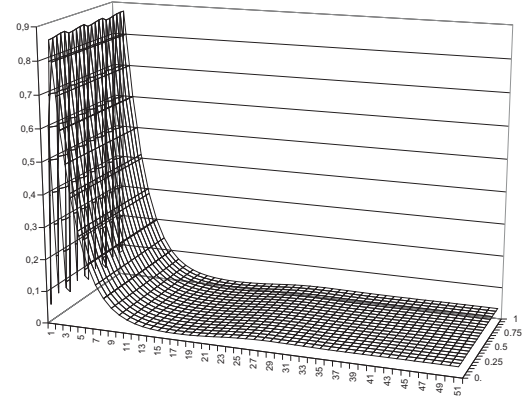
Figure 3. Numerical solution of (1) for $K_1 = 0.02$ and $r_{k1} = 0.8$ ($x_1(t,p)$).

CONCLUSION

In this paper, we have first contructed a simple food network described by the time-delayed partial differential equations. Then we have verified the existence and stability of equilibria of food network. Finite difference solution was constructed. To simulate iterative scheme of finite difference solution a routine using software Matlab was created. Our simulation results demonstrate that periodic solution occurs.

ACKNOWLEDGEMENTS

APPENDIX I

For $\bar{x} \in S_c$ the matrix $J(\bar{x})$ has the form:

$$J(\bar{x}) = \begin{pmatrix}
j_1 & 0 & 0 & . & 0 & 0 \\
0 & \ddots & j_n & . & 0 & 0 \\
-u_1 & 0 & e^{-\lambda r_n} e_n & . & K_1 e^{-\lambda r_{k1}} & e^{-\lambda r_{2n+2}} e_{2n+2} \\
e_1 & 0 & 0 & . & 0 & 0 \\
. & . & . & . & . & \ddots \\
0 & e_{n-1} & -u_n & . & 0 & 0 \\
m_1 & \cdots & m_n & . & -K_1 e^{-\lambda r_{k1}} & 0 \\
0 & 0 & 0 & . & 0 & j_{2n+2}
\end{pmatrix}$$

and $e_i = E_i(\bar{x})$, $m_i = M_i(\bar{x})$, $u_i = U_i(\bar{x})$, $j_i = e^{-\lambda r_i}(u_i - e_i) - m_i$ $i = 1, \ldots, n$ and $2n+2$.

APPENDIX II

Theorem 1 *All roots of the equation $(z + a)e^z + b = 0$, where a and b are real, have negative real parts if and only if*

$$
\begin{aligned}
a &> -1 \\
a + b &> 0 \\
b &< \zeta \sin \zeta - a \cos \zeta
\end{aligned}
\tag{12}
$$

where ζ is the root of $\zeta = -a \tan \zeta$, $0 < \zeta < \pi$, if $a \neq 0$ and $\zeta = \frac{\pi}{2}$ if $a = 0$.

REFERENCES

Busenberg, S., Kumar, S. K., Austin, P., and Wake, G. (1990) The Dynamics of a Model of a Plankton-Nutrient Interaction. *Bull. of Math. Biol.*, 52:677–696.

Candaten, M., and Rinaldi, S. (2003) Peak-to-paek dynamics in food chain models. *Theoretical Population Biology*, 63:257–267.

Ellermeyer, S. F. (1994) Competition in the Chemostat:Global Asymptotic Behavior of a Model with Delayed Response in Growth. *SIAM J. APPL. MATH.*, 54:456–465.

Fath, B. D., Scharler, U. M., Ulanowicz, R. E., and Hannon, B. (2007) Ecological network analysis: network construction. *Ecological Modelling.*, 208:49 – 55.

Gantmacher, F. R. (1959) *The theory of matrices*. New York, Chelsea.

Hale, L. S., and Verduyn Lunel, S. M. (1993) *Introduction to Functional Differential Equations*. Springer Verlag, New York.

Halnes, G., Fath, B. D., and Liljenstrom, H. (2007) The modified niche model:Including detritus in simple structural food models. *Ecological Modelling.*, 208:9 – 16.

Kmet, T. (1996) Material recycling in a closed aquatic ecosystem. I. Nitrogen transformation cycle and preferential utilization of ammonium to nitrate by phytoplankton as an optimal control problem. *Bull. Math. Biol.*, 58:957 – 982.

Kmet, T. (1996) Material recycling in a closed aquatic ecosystem. II. Bifurcation analysis of a simple food-chain model. *Bull. Math. Biol.*, 58:983 – 1000.

Kmet, T. (2002) Diffusive Mathematical Model of Nitrogen Transformation Cycle in Aquatic Environment. *Folia Fac. SCi. Nat. Univ. Masarykiane Brunensis, Mathematica* 11:105–114.

Kmet, T. (2003) Parabolic System Modelling a Simple Food Web in a Closed Aquatic Ecosystem. *Proceeding 2nd International Conference APLIMAT* 149–158.

Nisbet, R. M., McKinstry, J., and Gurney, W. S. (1983) A "Strategic" Model of Material Cycling in a Closed Ecosystem. *Mathematical Biosciences*, 64:99 – 113.

Pao, C. V. (1996) Dynamic of Nonlinear Parabolic Systems with Time Delays. *J. Math. Anal. Applic.*, 198:751–779.

Pao, C. V. (1999) Numerical Methods for System of Nonlinear Parabolic Equations with Time Delays. *J. Math. Anal. Applic.*, 240:249–279.

Ulanowicz, R. E. (1972) Mass and Energy Flow in Closed Ecosystems. *J. Theor. Biol.*, 34:239 – 253.

Wolkowicz, G. S. K., and Xia, H. Global Asymptotic Behavior of a Chemostat Model with Discrete Delays. *SIAM J. APPL. MATH.*, 57:1019–1043.

Wolkowicz, G. S. K., Xia, H., and Ruan, S. (1997) Competetion in the Chemostat:A Distributed Delay model and its Global Asymptotic Behavior. *SIAM J. APPL. MATH.*, 57:1281–1310.

Wu, J. (1996) Theory and Applications of Partial Functional Differential Equations. *Spinger, Applied Math. Sciences*, 119.

AUTHOR BIOGRAPHIES

TIBOR KMET was born in 1956. His email is `tkmet@ukf.sk` and his personal webpage is at `http://www.ki.fpv.ukf.sk`.

A DYNAMIC RULE IN CELLULAR AUTOMATA

C. E. Somarakis
PhD Student
School of Electrical &
Computer Engineering,
*National Technical University
of Athens*, GR
cslakonas@yahoo.gr

G. P. Papavassilopoulos
Professor
School of Electrical &
Computer Engineering,
*National Technical University
of Athens*, GR
yorgos@netmode.ntua.gr

F. E. Udwadia
Professor
Viterbi School
of Engineering,
*University of Southern
California*, USA
fudwadia@usc.edu

Abstract—In this paper a new rule in cellular automata is introduced. Through phenomenological studies and basic numerical explorations, its generic dynamical behavior is outlined. Properties such as periodic and complex behavior are reported and studied at an elementary level. Further research on the rule is recommended since under certain variations (many of which are reported) this rule can be a reliable model for simulating procedures among various fields of study.

Index Terms—Cellular Automata, Phase Transition, Complex Dynamics

I. INTRODUCTION AND HISTORICAL NOTES

Cellular automata are among the simplest mathematical representations of complex dynamical systems. They are a class of spatially and temporally discrete, deterministic systems characterized by local interaction and an inherently parallel form of evolution. The history of cellular automata can be traced back to 1948, when *J. L. von Neumann* introduced them to study simple biological systems [Neumann(1948)]. As his work progressed, *Neumann* started to cooperate with *S. M. Ulam*, who introduced him to the concept of cellular spaces [Wolfram(2000)]. These described the physical structure of a cellular automaton, i.e., a grid of cells which can be either "on" or "off". Shortly after, *A. M. Turing* proposed, in 1952, a model that illustrated reaction-diffusion in the context of morphogenesis [Turing(1952)]. In the 1970s, cellular automata found their way to one of the most popular applications called *simulation games*, of which *J. H. Conway's*, *Game of Life* is probably the most famous [Gardner(1970)]. However, the widespread popularisation of these systems was achieved in the 1980s through the work of *S. Wolfram*. Based on empirical experiments using computers, he gave an extensive classification of cellular automata as mathematical models for self-organising statistical systems [Wolfram(2000)]. *Wolfram's* systematic research is to relate cellular automata to all disciplines of science (e.g., sociology, biology, physics, mathematics, economy, *etc*).
In this paper, we will introduce a new dynamical rule (we shall call it $\mathcal{F}$-*rule*). This model came as a result of comprehensive research in dynamical structures of networks as these were constructed out of game theory and statistical physics. The $\mathcal{F}$-rule is an outgrowth of the *Prisoner's Dilemma* game as it is stated by *Nowak et al* [Nowak and May(1992)]. In fact, the very first introduction of the $\mathcal{F}$-rule was in [Wiederien and Udwadia(2004)]. The present paper can be considered as a continuation of [Wiederien and Udwadia(2004)] where we consider more general games played among the neighbors. We consider the model as a discrete dynamical system evolving in a discrete state space. Our aim is to explore its properties and outline the emerging dynamics and the degree of complexity. For some values of the parameters of the local interaction drastically different behaviors are observed. Our aim is to characterise the long-term behavior in terms of average values as well as the identification of critical values of the parameters where various patterns (behavior or phase transitions) occur.

A. Notations and Definitions

Before introducing our model, it is necessary to present four generic characteristics of cellular automata. These primary concepts will assist in getting a picture of the rule's behavior both locally and globally.

1) The State Space: We define an space $\mathcal{L} : \mathbb{N}^d$, where $\mathbb{N}$ is the set of natural numbers, as the discrete state space. This is the lattice of *d*-dimensional sites upon which our automata live, and their dynamics unfold. Every individual site can be defined by a $(1 \times d)$ vector $\overrightarrow{x}$. In this paper, we mainly consider two-dimensional spaces (see Fig. 1(a)).
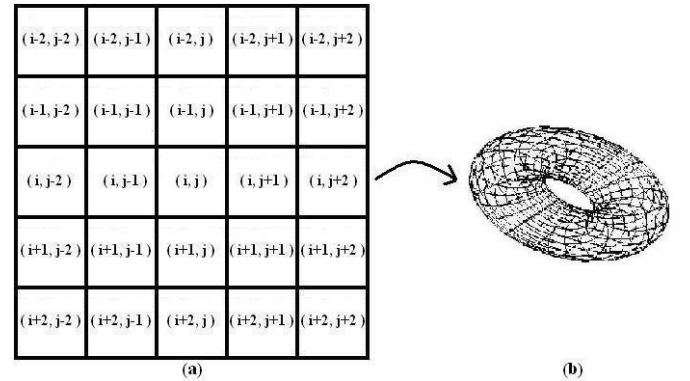
Fig. 1. (a)The plane lattice is the state space of our system. Every cell can be defined as site of a regular 2D matrix. (b)The periodic conditions imply that the opposite edges of the lattice get glued to form a torus.

2) Local Value Space: Each cell $\overrightarrow{x} \; \epsilon \; \mathcal{L}$ can assume only a finite number of different values:

$$\sigma(\overrightarrow{x};t) \; \epsilon \; \Sigma = \{0, 1, 2, ..., k-1\} \times \mathbb{N} \qquad (1)$$

where $\sigma(\overrightarrow{x};t)$ is the value of $\overrightarrow{x}$ at time $t\ \epsilon\ \mathbb{N}$. In our paper, we set $k = 2$. The set of possible states at time t can be either $\sigma = 0$ or $\sigma = 1$. In the presented simulations below, a black-coloured site means a site in 0 state, whereas a white-coloured site is in state 1.

3) Neighbourhood of a cell: Every cell changes it's state after communicating with it's neighboring cells. We note by $\mathcal{N}(\overrightarrow{x}, r)$ the range-r neighbourhood of $\overrightarrow{x}$,without $\overrightarrow{x}$ itself, and by $\underline{\mathcal{N}}(\overrightarrow{x}, r)$ the range-r neighborhood, including $\overrightarrow{x}$, i.e.

$$\mathcal{N}(\overrightarrow{x}, r) = \{\overrightarrow{y} \epsilon \mathcal{L} : 0 < ||\overrightarrow{x} - \overrightarrow{y}||_\infty < r\} \quad (2)$$

$$\underline{\mathcal{N}}(\overrightarrow{x}, r) = \{\overrightarrow{y} \epsilon \mathcal{L} : 0 \leq ||\overrightarrow{x} - \overrightarrow{y}||_\infty < r\} \quad (3)$$

where $|| * ||_\infty : \mathbb{N}^d \to \mathbb{N}$ is the infinity norm. In our work we fix $r = 1$ which means that for the neighbourhood of a given center site $\overrightarrow{x}$ is the set of sites which are immediately adjacent to $\overrightarrow{x}$. This scheme is generally known as *Moore scheme* [Wolfram(2000)], [Ilachinski(2002)]. Having defined the basic topology of our lattice, we denote by $\mathcal{P}(N^2)$ the *power set* of $\mathcal{L}$. This is the set of all possible global states $G(t) = \{\sigma(\overrightarrow{x};t)\}\ \forall\ \overrightarrow{x}\ \epsilon\ \mathcal{L}$ at time t. For $k = 2$, $\text{card}\mathcal{P}(N^2)=2^{N^2}$.

4) Boundary Conditions: Although cellular automata are assumed to live on infinitely large lattices, computer simulations necessarily run on finite sets. Thus, it is also essential to define conditions on the boundaries of the lattices. Among various types of boundary conditions that have been proposed [Wolfram(2000)], [Ilachinski(2002)], in this paper, finite lattices are exclusively considered with *periodic* boundary conditions as depicted in Fig. 1(b).

B. The $\mathcal{F}$-Rule

We now move on to define the transition rule under which the states of sites in $\mathcal{L}$ are updated. To every $\overrightarrow{x}\ \epsilon\ \mathcal{L}$, we assign a cost function $V(\overrightarrow{x}, \mathcal{N}(\overrightarrow{x})) : \Sigma^2 \to W = \{a, b, c, d\} \subset \mathbb{R}$ such that:

$$V(\overrightarrow{x}, \overrightarrow{y}) = \begin{cases} a & \text{if } \sigma(\overrightarrow{x};t) = \sigma(\overrightarrow{y};t) = 0 \\ b & \text{if } \sigma(\overrightarrow{x};t) = 0, \sigma(\overrightarrow{y};t) = 1 \\ c & \text{if } \sigma(\overrightarrow{x};t) = 1, \sigma(\overrightarrow{y};t) = 0 \\ d & \text{if } \sigma(\overrightarrow{x};t) = \sigma(\overrightarrow{y};t) = 1 \end{cases} \quad (4)$$

The payoff matrix of (4) is as in Fig. 2(a). By this figure, one can perceive a primary physical hypothesis of these cost values. They, in fact, reflect the tension of local interactions between individual cells. So, the initial step of our rule is that for a fixed value $\sigma(\overrightarrow{x};t)$ and for every $\sigma(\overrightarrow{y};t)$ of $\overrightarrow{y}\ \epsilon\ \mathcal{N}(\overrightarrow{x})$ we adjust a number $w \epsilon W$ to $\overrightarrow{x}$. For a fixed arrangement of states in $\mathcal{N}(\overrightarrow{x})$, site $\overrightarrow{x}$ receives an overall cost

$$\mathcal{V} = \sum_{\overrightarrow{y} \epsilon \mathcal{N}(\overrightarrow{x})} V(\overrightarrow{x}, \overrightarrow{y}) \quad (5)$$

Finally, $\forall\ \overrightarrow{x}\ \epsilon\ \mathcal{L}, \overrightarrow{z}\ \epsilon\ \underline{\mathcal{N}}(\overrightarrow{x})$ the update rule is defined to be:

$$\sigma(\overrightarrow{x};t+1) = \mathcal{F}(\sigma(\overrightarrow{z};t)) = \sigma(\overrightarrow{z};t)$$

where $\overrightarrow{z}$ is such that

$$\mathcal{V}(\overrightarrow{z}) = \max_{\overrightarrow{z} \epsilon \underline{\mathcal{N}}(\overrightarrow{x})} \mathcal{V}(\overrightarrow{z}) \quad (6)$$

In the following, we will symbolize by $\mathcal{F}_{(a,b,c,d)}$ a sufficiently defined model.

1) Properties and Remarks: It can be easily shown that:

1. $\mathcal{F}_{(a,b,c,d)} \equiv \mathcal{F}_{(a+q,b+q,c+q,d+q)} \quad \forall\ q\ \epsilon\ \mathbb{R} \quad$ (7a)
2. $\mathcal{F}_{(a,b,c,d)} \equiv \mathcal{F}_{(aq,bq,cq,dq)} \quad\quad\quad \forall\ q\ \epsilon\ \mathbb{R}_+ \quad$ (7b)

These two properties declare the model's invariance under additivity and multiplication, which imply that the dynamics occuring from some range of parameters image a more generic parameter subspace.

At this point, we should make two remarks concerning the nature of this rule. First, this model is a purely deterministic dynamic procedure. Neither stochastic fluctuations, nor any sort of noise, affect the evolution of the cells. Secondly, a carefull insight into the rule's local behavior reveals that the interactions are in fact within a range $r = 2$ rather than $r = 1$. In the previous paragragh we noted that the decision of $\sigma(\overrightarrow{x};t+1)$ depends on the values of $\mathcal{V}(\overrightarrow{z}\ \epsilon\ \underline{\mathcal{N}}(\overrightarrow{x}))$ at time t. However, every cost $\mathcal{V}$ is, subsequently, a result of local interaction between the neighbours of the center site and their own neighbors. What, actually, happens is the local alternation of two, interconnected, procedures. If $G(t), J(t)$ are the instant global state and cost values of $\mathcal{L}$ respectively, then $\mathcal{F}$-rule can be separated into alternating between sub-rules:

$$\mathcal{F} = \begin{cases} f_1 : G(t) \to J(t) \\ f_2 : J(t) \to G(t+1) \end{cases} \quad (8)$$

where f_1 is the function that sets the costs out of the states and f_2 is the function that updates the (new) states out of the costs. Next questions arise intuitively. "What is so interesting about this rule?", "What kind of behavior is this rule capable of exhibiting?". In the following, we will attempt to provide, hopefully intriguing, answers.

II. Evolution from Simple Seeds

To this extent, we shall furtherly simplify the rule by fixing the cost values $a = 0$, $d = 1$ and $b, c > 0$ that are free to vary. We will provide sufficient evidence that the variation of these two parameters causes intricate phenomena. In this section, we will discuss the spatial evolution of the rule, starting from a specific initial configuration known as *simple seeds*. Here, the lattice consists of a zero-valued (black) site, in a background of non zero-valued (white) sites (see Fig. 2(b1)). The growth of cellular automata from such setup provides models for a variety of physical and other phenomena such as crystal growth [Packard(1985)]. In general, this type of initial conditions will help us to understand the complexity of the rule's dynamics in both space and time.

A. An analytical approach

Like the vast majority of cellular automata, $\mathcal{F}$-rule generates a -simple to understand, difficult to solve- evolution. One will realise this, by simply speculating the rule's formalistic definition (eq. (4)-(6)): three equations for the update of a single cell (!). By the last simplification of the $\mathcal{F}$-Rule we have

cut it's control parameters down to two (b and c). Nevertheless, it is may be unwise to begin setting arbitrary values for the b and c costs. It is highly possible either to keep selecting values that are "not-interesting" or, at least, exhaustingly difficult to organize the potential "interesting" behaviors. The simple seed setup is a, convenient enough, pattern as to run the first steps of the rule "by hand". In Fig 2(b1) a window of our initial lattice is presented, where every value is at $\sigma(\overrightarrow{y}; 0) = 1$, apart from the central site $\overrightarrow{x}$ where $\sigma(\overrightarrow{x}; 0) = 0$. By f_1, the next step is to assign each site a cost value. In Fig. 2(b1) we present the results for our neighborhood of cells: It is easy to see that due to the symmetry of the pattern the arising costs are:
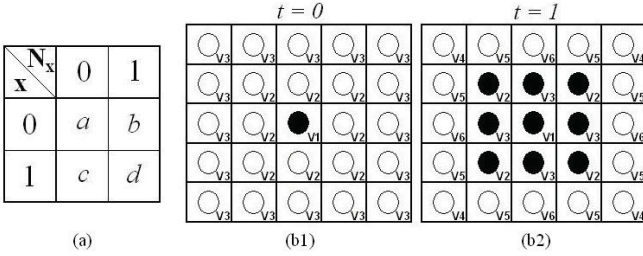
Fig. 2. (a)The $\mathcal{F}$ payoff table. It is derived from (4) and specifies the effect between contigious cells. Namely it defines the local interaction of sites at every time step. (b)The simple seed configuration. (c) For (b, c) values: $7 + c < 8b$ and $b > 1$,zero state expands to its neighboors. Because of the special initial conditions, symmetry is, throughout the evolution, preserved.

$$\left\{ \mathcal{V}_1 = 8b \quad \mathcal{V}_2 = 7 + c \quad \mathcal{V}_3 = 8 \right\}$$

For the remaining cells of the lattice the cost values are, obviously, $\mathcal{V} = \mathcal{V}_3 = 8$. Hence, at $t = 0$ the turn-up costs are $8b$, $7 + c$, 8. Now, if $7 + c \geq 8b$ we should expect that our sole site will step to *white* and stay unchanged for the rest of the time. Else, if $7 + c < 8b \leq 8$ there should be no change in the states of the cells: $\overrightarrow{x}$ has the largest cost among its neighbors $\mathcal{N}(\overrightarrow{x})$, while the latter cells adjoin to cells with larger than $\mathcal{V}(x) = \mathcal{V}_1$ costs. Therefore, our pattern remains static forever. The last case is when $7 + c < 8b$ and $b > 1$ where we expect that the neighborhood will turn to the one in Fig 2(b2). The costs at $t = 1$ are:

$$\left\{ \begin{array}{lll} \mathcal{V}_1 = 0 & \mathcal{V}_2 = 5b & \mathcal{V}_3 = 3b \\ \mathcal{V}_4 = 7 + c & \mathcal{V}_5 = 6 + 2c & \mathcal{V}_6 = 5 + 3c \end{array} \right\}$$

The future evolution of our automaton depends on the relative values of the b and c parameters as they were set by the $\mathcal{V}$-costs. The equalities define the (b, c) phase transition diagram of the model's dynamic behavior. In Fig 3(a), we have sketched a raw phase transition pattern on the (b, c) plane. For the simple-seed setup, seven different schemes have been identified (see Figs. 4 and 5):

- **P1**: The system *directly* (i.e. without transient states) evolves to homogeneous state where all cells attain the same value (i.e. $\sigma = 1$).
- **P2**: The system directly evolves to periodic behavior of period 2.
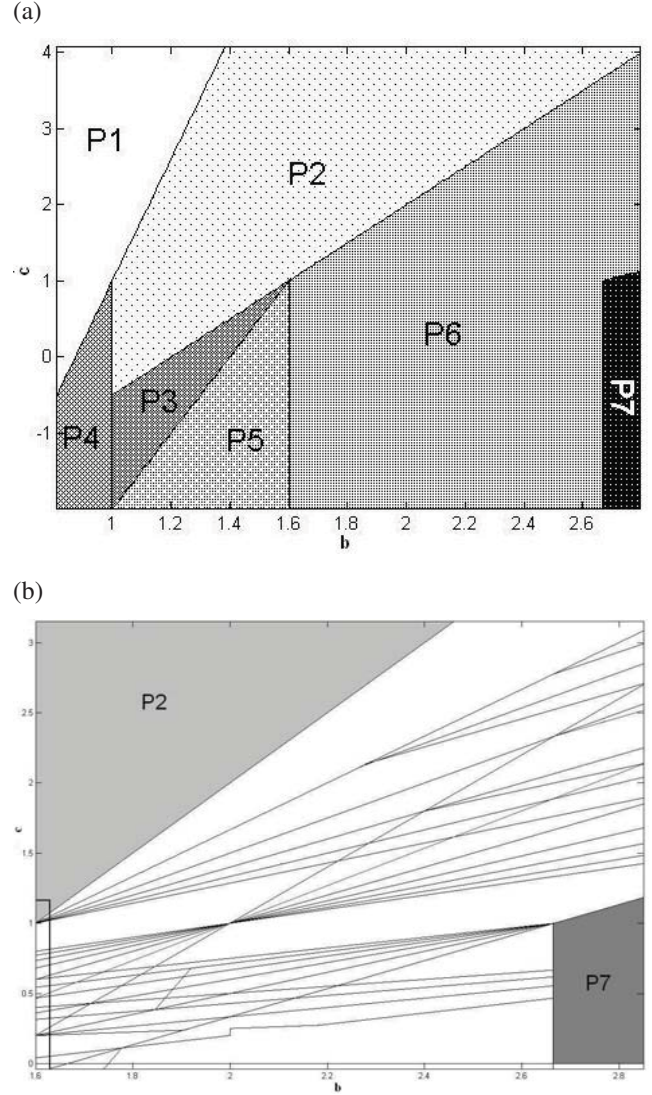
Fig. 3. (a) A characteristic window of the (b, c)-plane, where we have sketched a generic phase transition diagram. While regions P1,P2,P3,P4,P5 and P7 appear regular dynamics, P6 is rich in irregular behavior (see Fig(1a)).(b) The phase transition diagram of a part of P6 region ($1.6 < b < 2.85$). See text for details. The linear boundaries were numerically estimated. The calculation's precision is 10^{-3}. The bounded rectangular area shown at the lower left corner of the diagram is thoroughly explored (see §IIB1).

- **P3**: The system directly evolves to periodic behavior of period 3.
- **P4**: The system directly evolves to periodic behavior of period 1.
- **P5**: The system expands in the first iteration and remains static ever after.
- **P6**: The system expands to an irregular complex behavior.
- **P7**: The system expands uniformly. At each time step, a regular pattern with a fixed density of zero sites is produced.

The boundaries of the diagram are, obviously, the set of critical (bifurcation) values. For instance, at the P5, P6 boundary we

	t=0	t=1	t=2	t=3		t=0	t=1	t=2	t=3
P1	·				P4	·	·	·	·
P2	·	■	·	■	P5	·	■	■	■
P3	·	■	+	·	P7	·	■	■	■
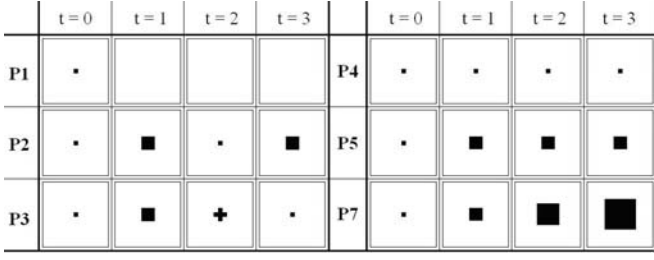
Fig. 4. Dynamic evolution of the *regular* regions of (b, c)-plane P1,P2,P3,P4,P5,P7. Plain periodic patterns with uniform structure may only appear.

step on the critical vertical line of $b = 8/5$. Similarly, the P6 to P7 boundary consists of the connected lines $b = 8/3$ and $c = b - 5/3$. These values together with the rest of the limit lines have been, analytically, derived from the above procedure. We directly realize, though, that it only takes few steps for the system to become forbiddingly complicated towards any analytic approach. So we have no choice but running computer simulations for any further analysis. For the moment we are, primarily, interested in investigating this "irregular" P6 region.

B. The P6 region

Contrary to the other sections, P6 is a parameter sub-space, where the automaton appears to exhibit extraordinary dynamics. Characteristic images of this region are presented in Fig. 5. Unlike the other six regions, patterns have no simple faceted form and in most cases non-uniform interior. Additionally, due to symmetric initial states, these patterns are completely invariant under all the rotation and reflection symmetry transformations. In order to identify part of the P6 dynamics, we implemented various techniques presented below. Our first step was to numerically estimate the P6 transition phase space, a part of which is presented in Fig. 3(b). At first sight, one may observe that P6 is a collection of mutually disjoint subsets of (b, c)-plane. These subsets have occurred out of subsequent straight line intersections. This is an expected outcome, if one intuitively expands those first steps of the algorithm back in §IA. In each of these subsets, our system generates the same pattern and, of course, exhibits the same dynamic behavior. So the following question would be "What is this dynamic behavior?".

1) Global dynamics on a finite, fixed $\mathcal{L}$: Our first approach was to keep the lattice size constant at N=50. On this 50×50 space $\mathcal{L}$ with periodic boundaries, we explored the system's long term behavior selecting cost values from the rectangular sketched in Fig. 3(b). Each simulation test was run for at most $20,000$ iterations. The observed results are classified in three qualitative classes:

A After a transient behavior, all sites of $\mathcal{L}$, eventually, attain the same value.

B After a transient behavior, all sites of $\mathcal{L}$, converge to *periodic (period-T)* motion. There

$$\exists\, t_{trans} \,\epsilon\, \mathbb{N}: \; \sigma(G; t) = \sigma(G; t + T) \; \forall\, t < t_{trans}$$

where $G \,\epsilon\, \mathcal{P}(N^2)$.

C The dynamic evolution does not converge to any of the previous two classes. A typical C-class behavior is the one that after a sufficiently large number of iterations $(20,000)$ the pattern neither, strictly, repeats itself, nor turns out to a homogeneous state.

In Table I, we present the numerical results in details.

Class	Parameter c range	Transient time (t_{trans})	Steady state ($t > t_{trans}$)
A	$(0.0000, 0.0059)$	100	$\sigma(G; t) = 0$
A	$(0.0600, 0.2062)$	643	$\sigma(G; t) = 0$
A	$(0.2063, 0.2166)$	1388	$\sigma(G; t) = 0$
C	$(0.2167, 0.2374)$	-	*no convergence*
C	$(0.2375, 0.2999)$	-	*no convergence*
B	$(0.3000, 0.3285)$	872	$\sigma(G; t) = \sigma(G; t + 2)$
B	$(0.3286, 0.3899)$	118	$\sigma(G; t) = \sigma(G; t + 2)$
C	$(0.3900, 0.4124)$	-	*no convergence*
C	$(0.4125, 0.4916)$	-	*no convergence*
B	$(0.4917, 0.5333)$	2569	$\sigma(G; t) = \sigma(G; t + 4)$
B	$(0.5334, 0.5642)$	802	$\sigma(G; t) = \sigma(G; t + 4)$
B	$(0.5643, 0.6187)$	1111	$\sigma(G; t) = \sigma(G; t + 12)$
B	$(0.6188, 0.6499)$	2935	$\sigma(G; t) = \sigma(G; t + 4)$
B	$(0.6500, 0.7199)$	914	$\sigma(G; t) = \sigma(G; t + 6)$
B	$(0.7200, 0.7666)$	515	$\sigma(G; t) = \sigma(G; t + 6)$
B	$(0.7667, 0.7999)$	299	$\sigma(G; t) = \sigma(G; t + 6)$
B	$(0.8000, 0.8249)$	199	$\sigma(G; t) = \sigma(G; t + 4)$
B	$(0.8250, 1.0357)$	494	$\sigma(G; t) = \sigma(G; t + 4)$
B	$(1.0358, 1.0416)$	40	$\sigma(G; t) = \sigma(G; t + 2)$
B	$(1.0417, 1.0490)$	77	$\sigma(G; t) = \sigma(G; t + 2)$
B	$(1.0500, 1.0624)$	33	$\sigma(G; t) = \sigma(G; t + 2)$
B	$(1.0625, 1.0833)$	22	$\sigma(G; t) = \sigma(G; t + 2)$
B	$(1.0834, 1.1249)$	29	$\sigma(G; t) = \sigma(G; t + 2)$

TABLE I
*

Results from numerical exploration of the bounded phase transition area in Fig. 3(b). For each subset of the diagram we generated simple seed pattern and let the simulation run for approx. $20,000$ iterations. The results are presented above. In most cases our system falls in periodic states after some transition time, while other control values of (b, c) generate patterns that seem not to converge. Note that the maximum number of iterations is much greater than the maximum transient time reported. This secures somehow the reliability of our analysis. One should not expect an everlasting aperiodic evolution on a finite space $\mathcal{L}$, anyway.

C. Complex Behavior

In this section, we will provide sufficient evidence to argue that our automaton is rightfully characterized as complex. The cost parameters come mainly from the P6 region, although another cost setup is also presented. The system initiates from a simple-seed configuration and evolves on a large size (practically infinite) matrix $\mathcal{L}$.

1) Stability: An important qualitative characterization of complex behavior in deterministic dynamical systems is their sensitivity to small changes made to the initial states. A convenient measure for quantifying the divergence between evolutions starting from two similar binary valued global states, $\sigma(G_1)$ and $\sigma(G_2)$ is the *Hamming distance*,

$$H_{\sigma(G_1), \sigma(G_2)}(t) = \sum_{i,j=1}^{N} \sigma((i, j)_1; t) \oplus_2 \sigma((i, j)_2; t) \quad (9)$$
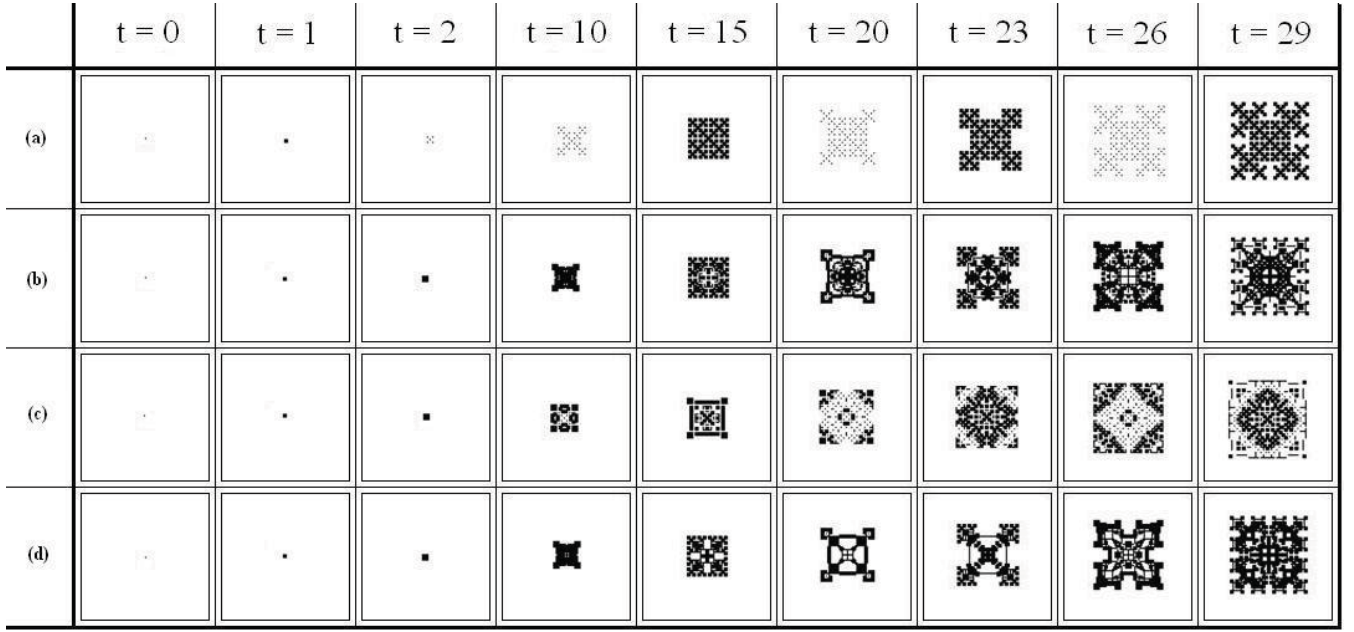
Fig. 5. Examples of classes of patterns generated by evolution of $\mathcal{F}$-rule. For each of these automata, a sequence of frames shows the two-dimensional configuration and evolution.(a) $\mathcal{F}_{(0,1.65,1.0834,1)}$,(b) $\mathcal{F}_{(0,1.65,0.4125,1)}$,(c) $\mathcal{F}_{(0,2.6,0.2252,1)}$,(d) $\mathcal{F}_{(0,1.7,0.4,1)}$

where $\oplus_2$ is the *modulo-2* sum operator.

Hamming distance is a fundamendal measure in Coding Theory and has already been used in 1-D cellular automata to primarily characterise the emerging difference from two *nearby* initial configurations [Wolfram(2000)]. These configurations are patterns that differ at only one site value (i.e. $H(0) = 1$). Suppose that G_1 and G_2 are global configurations which differ at one site. For simple update rules, the difference remains localized to a few sites, and the total Hamming distance tends rapidly to a small constant value or remains bounded. On the other hand, complex rules generate a growing with time $H(t)$ curve. This implies their instability to small perturbations. This is the case presented in Fig. 6. There, $\mathcal{F}_{(0,1.65,1.0834,1)}$ unfolds on a 560×560 square space. The 6(b) plot depicts this *rapid* divergence of nearby initial setups. The average Hamming distance, smoothed over many time steps, behaves as $H(t) \propto t^{1.95}$. As it is stated in [Wolfram(2000)], [Ilachinski(2002)], in case of one-dimensional cellular automata, unbounded $H(t)$ characterizes rules that generate chaotic patterns.

2) Growth Dimensions: The limiting structure of patterns generated by the growth of cellular automata from simple seeds can be characterized by various *growth dimensions*. Growth dimensions, in general, describe the logarithmic asymptotic scaling of the total sizes of patterns with their linear dimensions. The type of dimension we will make use of depends on the *boundary* of the pattern. The boundary may be defined as the set of black sites that can be reached by some path on the lattice that begins at "infinity" and does not cross any zero (black) sites. This set of limiting cells can thus be

found by a simple recursive procedure:

$$D_g = \lim_{t \to \infty} \frac{log\overline{N}(t)}{\log t} \qquad (10)$$

Where $\overline{N}(t)$ is the number of boundary (black) cells of our system at time t. In Fig. 6a we see the evolution of $\mathcal{F}_{(0,1.65,1.0834,1)}$ and the dendritic geometry of it's boundaries. In Fig. 6c we present this logarithmic plot in which $D_g = 1.8 \pm 0.1$.

3) Space-Time Patterns: A direct and most applicable technique of examining an asymptotical behavior is through a state subspace analysis. One may choose to define $Poincar\acute{e}$-like sections and study the dynamic evolution of this subspace. Specifically, we consider the center horizontal or the diagonal line of the two-dimensional lattice with time (see Fig. 9(a)). The reason we choose these specific sections is rather intuitive. They are the two axes of symmetry and a view of their evolution in time would be most enlightening. In Figs. 7(b) and 7(c) we present examples of space-time patterns from the center-horizontal and diagonal lines, respectively. Fig. 7(b) presents the dynamics of rule $\mathcal{F}_{(0,1.61,0,1)}$ after 250 iterations. The evolution converges to this scale invariant, image. Its fractal dimension is estimated at $D_F = 1.882 \pm 0.002$. In Fig. 7(c) we present the pattern generated by $\mathcal{F}_{(0.5,0.4,1,0.22)}$. This rule produces the well-known $Sierpi\acute{n}ski\ gasket$, the dimension of which is analytically calculated at $D_F = \ln(3)/\ln(2) \approx 1.585$. It is worth mentioning that the diagonal space-time pattern of that rule is identical to the elementary (1D) cellular automata (**R154**), considering Wolfram's notation [Wolfram(2000)].
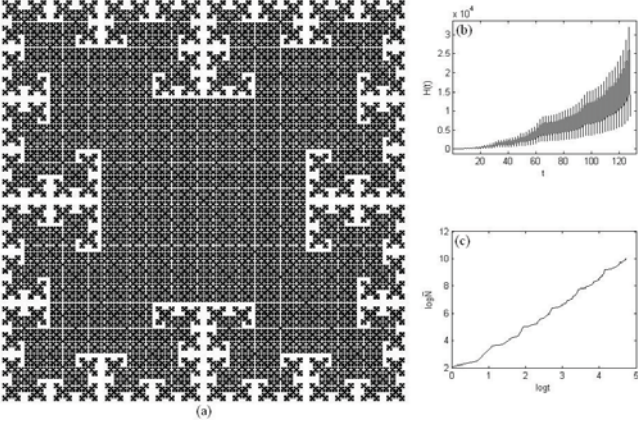
Fig. 6. Rule $\mathcal{F}_{(0,1.65,1.0834,1)}$. (a) Evolution on a 560×560 lattice. Although the interior appears uniform structure, it's boundary shapes dendritic curve.(b) The Hamming distance of the rule as it results from two, almost similar, initial configurations. Here $H(t) \propto t^{1.95}$ which indicates high sensitivity to initial conditions.(c) The logarithmic plot of eq. (10) reveals the model's scaling law between boundary cells and time. The slope gives an estimation of the growth dimension $D_g \approx 1.8$.
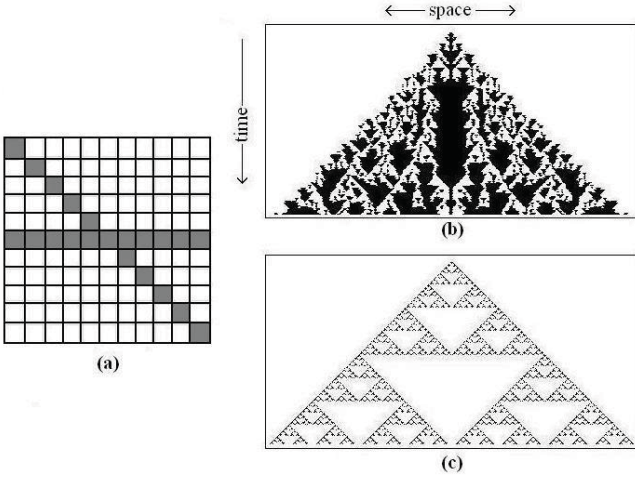

Fig. 7. (See text for details.) (a) The space time pattern method to identify complexity in 2D cellular automata. As the system process its dynamics on the lattice a usefull technique is to isolate a line of cells and watch their progress separately [Wolfram(2000)]. A reasonable choice is this of the center horizontal or the diagonal line.(b) A space-time section of $\mathcal{F}_{(0,1.61,0,1)}$ shows the evolution of the center horizontal line of sites. The generated pattern asymptotically converges to a self-similar object.(c) Diagonal space-time pattern of $\mathcal{F}_{(0.5,0.4,1,0.22)}$. The system results in a familiar fractal.

III. Evolution from Random Seeds

A random seed configuration results when each site $\overrightarrow{x} \in \mathcal{L}$ chooses to be black or white with probability $p = 1/2$. This is what is refered as *completely disordered* global state. Such disordered configurations are typical members of the set of all possible configurations. Patterns generated from them are thus typical of those obtained with any initial state. The presence of structure in these patterns is an indication of self-organization on the lattice [Wolfram(2000)]. This type of evolution, though very important, is beyond the scopes of this work. However,

for consistency reasons, we will report a few preliminary results coming out of some superficial explorations. Qualitatively speaking, three types of collective behaviors have been identified. The first one is this when the system sharply converges to a static equilibrium (Fig. (8a)). The second is when, after a transient state, the system converges to a periodic attractor, where the period appears to (perhaps sensitively) depend on the initial conditions (Fig. (8b)). The third is when the system, after thousands of iterations, still exhibits aperiodic behavior (Fig. (8c)). This classification is pretty resemblant to that in §IIB1. A visual inspection of Figs. 8a-c is enough for someone to observe the striking similarities of these patterns to those generated from the probabilistic *Ising* spin model [Ising(1925)], [Kindermann and Snell(1980)]. In any case, further research is, required, especially from a global point of view.
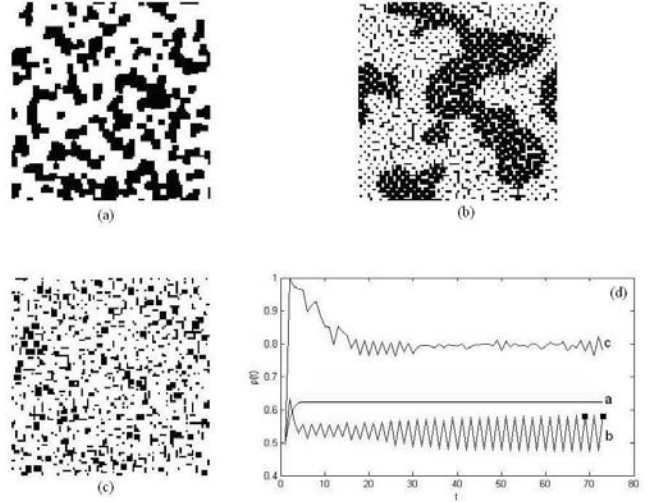
Fig. 8. Patterns generated from random initial conditions. (a)$\mathcal{F}_{(1,0,0,1)}$: The system rapidly reaches an equilibrium state of black and white neighborhoods (see subfig. (d) index $\rho(t)$ curve (a)). (b)$\mathcal{F}_{(0,1,1,0)}$: The system generates large amplitude oscillatory behaviors. After a transient mode the system converges to periodic attractors (here period-4, see black dots in subfig.(d)). (c)$\mathcal{F}_{(0.5,0.4,1,0.22)}$: The system behaves aperiodically. Weak oscillatiory behavior leads to aperiodic evolution.(d) Index $\rho(t)$. It is defined as the instant ratio of white cells to the total number of sites.

IV. Discussion and Concluding Remarks

Up to this point, an extensive effort to explore a proposed mathematical model is made. In this section, we will make suggestions for further research on this model. Furtheremore, we shall propose types of variations which we believe are of special interest. In the end, we will outline fields on which such model may be applicable.

A. Where do we go from here?

In this preliminary paper, our purpose was to study a new update rule in cellular automata and to shed light upon its perspectives. However, many more things are left to do. One line of research could be on the direction of phase transition diagrams which we outlined above. Another issue is this

of dynamic complexity. The Hamming measure used in this paper provides a necessary but not sufficient condition of *chaos*. Nearby trajectories may diverge in time but this does not mean that either this divergence in exponential, or that it is independent of initial configurations. A more elaborate mathematical tool would be the discrete *Green function*. In cellular automata theory, this function is applied to *difference plots*, out of which quantities analogous to *Lyapunov exponents* are, at least heuristically, derived [Wolfram(2000)], [Ilachinski(2002)]. A different perspective that was, briefly, presented above, is the evolution from a random setup. This point of view requires, in principle, an analysis on the base of global state transitions. We must enhance our armory with mathematical equipment from *Statistical Physics* and *Markov Random Fields* [Wolfram(2000)], [Kindermann and Snell(1980)].

B. Variations of the $\mathcal{F}$-rule

Perhaps, the most significant advantage of this model, is its structural flexibility. Conventional variations of cellular automata are the various definitions of neighborhoods $\mathcal{N}(\vec{x}, r)$ (*Moore* or *Von Neumann* scheme, among others [Ilachinski(2002)]) or the state space topology (e.g. boundary conditions), the dimensionality of the rule ($\mathcal{L}^d$) or perhaps the number of elements in Σ. Apart from typical changes we can also vary our rule in terms of:

Cost function f_1. In this paper we assumed that at the end of every iteration, costs are reset to zero. Namely, we made use of instant costs upon the update decision of cells. A variation with physical meaning would be this of accumulative or discount costs per iteration. Each cell saves its cost values from previous states adding it on the future payoffs. For every iteration, a constant discount factor is multiplied with all past costs before the new cost is added. Preliminary simulations have revealed that the collective behavior of such automata is totally different.

Decision function f_2. We have assumed that the update rule defines the new state of site $\vec{x} : \sigma(\vec{x}; t+1)$ to equal to the state of its neighbor with the maximum cost. An important variation of f_2 would be this in which every $\vec{x} \in \mathcal{L}$ to do the following: It calculates the average cost among it's bad neigbors and the average cost among it's good neighbors. Then $\vec{x}$ decides to step to the state with the highest average cost. Of course, function f_2 could be configured to decide according to the minimum of costs. Such behaviors may also be observed in practice.

C. Applications

This automaton could reliably simulate procedures in many fields of life, some of which are reported below:

Social networks

If we consider $\mathcal{L}$ to be a compact society of people (cells), then white, $\sigma = 1$, state would adjust to a *good* man while the black, $\sigma = 0$, would adjust to the *bad* man. Our lattice is, thus, a collection of concrete neighborhoods which interact, according to a cost function (eq. (4) or Fig. 2(a)), locally and

eventually globally. So let's consider the neighborhood $\mathcal{N}(\vec{x})$ of cells. Suppose that $\vec{x}$ is a good man with s also good neighbors and $(8 - s)$ bad neighbors. Then, for every good $\vec{y} \in \mathcal{N}(\vec{x})$, $\vec{x}$ gains a reward of d units while for every bad, our hero may get a penalty of c units. From this point of view, table cost is a locomotive regulator of interactions of "good" over "bad" and vice versa. Together with this regulator one can define static controllers as fixed regions of sites (perhaps on the boundary) whose value remain unchanged throughout simulation. Such areas would imitate the role of a church or a police station around a neighborhood.

Economic Networks

Modelling market interaction. Economics is a subject of social networks, in which the procedure of learning or imitation and then reply among interacting individuals is fundamendal (a game theory approach can be found here [Cartwright(2007)]). In $\mathcal{F}$-realm, sites would be sellers and buyers. To make this more intriguing one could raise the number of possible states and then separates them in two categories: people who sell certain goods, and of people who buy them. Cost values would form the relative value between goods.

Other applications

We shall briefly report potential applications in physical (statistical mechanics - lattice gas theory) [Kindermann and Snell(1980)], [U. Dieckmann and Metz(2000)] or biological (interaction between malignant and non-malignant cells) networks as well as in computer networks. However in each field further research on this model is required.

D. Conclusions

In this paper, we studied a model in cellular automata and presented our results. We attempted to shed light upon its dynamics by presenting some of it's basic characteristics. Despite the fact that our research focused on a simplified version initiating from a specific type of initial configuration, our model revealed a great degree of spatial and time complex behavior. Finally, we proposed nontrivial variations of the rule that could have significant impact on the evolutionary dynamics and discussed possible applications.

REFERENCES

[Cartwright(2007)] Cartwright, E., 2007, Int. J. Game Theory **36**, 116.
[Gardner(1970)] Gardner, M., 1970, Scientific American , 120.
[Ilachinski(2002)] Ilachinski, A., 2002, *Cellular Automata. A Discreet Universe* (World Scientific, NJ).
[Ising(1925)] Ising, E., 1925, Z. Physik **31**, 235.
[Kindermann and Snell(1980)] Kindermann, J., and J. Snell, 1980, *Markov Random Fields and Their Applications* (AMS, NY).
[Neumann(1948)] Neumann, J. V., 1948, Celebral Mechanisms in Behavior - The Hixon Symposium , 1.
[Nowak and May(1992)] Nowak, A., and R. May, 1992, Nature **359**, 826.
[Packard(1985)] Packard, N., 1985, Institute for Advanced Study (preprint).
[Turing(1952)] Turing, A., 1952, Phil. Trans. R. Soc. London **237**(641), 37.
[U. Dieckmann and Metz(2000)] U. Dieckmann, R. L., and J. Metz, 2000, *The Geometry of Ecological Interactions: Simplifying Spatial Complexity* (Cambridge University Press, UK).
[Wiederien and Udwadia(2004)] Wiederien, R., and F. Udwadia, 2004, Int. J. Bifurcation and Chaos **14**(8), 2555.
[Wolfram(2000)] Wolfram, S., 2000, *Automata and Complexity. Collected Papers* (Addison-Wesley).

Simulation of Intelligent Systems

QUESTION-ANSWER MODELS OF DECISION-MAKING TASKS IN AUTOMATED DESIGNING

Petr I. Sosnin
Department of Computer Science
Ulianovsk State Technical University
Z-432027, Ulianovsk, Russia
E-mail: sosnin@ulstu.ru

KEYWORDS

Automated designing, Collaborative work, Decision-making, Modeling, Software engineering.

ABSTRACT

The key problem of the successful development of a software intensive system (SIS) is adequate conceptual interactions of stakeholders at the early stages of designing. Success of development can be increased with using of Artificial Intelligence means including models of reasoning. In this paper a number of question-answer means (QA-means) for decision-making (DM) is suggested. The base of such means is a set of the DM tasks for each of which the QA-model with the typical architecture structure is being built and used. Suggested and implemented means are embedded to the QA-processor adjusted for automated designing the SIS.

INRODUCTION

Designing of the Software Intensive Systems too often gives the results which are not corresponding to the planned expectations. The significant number of the SIS developments either are being stopped, or are being exceeded planned time and/or finance, or reach the end in the poorer version (Charette 2005). The degree of success (expressed in percentages of the number of projects, coming to the end according to the initial plans) is extremely low (about 35 %)

The named kind of works is impossible without using the strict technological discipline in the DM acts which are being usually fulfilled collaboratively in the corporate network. It is important to notice that the collaborative decision-making process is a danger source of mistakes caused by misunderstanding and other problems of using the joint intellectual activities. All of these problems decrease the degree of the successfulness of designing the SIS.

Therefore special means are needed to use for rational integrating the intellectual activities of designers in the corporate net during solving the project task including the DM tasks.

It is reasonable to choose useful means for integration from a set of relevant AI means and includes them to the instrument of designing the SIS. First of all the relevance must be estimated in the context of intellectual capabilities such as "consciousness" and "understanding" used for control the consciousness processes.

When we choose or build the relevant AI means we need to remember that the base form of consciousness work revealing is a reasoning and the dialogue (question-answer process in the brain structure) is a nature of reasoning.

One way for modeling collective (collaborative) "consciousness" is to create the question-answer model (QA-model) of collaborative reasoning which is reflecting consciousness process. But for feedback the joint QA-model in any its current state must be open for the interactive "pressure" on the brain of any member of collective "consciousness" if it is useful for solving tasks. In this case individual consciousness will be combined with the interactive "pressure" of the QA-model of collaborative reasoning (collaborative consciousness).

The QA-model of collective consciousness can be used for creating the model of collective understanding which must be useful for checking intellectual activity applied to solving tasks.

This paper presents the QA-model of the DM tasks in automated designing the SIS in the corporate network. Developed question-answer means are included to the technology of designing. They allow the real time integrating of intellectual activities, which can increase the degree of success in the DM and not only in developing the SIS.

RELATIVE WORKS

The problem of rational reasoning into the development process of the SIS is well known. More then 10 years this problem is investigated in the Software Engineering Institute (SEI) of Carnegie Mellon University (Bass and Merson 2005). But the question-answer approach is not used and the problem of "a real time integration of intellectual efforts" is not indicated in interests of the SEI to the schemes of reasoning and their formalizing.

Artificial intelligence means don't use for supporting reasoning of developers in such well-known technology as Rational Unified Process (Kroll and Kruchten 2003) and in other similar technologies, for example, in Microsoft Solution Framework and Eclipse.

It is very interesting because there are many types of reasoning which are investigated and modeled in AI.

For example, the Programs of the European Conferences on AI (ECAI) include about 20 topics connected with modeling reasoning (analogical reasoning, case-based reasoning, common-sense reasoning, reasoning about actions and others).

We have answer to the question "Why AI means don't use in technologies for developing the SIS?"

Adequate AI means which can increase the successfulness of designing the SIS are absent till now because problem-solving and decision-making based on the real time integration of intellectual resources are investigated in AI only partially (different kinds of models for reasoning which are useful in definite classes of design situations, first of all case-based reasoning models).

We are convinced, that investigation of question-answer reasoning is a perspective way for finding the AI means which can give the positive results helping to solve the successfulness problem of designing the SIS.

In the number of relative works using "questions and answers" (or QA), for example, we can mention reasoning in the "inquiry cycle" (Potts et al. 1994) for working with requirements, "inquiry wheel" (Reiff et al. 2002) for scientific decisions and "inquiry map" (Rosen 2008) for education. Similar ideas are used in the special question-answer system which supports development of the SIS (Henninger 2003). The typical schemes of reasoning for the SIS development are presented in (Yang et al. 2003), in paper (Rich and Feldman 1992) reasoning is presented on seven levels of its application together with used knowledge and in (Lee 2000) model-based reasoning is presented as useful means for software engineering.

But in all publications referenced above, issue (Burger et al. 2001) and special report (Hirschman and Gaizauskas 2001) the task of real time integration of the intellectual resources in processes of problem-solving and decision-making is not mentioned.

QUESTION-ANSWER MODEL OF THE TASK

In developing the SIS for each problem of any degree of complexity the concrete developer is appointed for its decision. His colleagues are entered to the solving process in that case when the intellectual help is required for the developer. Externally the general intellectual activity of the developer and its colleagues is observed in the form of their reasoning which can be registered by the useful way.

We suggest use the QA-model (named above) for simulating collaborative reasoning in any state of the decision process for operative including the such kind of models into the intellectual activity of the developer appointed to the task. The QA-model which reflects collaborative reasoning in the frame of the definite task Z(t) we shall name as the QA-model of this task. The QA-model of the task is a model of collaborative reasoning (and integrated consciousness) in the real time process of solving the task.

The QA-model is a systematized representation of reasoning used during the decision of the task Z(t) and kept in the special QA-database. Any QA-model is a set of interactive objects such as "question", "answer" and "task" with the certain attributes and operations.

Therefore specifications of the QA-models will be presented from the interactive system viewpoint or another words as specification of a specialized software intensive system SIS^{QA}. Such position gives the possibility to use the experience of the SIS to the SIS^{QA} first of all the experience of the architecture description. We defined and investigated the QA-model of the task which is architecturally presented on the Figure 1.
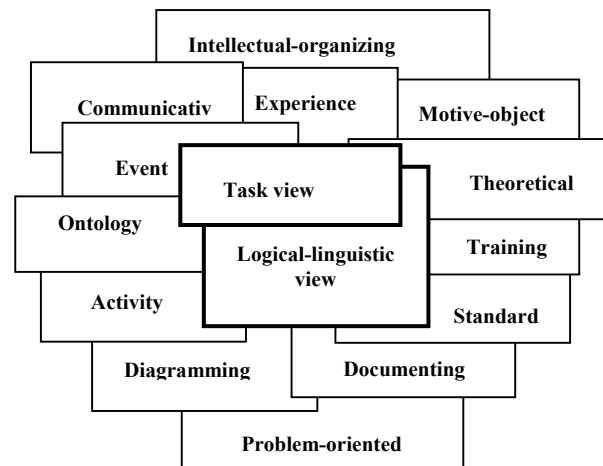
Figutre 1. Architectural Description of QA-Model

The structure and content of the QA-model are being described as a system of architectural views which includes following main views:

1. Logical-linguistic view presents QA(Z(t)) within the frames of logic and linguistics of questions and answers. The visual representation of the view (Figure 2) includes a system of QA-protocols corresponding to the task tree of Z(t). In general case any task Z can include subordinate tasks.

Figure 2. System of QA-Protocols

The logical part of the view describes the hierarchical tree of "questions" and "answers". Each "question" or "answer" is an interactive object with the unique name and rich visual presentation. The QA-tree as a whole consists of the main QA-tree (corresponding to task $Z(t)$) and a set of subordinated QA-trees for subordinated tasks.

The linguistic part of the view includes a system of texts presented the information content of questions and answers. The text of each unit of the view is a result of QA-formalization which is applied to the definite volume of reasoning at the natural language. Such result can be interpreted as a translation of reasoning from the natural language to the QA-language.

Dynamics of the view reflects the history of reasoning registered step-by-step as the history of each unit of the view. Therefore the logical-linguistic view was named as the QA-protocol.

2. Task view opens $QA(Z(t))$ as an interactive task tree (Figure 3) including the interactive model of $Z(t)$ with models of all subordinated tasks.
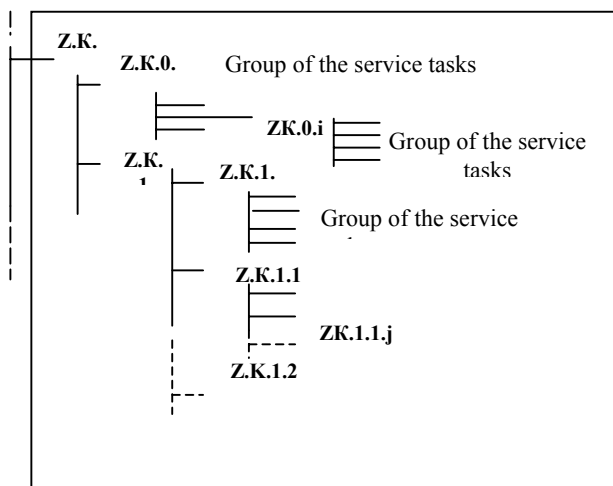
Figure 3. Fragment of the Task Tree for Z=Z.K: K-index name, symbol "0" indicates service tasks, other names indicate the project tasks

Other views are based on the logic-linguistic and task views describing them with different and important point of views. Some of these views will be opened below.

In a basis of integration of views presented above, we shall put the usage of means for modeling of QA-reasoning which reflects the processes of integrated consciousness and integrated understanding at the solution of any tasks.

In integration of views we shall distinguish two interconnected components:

- declarative-visual integration, within the frame of which all artifacts of the QA-model are presented and interactively visualized ;
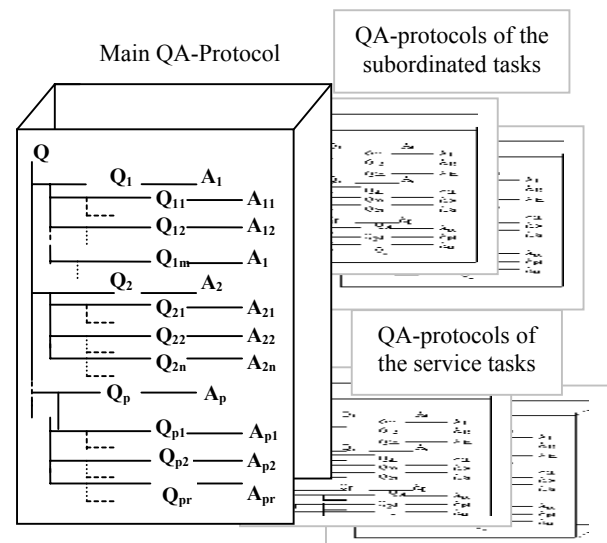- procedural integration as the techniques system for QA-modeling.

Declarative-visual integration is responsible for coding, structuring, ordering, storage and delivery by inquiries of those artifacts which are parts of the QA-model. The function of integration of such type we shall assign to the question-answer database of the SIS project. Such solution allows save all QA-models at the file-server accessible for any workplace of the corporate network. It opens the possibility to include in the architecture of the SIS^{QA} two styles – repository and client-server styles.

Visualization of artifacts is another problem for decision of which it is rational to use architectural style MVC (Model-View-Controller) widespread in practice of development the SIS. Architectural style MVC is the style oriented specially on the human-computer interaction.

QUESTION-ANSWER MEANS

The QA-model of the task $Z(t)$ and its system of views are defined as the problem-oriented base of the software intensive system SIS^{QA} designated for supporting the design of the SIS. In general case the task $Z(t)$ is a task $Z^*(t)$ solving of which is conceptual designing the SIS. Therefore the SIS^{QA} must be built as a complex of QA-means designated for conceptual designing any SIS with the help of real time creating and using the QA-models of the task $Z^*(t)$ and all its subordinated tasks.

The system of QA-means named as QA-processor WIQA (Working In Question-Answers) has been implemented in several versions. Developments of the two last versions were based on architectural views of the QA-model and usage of repository, MVC, client-server and interpreter architectural styles. Moreover in developing the versions have been used object-oriented, component-oriented and service-oriented architectural paradigms. One of the last versions named as NetWIQA has been programmed on Delphi 6.0 and the second version (named as WIQA.Net) has been created on C# at the platform of Microsoft.NET 2.0.

The system WIQA.Net was developed for its usage as a kernel of the product line each unit of which is an application based on this kernel. Product line includes applications "Conceptual design", "System of documenting", "Decision-making system" and "Training system" adjusted for their usage in the corporate network. The NetWIQA was used as a reach source of assets (Sosnin 2004) for developing the kernel.

Server possibilities of the WIQA.Net are opened for thin clients in the corporate network and for Web-clients via the Web-shell programmed with using means of ASP.NET. The component structure of the WIQA.Net is implemented as the open sytem of plug-ins supported the implementation of all architectural views of the QA-model. It is generally (without means reflected and supported Web-access to the QA-means) presented in the Figure 4.
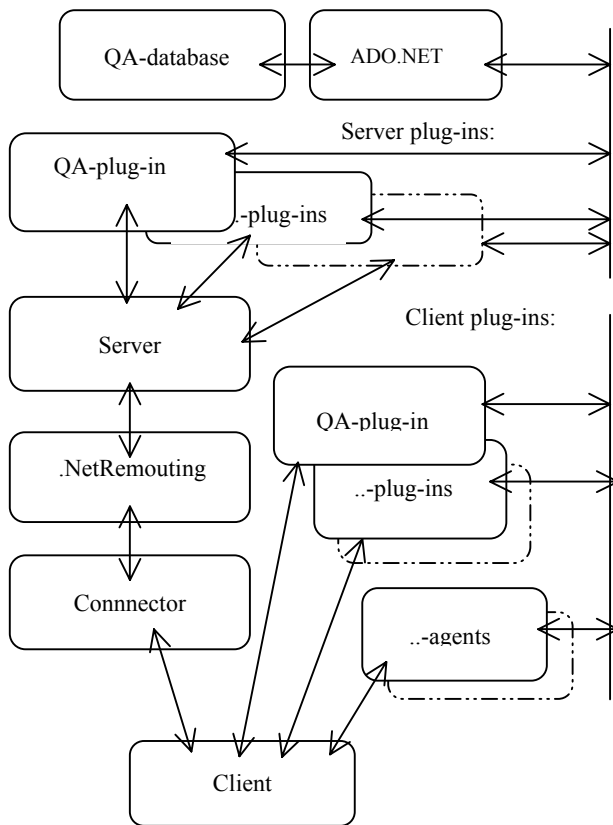
Figure 4. Component Structure of the WIQA.Net

All components are allocated on following tiers:

1. **Database tier** is implemented with using the technology ADO.NET. It is used as a repository of all data.

2. **Server plug-ins tier** encapsulates all function for interaction with the database. The tier is formed dynamically during starting the server with using the technology of a dynamic reflection of types which is supported by a platform .NET. Interfaces of the tier (server plug-ins) contain the functions intended for direct adding, updating and removing of data. These functions are called only by a client tier (client plug-ins).

3. **Server tier** is implemented with using the technology .NET Remouting, that allows address to server from the remote client workplace. When any client workplace is starting the server carries out registration of the client, giving out to it a unique key which is used for checking the access rights to various functions of the system. The server registers also starting of any component of a client plug-ins tier. The server is conducting also logging the most important events of the system (i.e. formation of magazine of events).

4. **Connector tier** (an intermediate tier). Usage of the technology .NET Remouting demands the certainty for all functions of the removed object for access to it. The connector hides the implementation of the server from the client. Therefore the interface of the server is being stored in the connector.

5. **Client tier** is an application with which begins the work a user on the client workplace. This tier is responsible for the program encapsulation of the work with the client plug-ins tier and also for encapsulation of the removed interaction with the server. In case of an allocation of the server and the client on one computer, through a configuration file it is possible to disconnect the usage of the technology .NET Remouting. It will raise the speed of work of the program. The client is also responsible for the reception of the name and the password of the user.

6. **Client plug-ins tier** consists of two parts. One part supports the active interaction with the user and other part is adjusted on the processing of the certain events in the system.

The first part gives the access for the user to the functionalities of the implemented architectural views. Each plug-in of this tier contains a unique key which gives the possibility to distinguish plug-ins of the tier for controlling the access of the user to the appointed plug-ins only. The QA-plug-in of this tier provides the access to commands for the work with the logic-linguistic view (with QA-protocols). Each plug-in of this part of the tier supports the interaction with objects of the corresponding architectural view. The second part of the tier supports the automatic work with a definite set of events during QA-modeling. It provides by the units programmed as agents.

7. **The tier of continuous developing** is activated as a real time access for the developer of additional plug-in (or agent) from the client workplace to the .Net 2.0 means in the context of the current state of the QA-processor. Such additional units are shown in Figure 4 by chain lines.

The Web-access in WIQA.Net is arranged as the one-page site (asp.net) with dynamic additional loading the data, executed on the technology AJAX. At initial loading the user registers the URL address of a resource on which site is placed.

QUESTION-ANSWER MODELING

Question-answer models, as well as any other models, are created "for extraction of answers to the questions enclosed in model". Moreover, the model is a very important form of representation of questions, answers on which are generated during interaction with the model. Questions are fixed in QA-models in the form of "objects-questions" obviously and implicitly in the forms of ambiguities used in textual QA-units.

The essence of QA-modeling is an interaction of stakeholders with artifacts included to the QA-model in their current states. For such interaction the developer can use the special set of commands (QA-commands), their sequences and a set of plug-ins combining with QA-commands. Such work is similar to programming (QA-programming) on the base of means of the special processor (QA-processor) the role of which fulfils the SIS^{QA}.

In order to get the definite positive effect from the concrete QA-model of the definite task the developer will need to program the definite volume of his interaction with this model and executes the created QA-program.

There are a number of expected positive effects for each of which the needed QA-program (or a set of alternative QA-programs) is rationally built. It has given the possibility to create a library of typical QA-programs accessible to the developer in the real time.

The main subset of positive effects of QA-modeling is connected with the real time integrating of intellectual resources and this subset includes:

- controlling and testing current reasoning of the developer with the help of "integrated reasoning" and "integrated understanding" included to QA-models;

- correcting individual understanding with the help of comparing it with "integrated understanding";

- combining models of collective experience with individual experience for increasing the intellectual potential of the designer on the definite working place;

- including individual experience of the developer in accordance with request and through evolving of QA-models in the designing on the other working places in the corporate network..

Any developer can get any programmed positive effect with the help of QA-modeling as "answer" on question actually or potentially included in the QA-model (figuratively shown on Figure 5).
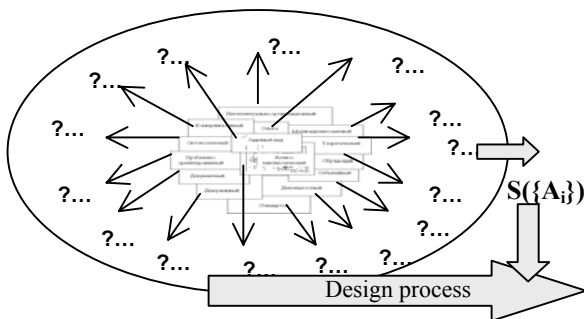

Figure 5. QA-Model as a Source of Answers

As it shown in this scheme any view is a source of answers accessible for the developer as results of his interactions with the QA-model. There are different forms for building answers with the help of QA-modeling, not only linguistic forms.

The developer fulfils the positive interaction in the frames of project, service and technological tasks of the designing process.

All project tasks $Z^P=\{Z^P_r\}$ are derived from the process described above. Any task Z^P_r is a question qualified by stakeholders as a task-question answer of which can be found only through the decision process.

Any service task Z^C_m has its QA-pattern kept in the special library. Such pattern helps to build the model $QA(Z^C_s)$ for the definite conceptual artifact. Service

tasks are defined and implemented for creating documents and visual diagrams, for supporting the typical schemes of communicating and training.

The work with questions, answers and other conceptual artifacts are executed with the help of technological tasks $Z^T=\{Z^T_n\}$ solving of each of which uses a scenario form. The solving of the definite technological task is an executing of the corresponding technique programmed for the QA-processor.

The main application of the QA-processor is a "Conceptual design" the process of which is based on workflows "Interactions with Experience". Each workflow is implemented on the base of the corresponding technological tasks.

To demonstrate the functions of technological tasks let us name a few of them, e.g. "To solve a task", "To answer a question", "To construct the answer", "To prove the answer", "To discuss the answer", "To fill-in the pattern", "To set the requirement", "To determine the front of work", "To plan", "To generate the educational unit", "To control knowledge". The given utilities show that some applications of the QA-processor support also the operative training for the members of a design team.

QUESTION-ANSWER PROGRAMMING

As told above means of the QA-processor gives the possibilities for special kinds of QA-programming and executing the QA-programs.

The first kind is similar to event-driven programming when the state of the QA-model is interpreted as the description of situation used for the next steps of designing. If any question or answer of the QA-model is under constructing then such unit is the object of the potential work. Analyzing the state of the QA-model (more truly "event view" of the model) we can choose its more priority question or answer for continuing or finishing the work with such interactive object ("activity view" of the QA-model).

In any state of any question or answer the developer can activate useful interaction with the chosen object using accessible QA-commands, plug-ins or techniques. Such actions are similar to the work of the QA-processor as "an interpreter".

The second kind is used for simulating the technique steps in the QA-form where "questions" are used for coding steps of the technique and "answers" are used for registering facts of executing steps of the technique. Any step of the technique as a definite volume of the work is being presented as "question" for which is needed to build "the answer".

There are many different types of the QA-commands, plug-ins and techniques used in designing with the help of QA-modeling. Any designer has the possibility for using a set of base techniques for designing, decision-making, communicating, documenting and training.

A set of especially important actions (operations, commands, plug-ins mechanisms, techniques) of the QA processor includes:

1. For questions: detection of obvious questions (on their indicators), predication (through translation on Prolog-like language), identification (on patterns), concrete definition (for types), assignment of meanings to attributes (as to the phenomenon of event type), argumentation of question.

2. For answers: creation, assignment of a type, change of a type, registration of a condition, editing of the contents, assignment of meanings to attributes (as to the phenomenon of event type), argumentation.

3. For QA groups: transformation to the node, expansion into the QA-structure, transformation to the event net, visualization of a network, analysis of a condition, choice of a direction of development, scrolling of dynamics (on inquiries).

4. For text: creation, transformation, grammar analysis, semantic analysis, transformation to the semantic graph, visual analysis, supporting of a phenomenon of attention.

Programs are created for their executing. For QA-programs three version of executing are used. The first version is happen during creation of any part of any QA-program. The first executing of each part of the QA-program is combined with the creating of this part.

The second version is using the "line" of the QA-program for controlling the design process by the declarative or procedural way.

The third version is the visual "pressure" chosen parts of QA-programs on the intellectual activity of the designer. The developer can include several parts from the different QA-program into the request for visualizing. There is a special subsystem in the QA-processor which supports the preliminary preparation of the visualizing information from the QA-database. Any prepared block of questions and answers can be visualized in the suitable temp by switching special windows on the screen or using effects of the 25-th card.

QA-PROGRAMMING IN DECISION-MAKING

There are many different types of DM methods. Choosing methods suitable for designing the SIS we need to take into account the specificity of such kind of activity. Therefore for certainty in QA-programing we are being limited the interactive decision-making, decision-making in group, decision-making by experts and also analysis models of situation and alternatives.

In order to use the possibilities of the QA-processor the typical task is defined for each DM method suitable for designing the SIS. Moreover each typical task is presented as the corresponding technique of the DM coded in the form of the QA-program. The QA-programming of the chosen DM tasks showed that

decision-making by the technique of the expert system has specicfity in implementing and executing.

At first we present the general version of QA-programming and then QA-programming for the expert DM task. For example, the QA-code of the technique for the Plus-Minus-Interesting analysis or PMI-analysis (Harris 1998) is presented on the block (Figure 6):

Question-Answer Template File
"The scheme of PMI management"
Q1. Specifying a problem of decision-making?
Q2. Creating a subtask of decision-making (PMI)?
Q3. Creating a subtask of documenting of decision-making?
Q4. Loading a pattern of the PMI method?
Q5. Identifying of problems and the purposes in a pattern?
Q6. Forming the list and an estimation of pluses?
Q7. Forming the list and an estimation of minuses?
Q8. Forming the list and an estimation of interests?
Q9. Loading of the module of gathering and calculation of estimations?
Q10. Calculation of the general estimation?
Q11. Analysing result and decision-making?
Q12. Creating documents

Figure 6. QA-Model of the Technique

The answers in the template are absent till its using. After loading this template in the task tree it will receive the status of "the QA-model" for the new subordinated PMI task and for each question of this model the answer will be opened. When the technique of the PMI task is executed the symbol "*" is added to the answers for questions which have been fulfilled as points of technique.

Usage of value "*" in answers gives the possibility for breaking the technique in any position and returning to the next point of the technique in suitable time. Moreover it gives the possibility for tracing the way of executing the technique with any executable structure.

Definite documents are used in the DM tasks and in presented PMI task also. For working with documents the QA-processor allows to use the special class of service tasks. Each task of such class also uses the QA-template and the corresponding QA-model. Such possibility is reflected in Figure 1 as "Documenting view".

For the PMI task its base document has the following table structure:

Table 1: Table Structure

Positive		Negative		Interesting	
Plus	Value	Minus	Value	Interest	Value

The document table structure is being transformed to the corresponding QA-protocol the potential fragment of which is presented on Figure 7.

```
……………………………………………
QJ.1. What is the first plus from the decision?
AJ.1.Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx.
QJ.1.1. What importance of the first plus (+1 till +10)?
AJ.1.1.        +6.
QJ.2. What is the second plus from the decision?
AJ.2.Yyyyyyyyyyyyyyyyyyyyyyyyyyyyyy.
QJ.2.1. What importance of the second plus ?
AJ.2.1.        +8.
……………………………………………
QK.1. What is the first minus of decision (-1 till -10)?
AK.1.        -5.
…………………………………………..
```

Figure 7. Fragment of the Estimation Results

Externally simple structure of the both QA-models hides the rich possibilities of their transforming, processing, and visualizing which are caused by the system of their architectural views.

Moreover the PMI task and the corresponding task of its documenting are included to the task tree of designing the SIS and all tasks of this tree have similar QA-models and are being controlled by unified rules.

Different kinds of relations between tasks are used in the task tree of designing the SIS. Relations been possible for the DM tasks are presented on the Figure 8 where tasks of the definite type are marked by labels.
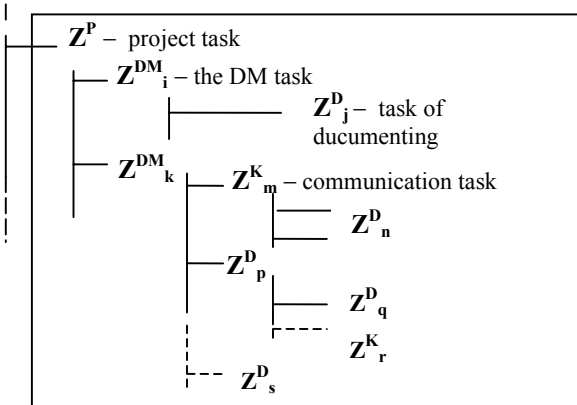

Figure 8. Examples of Relations between Tasks

In accordance with circumstances the developer can open for solving:

- the DM task Z^{DM}_i subordinated to the project task Z^P;
- the task of documenting Z^D_j subordinated to the DM task Z^{DM}_i;
- the communication task Z^K_m subordinated to the DM task Z^{DM}_k;
- the task of documenting Z^D_q subordinated other task of documenting Z^D_p;
- the communication task Z^K_r subordinated to the task of documenting Z^{DM}_p.

The developer can include to the any point of the task tree any task of any type if it is needed for him. The necessity of decision-making can be requested in any

time of designing the SIS. If such event the appropriate method of decision-making as a special task will be loaded in the task tree of designing the SIS.

If it will be useful for the developer he can include to the needed point of the task tree the suitable communication task supported by the QA-processor (special e-mail, the task of collective estimating, the task of argumentation or the brainstorming task). Estimations into the DM tasks are being fulfilled with the help of the communication tasks. For all tasks of communication the QA-templates are created and loaded in the common library of QA-templates. A set of communication tasks supports "the communicative view" of the QA-model.

EXPERT SUBSYSTEM FOR DECISION-MAKING

Into the complex of QA-means for decision-making the expert subsystem is included. Such subsystem is implemented as plug-ins of the QA-processor for supporting "the experience view" of any QA-model.

"The knowledge base" of the expert subsystem gives the possibility to keep precedents of two types – declarative and procedural precedents each of which includes the production rule combined with the QA-model of this rule. The fact of storage together with precedent of its QA-model opens additional (to known) means for checking adequacy of precedent and specification of its version for using the precedent in a situation of a choice. Such QA-model can be interpreted as a DM task positive solution of which confirms the relevance of the corresponding precedent. Moreover a set of keys can be appointed to the result of solving the DM task for potential working with precedents composing other precedents.

The relations between the expert DM task Z^E_j and subordinated expert DM tasks $\{ Z^E_{jk} \}$ in the task tree of designing the SIS are demonstrated on the Figure 9.
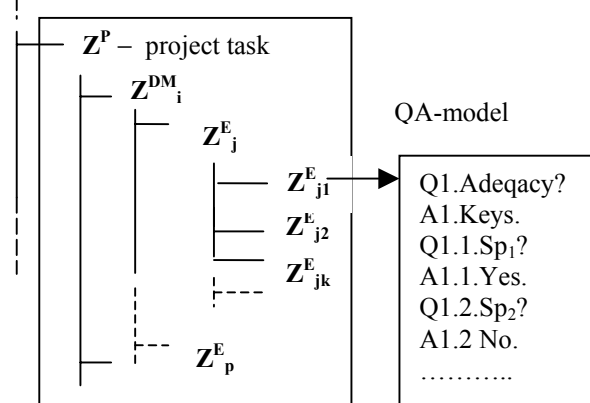

Figure 9. Examples of Relations between Expert Tasks

For working with the expert DM tasks in the WIQA.net following sequence of action can be used:

1. The statement of the project task (for example task Z^P on Figure 9) is transformed to the list of keys for accessing to the knowledge base.

2. The list of keys is used for interacting with the expert subsystem and forming the list of potentially relevant precedents.

3. The expert DM task (for example task Z^E_j) is being created for each potentially relevant precedent and this task is being loaded in the task tree.

4. If the task Z^E_j has a compound type its subordinated task are being loaded in the task tree also.

5. Decision-making the loaded task is being started with subordinated task if they are.

6. If the solution of the subordinated task (for example task Z^E_{j1}) confirms relevancy then keys of this task (answer A1) are being included to the common list of keys.

7. Changing the list of keys is a cause for the next access to the knowledge base.

8. The process of the expert decision-making is being finished when all loaded DM tasks are solved.

The general rules of working with the expert DM task after its loading to the task tree of designing the SIS are fulfilled as working with the others tasks in this tree. The expert DM task can be interrupted in any time with returning to the solution process in the suitable time. The useful service or technjlogical task (any the DM task even) can be subordinated to the expert DM task.

CONCLUSION

This paper presents system of QA-means for decision-making in designing the SIS with the help of QA-modeling. The base unit for QA-modeling is a task of any type which is solving during designing. Each type of the tasks has a specifity which is need to take into account in QA-modeling. For any DM task its technique is known usually. Such specifity gives the possibility to create a set of the QA-programs for the typical DM tasks and loads it to the special library of QA-templates. The designer can use the needed DM-template in any time when it will be useful for designing.
QA-modeling of the DM tasks opens a numer of positive effects such as: real time integrating the intellectual activities of the DM task solvers; effective monitoring of a DM-process; analysis of opportunities of parallel coordination of work in a design team; demonstration (at a suitable speed) of the former steps of the DM; personification of events for subsequent definition of authorship and contribution of the members of a design team.
When the QA-processor is used the DM tasks are included to the task tree of designing the SIS in accordancwe with general rules. Proposed means have confirmed the practical usefulness in development of a number of the SIS, including "The automated control

system for the sea ship" (the expert DM-task on the base of the collision avoided rules).

REFERENCES

Bass L.; Ivers J.; Klein M. and P. Merson. 2005. "Reasoning Frameworks," Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA, Tech. Rep. CMU/SEI-2005-TR-007.

Charette R.N. 2005. "Why software falls", IEEE Spectrum, vol. 42, #9, 36-43.

Burger J. et al. 2001. "Issues, Tasks and Program Structures to Roadmap Research in Question & Answering (Q&A)," Tech. Rep. NIST.

Harris R. 1998 "Introduction to Decision Making" available at http://www.virtualsalt.com/crebook5.htm

Henninger S. 2003. "Tool Support for Experience-Based Software Development Methodologies," Advances in Computers, vol. 59, 29-82.

Hirschman L.and R. Gaizauskas. 2001. "Natural Language Question Answering: The View from Here".Natural Language Engineering, vol. 7, 67-87.

Kroll P. and Ph. Kruchten. 2003. "The Rational Unified Process Made Easy: A Practitioners Guide to the RUP," Addison-Wesley.

Lee M.H. 2000. "Model-Based Reasoning: A Principled Approach for Software Engineering", Software - Concepts and Tools,vol.19, #4, 179-189.

Potts C.; Takahashi K. and A. Anton. 1994. "Inquiry-based Requirements Analysis, " IEEE Software, vol.11, #2, 21-32.

Reiff R.; Harwood W. and T. Phillipson. 2002. "A Scientific Method Based Upon Research Scientists' Conceptions of Scientific Inquiry," In Proc.2002 Annual International Conference of the Association for the Education of Teachers in Science, 546–556.

Rich C. and Y. Feldman. 1992. "Seven Layers of Knowledge Representation and Reasoning in Support of Software Development," IEEE Transactions on Software Engineering, vol, 8, # 6,.451-469.

Rosen D. J. 2008. "How to Make Inquiry Maps". Available : http://alri.org/pubs/im3.html

Sosnin P. 2004. "Question-Answer Processor for Cooperative Work in Human-Computer Environment", In Proc. the 2-d International IEEE conference Intelligent System, Bulgaria, 452-456.

Yang F.; Shen R. and P. Han. , 2003. "Adaptive Question and Answering Engine Base on Case Based and Reasoning Technology," Journal of Computer Engineering, vol.29, #11, 27-28.

PETER SOSNIN was born in Ulianovsk in the USSR, on July 12, 1945. He graduated from the Ulianovsk Polytechnic Institute (1968).
His employment experience included the Ulianovsk Polytechnic Institute and Ulianovsk State Technical University. His special fields of interest includes AI applications for computer aided design. P. Sosnin defended doctor degree in Moscow Aviation Institute (1994). He is an author of eight books and more one hundred articles.

MODELLING, SIMULATION AND BALANCE CONTROL OF A TWO-WHEELED ROBOTIC MACHINE WITH STATIC VARIATION IN LOAD POSITION

Khaled M K Goher and M O Tokhi
Department of Automatic Control and Systems Engineering
The University of Sheffield
Mappin Street, Sheffield, S1 3JD,
United Kingdom.
E-mail: k.mourad@shef.ac.uk

KEYWORDS

Wheeled robot, inverted pendulum, balance mode, PD control. Mathematical Modelling, Simulation.

ABSTRACT

This paper focuses on the design and implementation of a control algorithm for balancing of a *two-wheeled robotic machine* (TWRM). The machine comprises a rod on an axle incorporating two wheels. The balancing of the rod in the upright position is achieved using a proportional-derivative (PD) controller. The Newton-Euler dynamic formulation is used to develop a mathematical model of the system. Matlab Simulink is used as a simulation environment. The intermediate body (IB) is considered to be in the upright position. An external disturbance force is applied at different locations of the IB in order to test the robustness of the developed controller. A payload is considered, and this is attached at different locations along the rod. Investigations are carried out on the effect of changing the duration and location of disturbance force, and changing the location of the payload on the system in the balancing mode. Two PD controllers are developed, one for the angular position of the rod and one for the cart linear displacement. Simulation results are presented assessing the performances of the controllers.

INTRODUCTION

An inverted pendulum system is an under-actuated mechanical system and inherently open loop unstable with highly non-linear dynamics. It is thus a perfect test-bed for the design of a wide range of classical and contemporary control techniques. It has wide ranging applications from robotics to space rocket guidance systems. The concept of balancing a robot is based on the inverted pendulum model. This model has been widely used by researches worldwide in the design and control of wheeled legged robots, etc. (Formal and Martynenko 2005). The inverted pendulum problem is common in the field of control engineering. The uniqueness and wide application of technology derived from this unstable system has drawn interest of many researches and robotics enthusiasts around the world. In recent years, researchers have applied the idea of a mobile inverted pendulum model to various problems, such as designing walking gaits for humanoid robots, robotic wheelchairs and personal transport systems (Kim, *et al.* 2005). The type of intelligent robot proposed here is a mobile robot with a two wheeled inverted pendulum. This design was chosen because its mechanism has an inherently clumsy motion for stabilizing the robot's body posture. The robot has a body with two wheels for moving in a plane and a head similar to a human head for controlling the motion. Two independent driving wheels are used for position control and for fast motion in a plane without casters.

TWO-WHEELED INVERTED PENDULUM ROBOT

Several kinds of wheels can be attached to the wheeled mobile robot, but they fall into one of two categories: driving wheels and auxiliary wheels. The driving wheels are rotated to permit the robot to move when torque is applied to the axles. The auxiliary wheels merely ease the movement of the robot and enable its body to be suspended when no torque is applied to the axles. In most wheeled mobile robots at least one wheel is an auxiliary wheel. For a two-wheeled mobile robot, one can ask what would happen if the auxiliary wheels were removed altogether instead of being replaced with something different or instead of improving the performance of the wheels. By removing the auxiliary wheels, the number of wheels attached on the robot would be reduced. Moreover, the mechanical characteristics of the robot would be completely altered because no elements could suspend and balance the robot's body except for the driving wheels. That is, the robot would have to move and balance its body with only two driving wheels (Kim, *et al.* 2005). The mechanical structure of a robot with only two driving wheels is similar to an inverted pendulum (D'Andrea and Earl 2005).

MATHEMATICAL MODELLING

The dynamic characterisation of the robot is described in this section by introducing the governing main equations describing the mathematical model of the two-wheeled robotic machine (TWRM). The model is derived based on the Newton-Euler equations of motion. The dynamic model of the whole system consists of two

separate sub-models, namely: the non-linear model of the inverted pendulum, and a linear model of the permanent magnet (PM) direct current (DC) motor activating the cart.

Linear model of the DC motor

The robot is powered by two DC motors, Figure 1, where V_a represents the applied voltage, i the armature current. The mathematical model of the DC motor is derived and then used in the dynamic model of the robot to provide a relationship between the input voltage to the DC motors and the control signal needed to stabilise the robot.
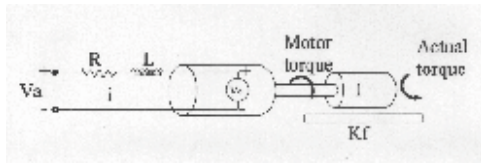

Figure 1 Schematic diagram of a DC motor

Using Figure 1, the following two fundamental equations of motion of the DC motor can be written as:

$$\frac{di}{dt} = \frac{R}{L}i + \frac{k_e}{L}\omega + \frac{V_a}{L} \qquad (1)$$

and

$$\frac{d\omega}{dt} = \frac{k_m}{I_R}i - \frac{k_e}{I_R}\omega - \frac{\tau_a}{I_R} \qquad (2)$$

where R and L are the armature resistance and inductance, V_e represents the winding *emf*, I_R the rotor inertia and K_f the friction constant, ω is the angular velocity of the motor shaft, k_e and k_m represent the back emf Constant and proportionality constant and τ_a is the motor shaft torque. Equations (1) and (2) are used to build the motor model in the *Matlab* environment. The inputs to the model are the applied voltage V_a and the applied torque τ_a. The model outputs are the angular displacement θ and angular velocity ω of the motor shaft. Both equations are first order and linear functions of the armature current and velocity of the motor shaft.

Non linear model of the two-wheeled inverted pendulum

The pendulum and wheel dynamics are analysed separately at the beginning, but this will eventually lead to two equations of motion which completely describe the behaviour of the TWRM.

Consider Figure 2 which represents the free body diagram of the right and left wheels of the robot, Where

H_{fr} and H_r are the friction force between the wheel and the ground and the interaction force at the rod and axle interface respectively for the right wheel.
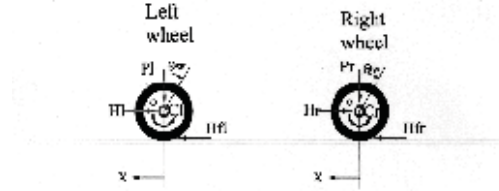
Figure 2 Free body diagrams of robot wheels

The dynamic equation of motion of the wheels can be described as:

$$2\left(M_w + \frac{I_w}{R_w^2} \right)\ddot{x}$$
$$= -\frac{k_m k_e}{RR_w^2}\dot{x} + \frac{k_m}{RR_w}V_a - (H_R + H_L) \qquad (3)$$

Consider Figure 3 which represents the free body diagram of the IB with the external applied disturbance force, F. Applying the Newton's second law of motion in the horizontal direction yields
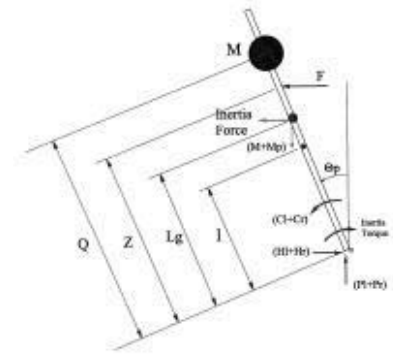
Figure 3 Free body diagrams of the IB

$$\sum F_x = (M + M_p)a_x$$
$$(H_R + H_L) - (M + M_p)L_g\ddot{\theta}_p \cos\theta_p$$
$$+ (M + M_p)L_g\dot{\theta}_p^2 \sin\theta_p - F = (M + M_p)\ddot{x} \qquad (4)$$

Considering the sum of the applied forces in a direction perpendicular to the rod gives

$$\sum F_{\perp p} = (M + M_p)\ddot{x}\cos\theta_p$$
$$(H_R + H_L)\cos\theta_p + (P_R + P_L)\sin\theta_p$$
$$- (M + M_p)(g\sin\theta_p - L_g\ddot{\theta}_p) - F\cos\theta_p$$
$$= (M + M_p)\ddot{x}\cos\theta_p \qquad (5)$$

Equating the sum of the applied moments around the global centre of mass yields

$$\sum M_o = I_g \alpha$$

$$\left(-(H_R + H_L)\cos\theta_p - (P_R + P_L)\sin\theta_p\right)L_g$$
$$-(C_R + C_L) - F\cos\theta_p(Z - L_g) = I_g\ddot{\theta}_p \qquad (6)$$

Manipulating the above equations yields the following two first order non-linear differential equations describing the motion of the system under the effect of an applied payload and impact disturbance force on the rod:

$$\ddot{\theta}_p = \frac{1}{\left(I_g + (M + M_p)L_g^2\right)}\left(\frac{2k_m k_e}{RR_w}\dot{x} - \frac{2k_m}{R}V_a\right.$$

$$\left. -(M + M_p)gL_g\sin\theta_p - (M + M_p)L_g\ddot{x}\cos\theta_p - FZ\cos\theta_p\right) \qquad (7)$$

and

$$\ddot{x} = \frac{1}{\left(2M_w + 2\frac{I_w}{R_w^2} + (M + M_p)\right)}\left(-\frac{k_m k_e}{RR_w^2}\dot{x} + \frac{k_m}{RR_w}V_a\right.$$

$$\left. -(M + M_P)L_g\ddot{\theta}_p\cos\theta_p + (M + M_P)L_g\dot{\theta}_p^2\sin\theta_p - F\right) \qquad (8)$$

Effect of Changing the Payload Location

Let the payload, M be located at a variable distance Q from the IB origin O , as shown in Figure 3, where Q can be expressed as a factor y multiplied by the rod half length l as follows:

$$Q = y\,l \qquad (9)$$

where y can take a value from 0 to 2 according to the position of the payload M from the origin O as follows:

For M to be concentrated at the upper end of the rod $y = 2$,

For M to be concentrated at the mid-span of the rod $y = 1$, and

For M to be concentrated at the lower end of the rod $y = 0$

Changing the payload position will affect both the overall moment of inertia, I_g and the location of the global centre of mass of the IB, L_g. The overall moment of inertia of the IB is affected and modified as:

$$I_g = \frac{M_p(2l)^2}{12} + M_p(L_g - l)^2 + M(Q - L_g)^2 \qquad (10)$$

The location of the global centre of mass of the IB will be affected as:

$$L_g = \frac{(M_p l + QM)}{(M_p + M)} \qquad (11)$$

BALANCE CONTROL USING PD CONTROLLER

At this stage the inverted pendulum rod is considered initially in the upright position. With a disturbance force applied to the rod, it will move from that upright position with an angle (measured from that position) proportional to the magnitude and direction of the disturbance force. In order to keep the whole system controlled, two PD controllers are developed; one for controlling the angular position of the rod for returning the rod back to the upright position after a change caused by the disturbance force, and one for keeping the cart wheels within a specified linear position from a specific reference position. The parameters of the two PD controllers are tuned manually to achieve the desired target at the balancing mode.

PD control parameters for the linear position of the wheels were tuned as $K_{p1} = 6.3$, $K_{D1} = 150$

PD control parameters for the angular position of the rod were tuned as $K_{p2} = 5$, $K_{D2} = 30$

SIMULATION RESULTS

Simulations were carried out on the system using the numerical parameters describing the system features, listed in Tables 1 and 2. The simulations presented to highlight the way the vehicle will behave with the effect of changing the values of three different variables and the control effort required to accomplish the required task. Those variables are the level of the applied disturbance force, the duration of the disturbance force and the position of the payload attached to the vehicle rod. The target was to estimate the limits of those variables beyond which the control algorithm will not be able to balance the system around the upright position. The disturbance force was applied after 10 seconds at the start of the simulation and then after each 90 second interval. This interview was determined as the safe limit for repeating such impact forces to the rod. Below such limit, it was found that the developed controller would not completely keep the IB at a stable position.

Table 1: DC Motor technical properties

| Parameter	Description	Units
$R = 3$	Nominal terminal Resistance	$Ohms$
$L = 0$	Rotor Inductance	H
$K_e = 0.00607$	Back EMF Constant	Vs/rad
$V_e = 0$	Back EMF Voltage	$Vlots$
$I_r = 0$	Rotor Inertia	kgm^2

Table 2: Two-wheeled inverted pendulum properties

| Parameter	Description	Units
$R_w = 0.051$	Wheel Radius	m
$M_w = 2$	Wheel Mass	kg
$M_p = 2$	Pendulum Mass	kg
$M = 50$	Payload	kg
$I_w = 0.000039$	Wheel Inertia	kgm^2
$I_p = 0.0041$	Pendulum inertia	kgm^2
$l = 0.07$	Length to the Pendulum COM	m

Effect of Different Disturbance Levels

The effect of changing the level of the applied disturbance force is considered in this section. Four levels of disturbance amplitudes including zero to test the validity of the developed controller in coping with external disturbances. The simulations are carried out by considering the disturbance force applied at different positions on the IB of the vehicle. Such assumptions may be useful in applications such as extended height of crane arms if subjected to sudden external forces through their stems especially during motion.

Figures 4, 5 and 6 show the system performance with the disturbance force applied at the upper end, mid-span and lower end of the IB respectively.
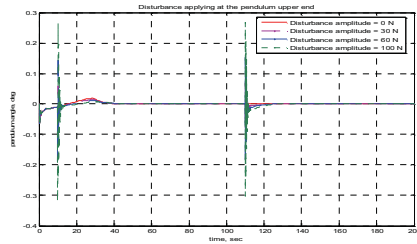
(a) Intermediate body tilt angle
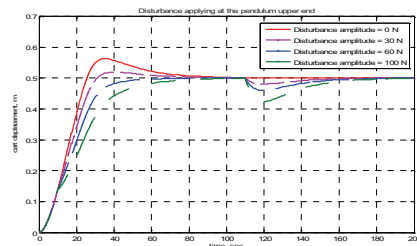
(b) Linear displacement of the vehicle wheels
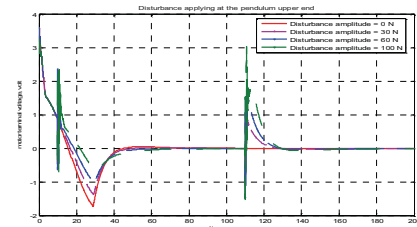
(c) Controller output signal to the system
Figure 4: System performance with disturbance applied at the IB upper end

(a) Intermediate body tilt angle

(b) Linear displacement of the vehicle wheels

(c) Controller output signal to the system

Figure 5: System performance with disturbance applied at the IB mid-span

It can be noted from Figures 4(a), 5(a) and 6(a) that the higher the level of the disturbance force the longer the fluctuation of the rod angle from the upright position. It can also be noted that the closer the location of the applied disturbance force to the IB origin the harder it is to balance the system as it takes the control system a long time to bring the rod to the target position.

The cart linear displacement is presented in Figures 4(b), 5(b) and 6(b). Increasing the level of the applied disturbance tends to decease the cart overshoot beyond the specified limit. That is because the amount of the applied force acts as a drag force for the cart and slows the cart which in turn increases the rise time of the system response. This phenomenon is clearly repeated when the force is reapplied to the system. The lower the level of the disturbance force the faster the cart in achieving the desired position.

Changing the level of the disturbance force also affected the control effort, as clearly presented through Figures 4(c), 5(c) and 6(c). Higher levels of the applied force tended to increase the time the control signal takes to

settle down. The location of applying the disturbance force also greatly affected the control behaviour. The closer the position of the force to the rod origin the more the control effort fluctuation and the higher the control effort at the time of applying the force.
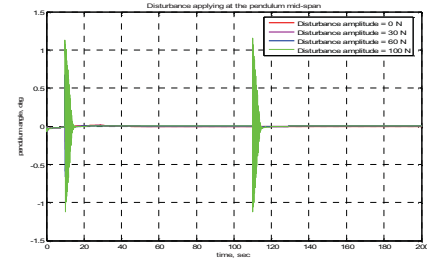

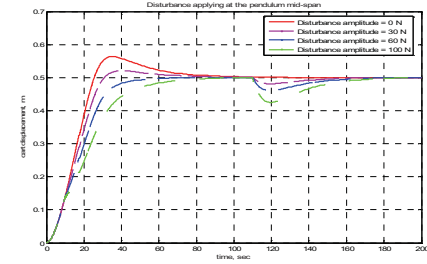(a) Intermediate body tilt angle
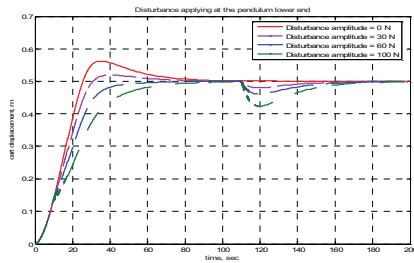
(b) Linear displacement of the vehicle wheels
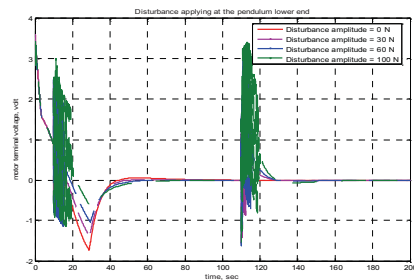
(c) Controller output signal to the system

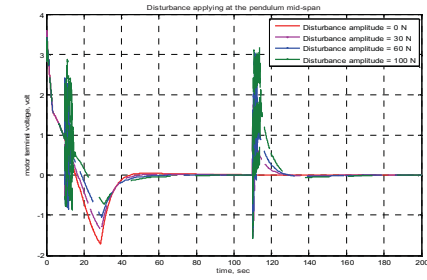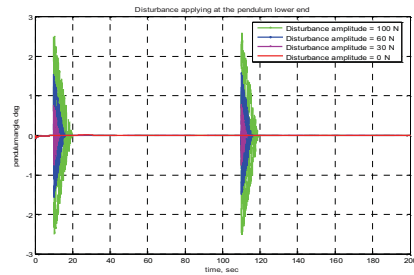Figure 6: System performance with disturbance applied at the IB lower end

Effect of Disturbance Duration

The duration of applying such disturbances is another variable of interest. Four different durations of applying the disturbance force are used to estimate the safe limits beyond which the developed controller will not be able to cope.

Figures 7 and 8 show the IB tilt angle the vehicle displacement, with different duty cycles of a disturbance force of 100 N and 30 N respectively. It is noted in Figure 7 that the amount of overshoot in the IB tilt angle increases as the period of the force is increased, and the controller still copes well with such higher periods of time especially for the pendulum tilt angle. The system can recover faster to the desired position with shorter force duty cycles

For the cart linear displacement, as described in Figure 8, the situation not as good as for the tilt angle; increasing the force duty cycle made the system slower in achieving the target and increased the rise time and decreased the cart overshoot. But for shorter duty cycles, the overshoot tended to be higher which is a characteristic of such impulse disturbance force but accelerated the system behaviour to reach the desired position.
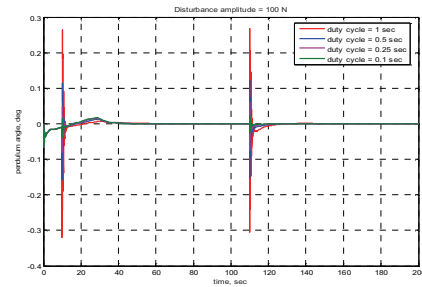
Figure 7 Intermediate body tilt angle
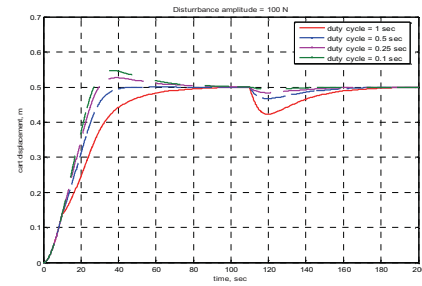
Figure 8 Linear displacement of the vehicle wheels

The control effort is presented in Figure 9. It is noted that the longer the disturbance duty cycle the higher control effort required and the longer the time for the control signal to settle down.
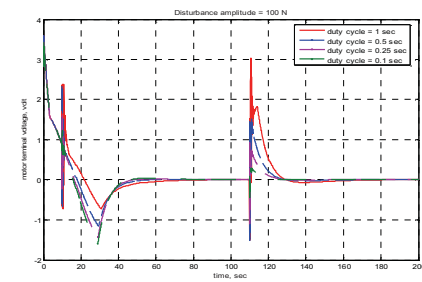
Figure 9 Controller output signal to the system

Effect of Payload Position

The third variable of interest is the location of the payload attached to the IB. The system behaviour including the IB tilt angle and the cart linear displacement are presented and the control effort required for three different locations of the payload. The load is positioned at upper, mid-span and the lower end of the IB. Changing the location of the load mainly affects the location of the global centre of mass and the

whole moment of inertia of the pendulum as presented before.

Two different levels of the disturbance force are implemented here with the assumption of the force at the upper end of the rod. Figures 10 and 11 show the system performance with different load positions and disturbance forces of 100 N and 30 N respectively.
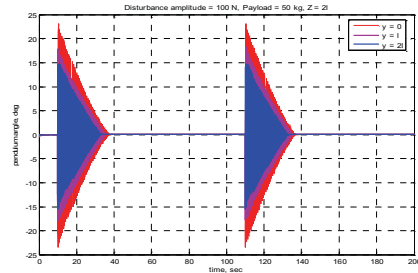
(a) Intermediate body tilt angle
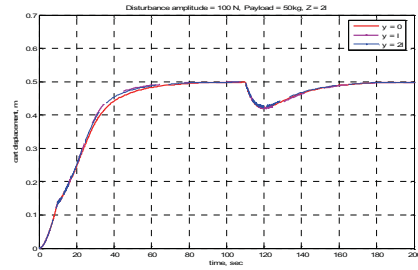
(b) Linear displacement of the vehicle wheels
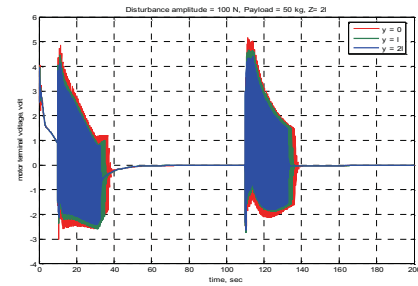
(c) Controller output signal to the system

Figure 10: System performance with load positions and 100 N disturbance force

It is noted from Figures 10(a) and 11(a) that the closer the payload to the IB origin the harder the balancing in the upright position. More fluctuations of the rod occur as the load is closer to the origin. This phenomenon is similar to the system behaviour for the case of applying the disturbance force at the lower end of the rod as previously described in Figures 6. Therefore, this means that the harder it is to balance the IB when either the force is applied at lower positions of the rod or the payload is closer to the origin. Hence, the worst and most hard case of balancing is more likely to happen when the payload and disturbance force are both close to the IB origin.

Changing the payload position in the previous manner did not affect the cart linear displacement much as noted in Figures 10(b) and 11(b). This is not expected to still happen for longer pendulum rod. However, the effect was more due to the value of the disturbance force applied on the rod.

The control effort seems to have more fluctuations when the payload is closer to the origin as noted in Figures 10(c) and 11(c).
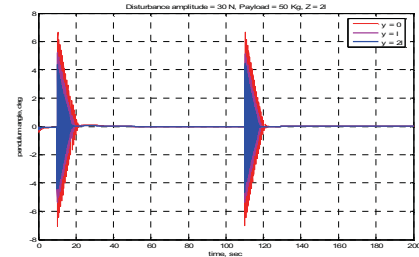
(a) Intermediate body tilt angle
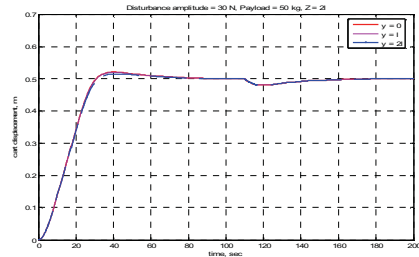
(b) Linear displacement of the vehicle wheels
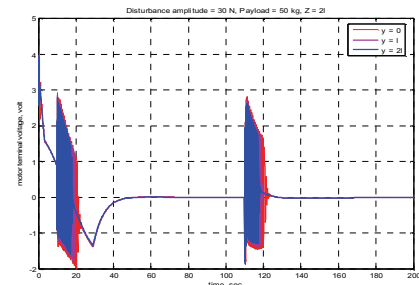
(c) Controller output signal to the system

Figure 11: System performance with load positions and 30 N disturbance force

CONCLUSIONS

The mathematical model of a robotic machine system with two wheels has been developed. An external disturbance force has been considered for application to the IB at different locations, and a payload has been attached to the rod. PD controller has been used to achieve the system performance target. The PD controller has been implemented on the developed model of the two-wheeled robotic machine in the balancing mode. Simulations have been carried out for different cases of changing three variables namely; the disturbance force level, the time duration of the

disturbance force and the position of the payload. The system dynamic behaviour has been presented with control effort for achieving the system performance target in those different cases. It has been noted that the developed control algorithm can cope well in stabilising and balancing the system under different scenarios involving variations in the disturbance force value and duty cycle, and load position.

REFERENCES

D'Andrea R., and M.G. Earl. 2005. "Design and Implementation of a Minimum Time Transition for an Inverted Pendulum". *In Proceedings of the Asian Conference on Industrial Automation and Robotics*, Bangkok, Thailand.

Formal'Skii A.M., and Y.G. Martynenko. 2005. "The Theory of the Control of a Monocycle". *Journal of Applied Mathematics and Mechanics, 69* (4), 516-528.

Goher K.M.K. and M.O. Tokhi. 2008. "Balancing of a Two-Wheeled Robotic Machine with Payload using PD/FL Controllers". *In Proceedings of the 27th IASTED international Conference,* (Innsbruck, Austria, Feb. 11-13), 226-231.

Kim Y., H.K. Kim, and Y. Kwak. 2006. "Improving driving Ability for a Two-Wheeled Inverted Pendulum-Type Autonomous Vehicle". *In Proceedings of IMechE.*

Kim Y., H. K. Kim, and Y. K. Kwak. 2005. "Dynamic Analysis of a Nonholonomic Two-Wheeled Inverted Pendulum Robot". *Journal of Inteligent and Robotic Systems,* 44(1), 25-44.

Randal H.K.C. 2005. "Balancing Wheeled Robot". *Research Project Toward the Degree of the Bachelor of Engineering,* Faculty of Engineering and Surveying, University of Southern Queensland,

Salerno A., J. Angles. 2004. "The Control of Semi-Autonomous Self Balancing Two-Wheeled Quasiholonomic Mobile Robot". *Romanc.*

Snike W.E., M.G.J. Gazdenman, and A. L. Hof. 2005. "The Condition of Dynamic Stability". *Journal of BioMechanics,* 38(1), 1-8.

Stan S., V. Maties, and R. Balan. 2005. "A Solution of the Inverted Pendulum on a Cart Problem Using Predictive Control". *In Proceedings of the IEEE ISIE*, Dubrovnik, Croatia.

Tsai C.C., S.C. Lin, and W.L.K. Luo. 2006a. "Adaptive Steering of a Self-Balancing Two-Wheeled Transporter". *In Proceedings of 2006 Automatic Control Conference,* Tamsui, Taiwan.

Tsai M.C. and J.S. Hu. 2006b. "Robust Control of Auto-Balancing Two-Wheeled Cart Motion Pilot". *In Proceedings of 2006 Automatic Control Conference,* Tamsui, Taiwan.

AUTHOR BIOGRAPHIES

KHALED M K GOHER was born in Egypt. He studied the Mechanical Engineering at El-Minia University where he got his BSc. Degree in 1997. In 2005, he got his MSc. Degree in the same Institution in Dynamic Analysis of Stewart Parallel Manipulators. He worked as a demonstrator and assistant lecturer at the Department of Production Engineering and Mechanical Design, Faculty of Engineering, Suez Canal University, Egypt from 2000 to August 2006. Currently, he is doing his research toward the PhD at the Automatic Control and Systems Engineering Department, the University of Sheffield, UK. His main research interests are in mobile robotics, dynamic systems, intelligent control techniques, assistive technologies and mechatronics applications. *Email address:* *k.mourad@shef.ac.uk*

OSMAN TOKHI obtained his BSc (Electrical Engineering) from Kabul University (Afghanistan) in 1978 and PhD from Heriot-Watt University (UK) in 1988. He has worked as lecturer and Senior Lecturer in Kabul University, Glasgow College of Technology (UK) and the University of Sheffield (UK) and as sound engineer in industry. He is currently a Reader in the Department of Automatic Control and Systems Engineering, The University of Sheffield (UK). His main research interests include active control of noise and vibration, adaptive/intelligent control, soft computing techniques for modelling and control of dynamic systems, high-performance computing for real-time signal processing and control, and biomedical applications of robotics and control. He has extensive publications including textbooks, journal and conference papers in these areas. *Email address:* *o.tokhi@shef.ac.uk* *http://www.sheffield.ac.uk/acse/staff/mot*

SIMULATION OF INTELLIGENT SHIP AUTOPILOTS

Webjørn Rekdalsbakken and Arne Styve
Institute of Technology and Nautical Science
Aalesund University College
N-6025 Aalesund, Norway
E-mail: wr@hials.no

KEYWORDS

Autopilot Simulation, Genetic Algorithm, Distributed Simulation Environment.

ABSTRACT

Due to global conditions there is a demand for more economically and ecologically driven ships. The development of autopilots with optimization criterions on essential parameters will therefore be an important research field in the future. In this effort the use of artificial intelligence (AI) methods will have a central position. Aalesund University College (AUC) has taken the consequence of this need and has in cooperation with relevant industry partners established a research activity on the simulation of intelligent ship autopilots. With the use of adequate and powerful simulation tools this gives the opportunity of real-time testing of adaptive methods for optimization of autopilots based on relevant criterions, such as fuel consumption, amount of waste, and distance covered during a given time span.

INTRODUCTION

Autopilots for ships are growing more and more versatile and complex. Aalesund University College (AUC) lies in the centre of a very vital ship building region, surrounded by companies that are involved in the challenges of safe and economic ship construction and operation. AUC has in cooperation with several of these companies been doing research on modern automation systems on ships, of which autopilots constitute a central field. In addition to traditional autopilots following the course between two points, there are several other needs and challenges, among which economically and ecologically optimal management of ships is a key concept. In the solution of such problems adaptive control systems and artificial intelligence (AI) methods will be vital ingredients of the total control system. To promote R&D activities in the maritime field the enterprise Offshore Simulation Centre (OSC) was established. This firm, which is owned by Rolls Royce Marine Dept., Farstad Shipping, the Norwegian Marine Technology Research Institute (Marintek) and AUC, has a close cooperation with the scientific staff at AUC. The aim of OSC is to develop and operate a complete simulator environment for maritime operations. This is done by building true-scale indoor ship bridges with authentic equipment, and other relevant hardware installations. In combination with mathematical ship and environmental models and

powerful vision systems, this creates a realistic scene for maritime operations. Also a bridge of a high speed craft vessel with a motion system for realistic ship movements is realized inside the laboratories. The backbone of this simulator system consists of a special network and communication framework called the Common Simulation Interface (CSI). By use of this framework a flexible and expandable distributed simulation system can be built. The present work is a small part within the much bigger project of building a complete distributed simulation environment for naval operations. It attempts to illustrate a practical and useful utilization of the simulation environment in the testing of ship autopilots. This is a research field of its own and the simulator environment offers a unique possibility to test all kinds of autopilots against realistic situations in real time. In the field of nonlinear control of marine crafts, including autopilots, dynamic positioning systems, waypoint tracking etc, the research group around Professor Thor I. Fossen at the Norwegian University of Science and Technology, is internationally known and in many ways a reference for much of the maritime simulation activities at AUC (Fossen 2002).

SHIP AUTOPILOTS

The history of ship autopilots has its origin with the development of the gyro compass. The first man to use the gyro compass in automatic steering of a ship was Elmer Sperry in 1911. This autopilot known as "Metal Mike" tried to imitate the naval officer by use of feedback control and automatic rudder adjustments. In 1922 the PID controller was introduced by Nicholas Minorsky. This regulator was developed by observing the officers manoeuvring of the ship and was much used in ship autopilots (Fossen 2002). The main function of the ship autopilot is to control the heading of the ship so that it follows a given direction. The direction can be given as a course in degrees, as a reference point on the map or as a series of reference points constituting a track to follow. To correct the course the autopilot needs to know the instant heading of the ship. Today this can be achieved in a number of ways, with a gyro compass, a Motion Reference Unit (MRU) or a GPS. When following a track of given waypoints it is necessary to continuously measure the distance and direction between the ship and the next reference point. This is done by use of common known formulas (Fossen 1994; Williams 2006; Weisstein 2007).

THE SIMULATION ENVIRONMENT

The Common Simulation Interface (CSI)

The CSI framework is based on the High Level Architecture (HLA) standard and is designed for distributed simulation of complex systems (Husteli 2005). This communication framework is developed by the Norwegian Marine Technology Research Institute (Marintek) in cooperation with Rolls Royce Marine Dept. and is written in Java.[1] Communication on he CSI bus is performed by XML streams over a TCP socket connection, and the framework consists of one server and one or more clients. The server is a multithreaded Java application with one thread serving each client. Each client consists of an application called a *federate*. The federate can be written in any language supporting socket connections. In addition to Java, implementations have been done using C++, C#, as well as Matlab. Each federate exists as an independent process in interaction with all other federates on the CSI bus. A *federate* contains one or more *objects*, which publish their states as *attributes*. The system can be modified or extended with new federates as required. For the system to operate in real time compared to an actual process, one federate must have access to a hardware clock to set an absolute pace of the execution. It is also possible to let a federate execute asynchronously of the other federates (Rekdalsbakken and Styve 2007).

The Control Development Platform

The Control Development Platform (CDP)[2] is a hardware independent software development environment for distributed real-time applications. The functionality of the modules constituting a CDP application is defined by CDP Component Models. A CDP Component Model is defined by the tool CDP Developer and consists of a state machine and all the data included in the module. All the information constituting the properties of the model is saved on XML format in the file *model.xml*. The CDP Component Models are compiled into C++ classes. The models of an application are connected through CDP Connectors, which constitute a framework for message exchange. To run an application the models have to be instantiated. This is done by creating a CDP Component on basis of the definitions in the *model.xml* file. The model configuration is given in a file called *component.xml*, which specifies the communication with other components. The file *application.xml* defines which CDP components the application contains and the chosen hardware platform. On basis of this framework CDP will automatically generate all project files, C++ and XML files for the application. The developer just has to include his specific code, e. g. an autopilot controller. There also exist a number of additional CDP tools to build GUIs, simulate the system behaviour, monitor and acquire process variables, and so on

---

[1] The CSI bus is a proprietary product of Marintek
[2] The CDP system is a proprietary product of ICD AS.

(Industrial Control Design 2006a). This integrated development environment introduces the researcher to the simulator environment by a much lower threshold than traditional programming systems, and considerably cuts the time to prepare and run the models.

System Integration

The communication medium is Ethernet, and the CSI framework allows all kinds of simulation models to be integrated in the simulation environment. The CDP platform is integrated in the CSI system through a CDP component called a CSI client for CDP (Industrial Control Design 2006b). This CSI client is able to create CSI objects and also to support the transfer of messages from the CDP system to federates (and objects) over the CSI bus. Also an UDPASCII server is included in the CDP system to communicate between the CDP system and an external user interface. Several ship- and propulsion models, which are developed by either Marintek or OSC, are available for testing, and new models are easily included. The map system MaxSea is available on an NMEA protocol over TCP interface for plotting of the ship position on basis of information available from the ship simulator. The principle of the simulation environment is shown in Figure 1.

TRADITIONAL CONTROL

With access to the described integrated simulation environment a context exists for the possible simulation of all relevant ship operations. With regard to autopilots both traditional regulators and all kinds of modern control methods may be tested under realistic conditions. The simulator runs in real time compared to the ship operations. This is achieved by including a *federate* with access to a hardware clock and letting this federate control the update frequency of the system. The time interval ΔT is chosen in accordance with the time constants of the simulated process, so as to keep track with the fastest responses. The drawback of this regime is that simulations take a lot of time. It takes time to move a ship and as long as real ship models are used, there is no possibility to speed up the process. In the following some examples of the use of traditional control systems are presented. These simulations are in many cases performed as student projects at AUC. The Figures and examples are taken from such projects, especially from one B.Sc. thesis (Haugen et. al. 2007).

Heading Control

Traditional heading control is usually realized by a cascaded control system with an internal feedback loop from the ship rotational speed. The internal loop will keep the ship rotational speed within limits, and a big overshoot from the reference heading direction will be avoided by a more moderate and contiguously change in the rudder angle. In this way better course following is obtained and the ship motion will be smoother. The principle is shown in Figure 2.
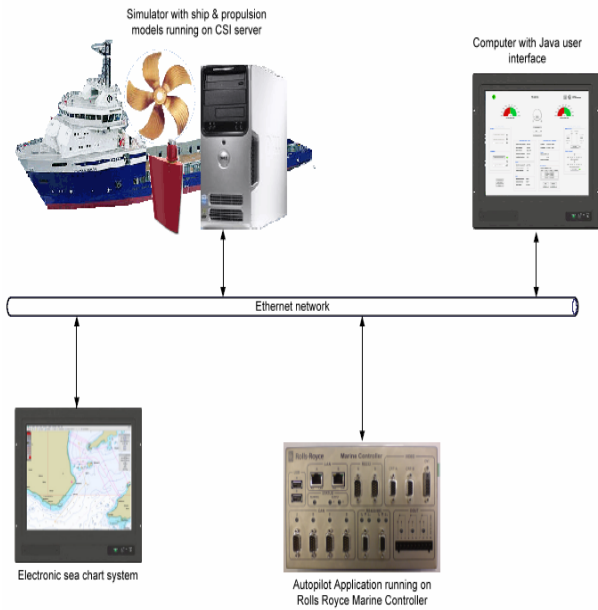
Figure 1: A Sketch of the Simulation Environment
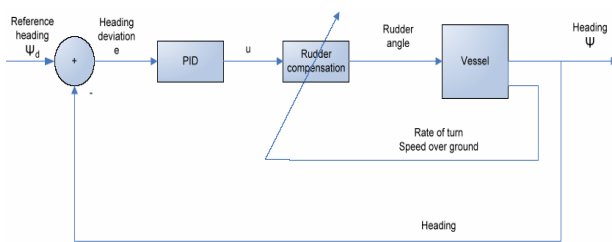
Figure 2: Principle of Heading Control

Waypoint Control

In many situations the ship is controlled to follow a track between given points on the map (waypoints). In this case the ship position must be known at each instant to obtain the distance and direction to the next waypoint (Williams 2006). By use of this information the reference heading is obtained, and traditional heading control can be performed as described above. The principle of waypoint control is shown in Figure 3, and an example of the ship route is shown in Figure 4.
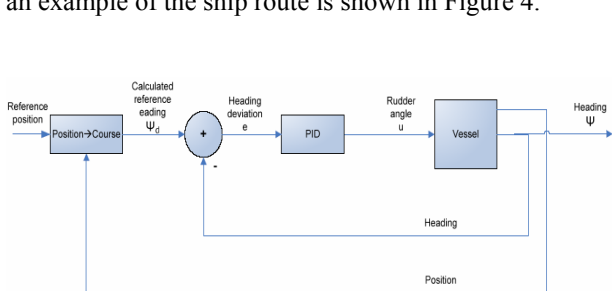
Figure 3: Principle of Waypoint Control
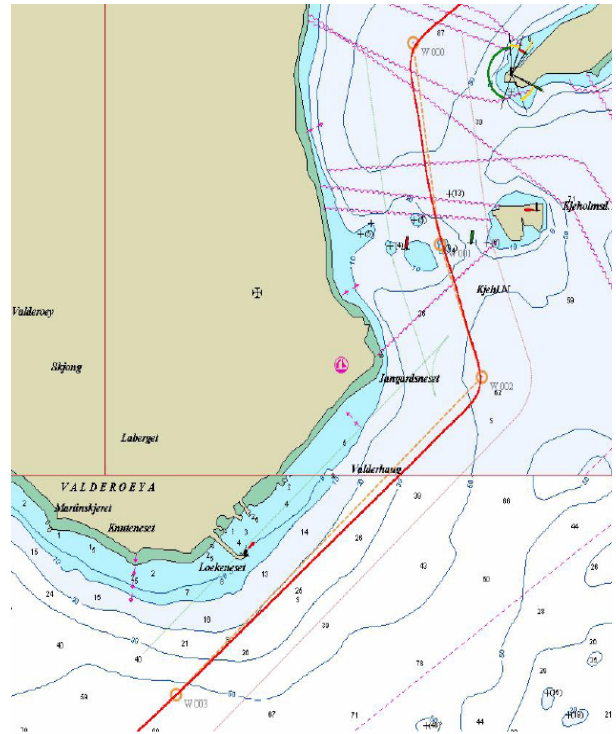
Figure 4: Result of Waypoint Control

Economic Speed Control

A speed control loop for the ship may be useful both as a part of the course control, but also for reasons of economy and comfortable ship movements. Traditional speed control uses all the available power of the ship engine to obtain the correct speed. This is of course not very economical regarding fuel consumption, wear and pollution. It may therefore be beneficial to use an internal loop from the ship angular velocity for the control of the engine power. In addition a feed forward loop from the speed reference is used as a reference for the internal loop. The principle is shown in Figure 5. By limiting the engine rotational speed the power consumption will be reduced, and also the ship movements will be smoother. Figure 6 shows a comparison in power consumption when raising the ship's speed from 7 to 11 knots for traditional and economic speed control.
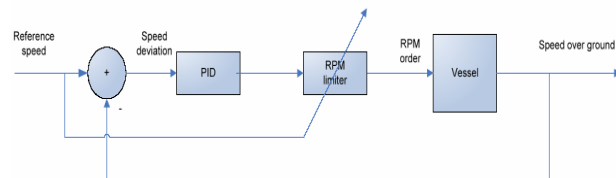
Figure 5: Principle of Economical Speed Control
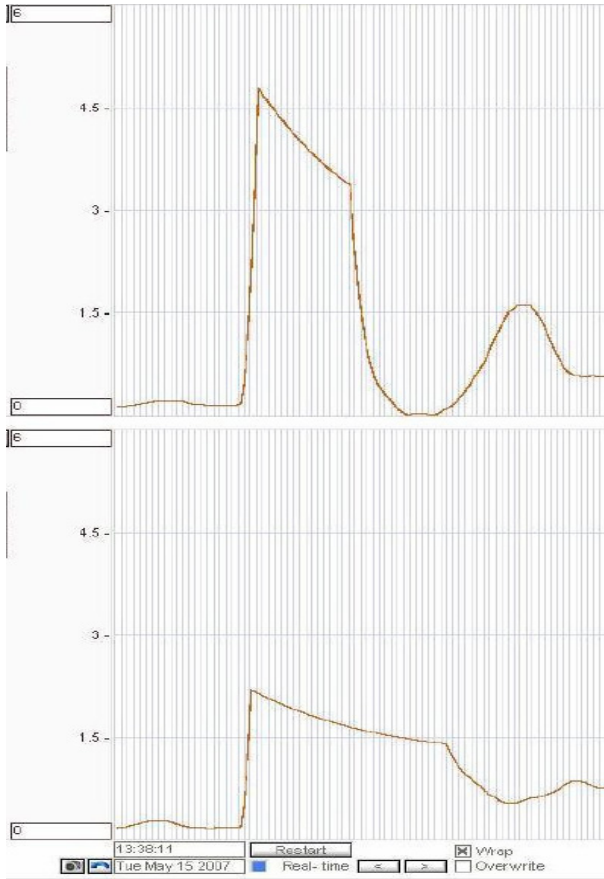
Figure 6: Comparison of Power Consumption with Traditional (top) and Economic (bottom) Speed Control

OPTIMUM CONTROL

The Control Problem

The future control focus will be on the total economy of the operation of the ship, e.g. fuel consumption, waste and pollution, time-limits on delivery, passenger comfort, etc. These problems invite for the use of optimization methods. Relevant solutions will seek the best compromises between several conflicting requirements. In such cases the choice of practical optimization criterions will be very important and have fundamental influence of the control decisions. In this context AI control methods will have a challenging arena, and may offer the best results.

PID versus AI Methods

In the initial simulations traditional PID regulators have been used in cascaded feedback loops from measurements of the relevant system state variables. Tuning of the PID parameters has been accomplished through extended testing of the heading control of the ship under different conditions. The Ziegler-Nichols closed loop method was also employed. Two sets of PID parameters representing the best results from this testing were chosen in the further simulations. On basis of this control situation with multiple cascaded feedback loops the use of AI methods was considered for further

development. The benefits of using a neural network (NN) controller were considered. The NN controller may cope with nonlinearities inherently in the system model, and it is possible to design a single combined control network incorporating all inputs to the system and providing all necessary control outputs. For the use of NN see (Negnevitsky 2002). The NN takes as inputs the relevant system variables; heading, position, and speed of the ship. A network was designed with two intermediate layers to be capable of dealing with both nonlinear and discontinuous characteristics of the process. The outputs of the network are the manipulating signals to the propulsion power and the rudder angle. It was tried to train the network with the back-propagation algorithm against a route of given waypoints, but the training of the NN showed to be more time-consuming than expected. The NN was abandoned and it was decided to explore the use of a GA algorithm for the optimization of the PID parameters (Rekdalsbakken 2007a).

Using a Genetic Algorithm

To obtain the best adaptation of the PID parameters to a real situation it was decided to use a GA, see (Negnevitsky 2002). A very simple chromosome was chosen, consisting only of three genes, representing the PID parameters. A population of 40 chromosomes was created. The initial values of the genes of each chromosome were randomly selected real numbers in the interval 0.0 to 1.0. To be sure that the population included at least one functioning regulator, one of the chromosomes was chosen as the standard PID parameter set from the previous tests. Each chromosome of the population was trained on-line in the simulator against a given regime. First the chromosome was trained to respond to a step in the ship heading of 25° for a time interval of 120 seconds, and then to respond to a step in the heading of 90° for an equivalent time interval. The chromosome fitness value was defined by the following optimization criterion:

$$fitness\ value = \sum_{i=1}^{N} e^2 = \sum_{i=1}^{T/\Delta T} e^2$$

where e is the deviation from the reference course and $N = T/\Delta T$ is the number of cycles in one training epoch. T is the duration of the training epoch, and ΔT is the simulator sampling interval. The training process was run in the simulator for 30 generations which amounted to a total time of 80 hours. The relative fitness for each of the chromosomes of a generation is calculated, and these values are used to select the chromosomes of the next generation (Rekdalsbakken 2007b). Elitism is used in the way that the two best fitted chromosomes of the current generation are always transferred unchanged to the next generation. Among the remaining chromosomes in the population there is a selection according to the tournament method, i.e. a

given number of chromosomes are randomly drawn, and among these the chromosome with the highest fitness value is chosen (Miller and Goldberg 1995). This procedure is repeated until a new chromosome population is formed. This population will now be exposed to a selection for crossover and mutation. The probability of crossover for a randomly selected pair of chromosomes was set to 70%. The probability for a gene of a chromosome to be exposed to a mutation was initially set to 1%. The mutation results in a new random value between 0.0 and 1.0 for the selected gene. The chromosome and the parameters used in the GA are shown in Table 1.

Table 1: Parameters used in the Genetic Algorithm

NAME	VALUE
No. of chrom. in population	40
No. of genes in chromosome	3
No. of variables in each gene	1
Initial value of the genes	random in [0.0,1.0]
Epoch time	240 seconds
Probability for crossover	70 %
Probability for mutations	1 %
Change in mutation	random in [0.0,1.0]

RESULTS

The simulation environment has shown the ability to run tests on all kinds of traditional control loops, and also to offer the possibility to test modern adaptive control strategies. The GA functioned very well in the adaptation of the PID parameters. The GA adapted the parameters of the PID controller very fast to a new reference path. The development of the best fitness value and the mean fitness value over the first 10 generations is shown in Figure 7 and Figure 8. These figures show that the fitness function of the training process settles quite fast to a new and better value. Table 2 shows a comparison between three different PID controllers. The first controller is obtained traditionally by systematic testing of the system, and the second is obtained with Ziegler-Nichols closed loop method. The third controller is the result of the GA optimization of the system. The fitness values for these three controllers were calculated using the procedure described above. Table 2 shows that compared to the standard PID controller the Ziegler-Nichols controller has got an improvement in fitness value of 12%, and the GA obtained controller has got an improvement of 42%. Figures 9 and 10 respectively show the responses for the standard controller and the GA controller to a step in the reference course of the ship of 80 degrees. One can see that the GA controller has a faster response than the traditional controller. However, it also has slightly less damping, which gives more oscillations about the reference track. In Figures 11 and 12 the two controllers are tested against a waypoint path. By inspection one can see that the GA controller follows the reference path (dashed line) much better than the standard controller.
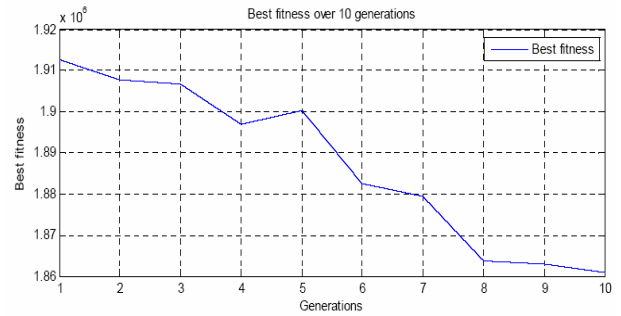
Figure 7: The Development of Best Fitness Value over the first 10 Generations
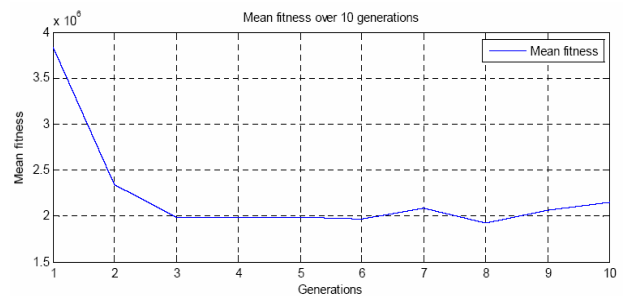
Figure 8: The Development of Mean Fitness Value over the first 10 Generations
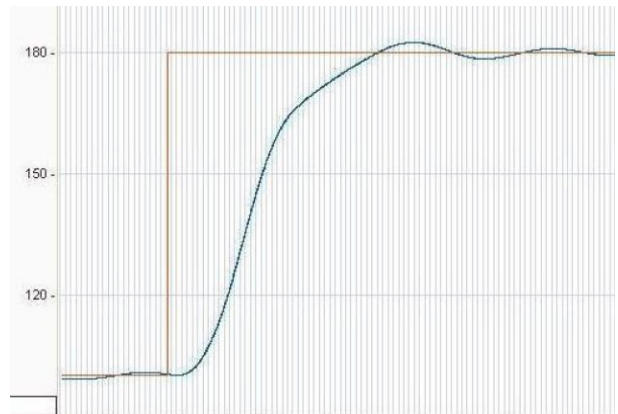
Figure 9: Step Response to a Change in Course for the Standard Controller. Unit on Time Axis is 3 Sec
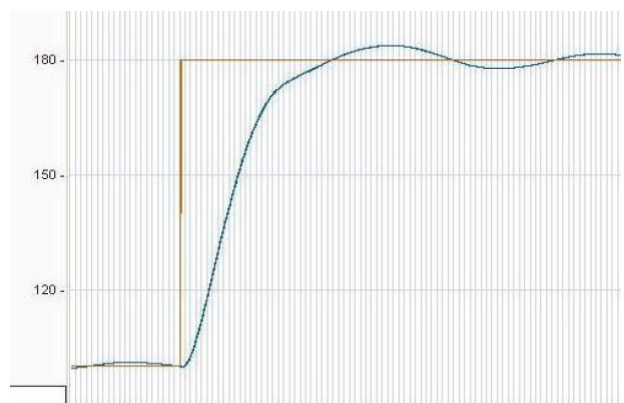
Figure 10: Step Response to a Change in Course for the GA Controller. Unit on Time Axis is 3 Sec

Table 2: Comparison of three different PID Controllers

	K_p	T_i	T_d	Fitness-verdi
Standard	1.000	0.001	1.000	2135859.7
Ziegler-Nichols	0.822	26	6.5	1884500.7
GA-optimalisert	0.473490	0.586760	0.081796	1229567.1
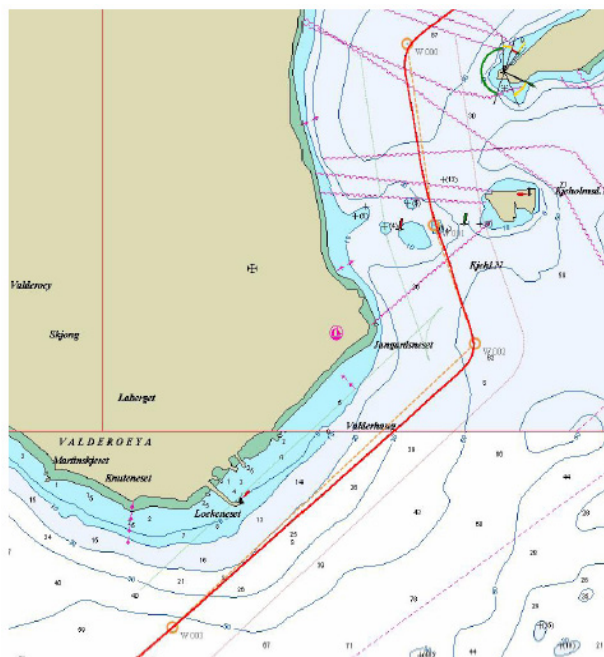
Figure 11: Waypoint Tracking with the Standard Controller
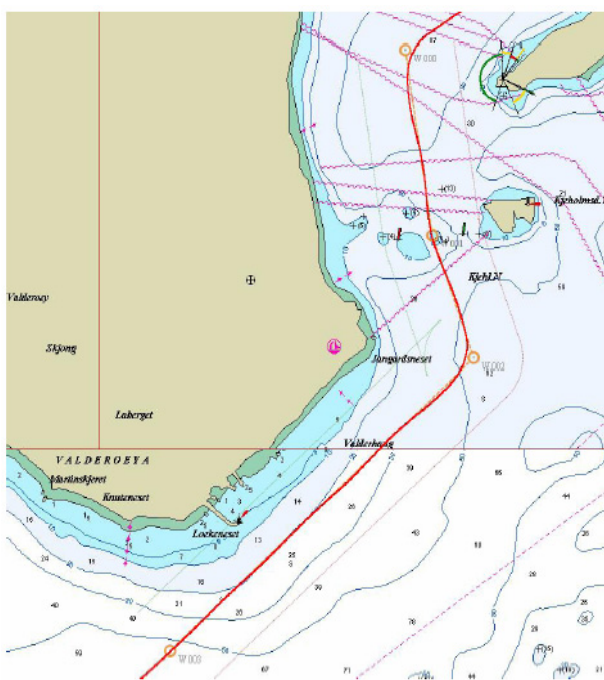
Figure 12: Waypoint Tracking with the GA Controller

CONCLUSION

The presented work is a small example of the utilization of a comprehensive integrated simulation environment. The superior goal is to develop an overall simulation environment capable of realistic simulation all kinds of ship operations. In this context an important aspect is to offer real-time autopilot simulations for all kinds of ships and propulsion systems, so that experiments can be performed to explore the use of adaptive control methods and optimum control strategies. In the development of new control strategies the dominant perspective goes in the direction of optimum utilization of a system of many resources within a frame of restrictions. This is a field very well suited for adaptive control techniques and especially AI methods. This article shows a simple example of such an optimization technique. Based on a traditional PID controller the ship ability to follow a given route is optimized by a GA. The optimization criterion is quite simple, to minimize the deviation from the direct route between points. The GA works very fast; after 6 generations of chromosome selection the GA controller has past the best of the traditional PID regulators. The resulting GA controller proved to be superior compared to traditional PID controllers when tested against a step change in the reference course of the ship, and also in the following of a given waypoint path.

DISCUSSION

The development of the integrated simulation environment is the work of the private company Offshore Simulation Centre (OSC) in cooperation with AUC. This environment gives access to true-scale simulators, using a combination of authentic equipment, mathematical models and vision and motion systems. This is a very extensive activity, and the simulator research program includes many different fields, with maritime operations as the main focus. The program is a long-term enterprise in collaboration between AUC and local industry partners including Rolls Royce Marine Dept. Through these activities AUC can perform realistic simulations of the behaviour of ship operations, including testing of modern control strategies, e. g. advanced AI techniques. This environment offers a unique possibility for building insight into ship behaviour and confidence in ship operations. Also, the understanding of control systems in general will improve. Through this program the scientific staff at AUC has already gained much valuable experience, and hopefully more will come in the near future.

REFERENCES

Haugen M, B. Maroe and E. Rangnes. 2007. "Smart Autopilot". AUC Report, Aalesund, Norway (June).

Husteli N. 2005. "Common Simulation Interface. Documentation and Tutorial". Marintek Report, Trondheim, Norway (Oct.).

Industrial Control Design. 2006a. *CDP Programmer's Manual.* CDP 2.2.0.8 ICD. Aalesund, Norway (Mar.).

Industrial Control Design. 2006b. *CDP System Manual.* CDP 2.2.0.8 ICD. Aalesund, Norway (Mar.).

Miller, B.L. and D.E. Goldberg. 1996. "Genetic Algorithm. Tournament Selection, and the Effects of Noise." In *Journal of Evolutionary Computation*, 4, No. 2, 113-131.

Negnevitsky, M. 2002. *Artificial Intelligence*, *A Guide to Intelligent Systems.* Pearson Education Limited. Addison-Wesley.

Rekdalsbakken, W. and A. Styve. 2007. "Real-Time Process Control with Concurrent Java." In *Proc. of 6th EUROSIM Congress on Modelling and Simulation*, (Ljubljana, Sept. 9-13), 120.

Rekdalsbakken, W. 2007a. "The Use of Artificial Intelligence in Controlling a 6DOF Motion Platform." In *Proc. of ECMS 21th European Conference on Modelling and Simulation*, (Praha, June 3-6), 249-254.

Rekdalsbakken, W. 2007b. "Feedback Control of an Inverted Pendulum by Use of Artificial Intelligence." In *Journal of Advanced Computational Intelligence and Intelligent Informatics,* 11, No. 9 (Nov), 1114 -1121.

Fossen, T.I. 2002. *Marine Control Systems.* Marine Cybernetics, Trondheim, Norway.

Fossen, T.I. 1994. *Guidance and Control of Ocean Vehicles.* John Wiley & Sons Ltd, Chichester, England.

Williams E. 2006. "Aviation Formulary Ver. 1.43". http://williams.best.vwh.net/avform.htm

Weisstein E. 2007. "Great Circle". MathWorld - A Wolfram Web Resource. http://mathworld.wolfram.com/GreatCircle.html

AUTHOR BIBLIOGRAPHY

WEBJØRN REKDALSBAKKEN got his MSc. in Physics at The Norwegian Institute of Technology in 1977. He has been a senior engineer in process control at Hydro Aluminium, Rector at Aalesund Engineering College, and is now leader of the BSc. programme in Automation at Aalesund University College.

ARNE STYVE is MSc. in Computer Science from University of Newcastle upon Tyne in 1991. He is a senior systems engineer at OSC, and Assistant Professor at Aalesund University College.

FLOW SHOP SCHEDULING USING SELF ORGANISING MIGRATION ALGORITHM

Donald Davendra
Ivan Zelinka
Faculty of Applied Informatics
Tomas Bata University in Zlin
Czech Republic
{davendra, zelinka}@fai.utb.cz

KEYWORDS

Flow shop scheduling, self organising migration algorithm, evolutionary heuristics.

ABSTRACT

This paper presents the application of self organinsing migration algorithm (SOMA) to the scheduling problem of flow shop. Flow shop is regarded as the most widely utilized shop management system. Two different benchmark problems are attempted with good results obatined both in comparison with the optimal and other published heuristics.

INTRODUCTION

Advanced manufacturing systems often rely on metaheuristics to solve time constrained scheduling problems. This is largely due to the intractable problems commonly presented in such systems. Flow shop scheduling (FSS) can be considered as one of the common manufacturing problems that is regularly realized using optimization techniques (Onwubolu, 2002).

The evolution of optimization techniques has been mainly attributed to the increase in complexity of problems encountered. Two branches of heuristics exist: constructive and improvement (Onwubolu and Mutingi 1999). Constructive methods are usually problem dependent (Cambell *et al.* 1970, Nawaz *et al.* 1983). Improvement methods are those involving population-based heuristics, which usually follow a naturally occurring paradigm. Some of these are genetic algorithms (GA), tabu search (TS), neural networks (NN), simulated annealing (SA) and particle swamp optimization (PSO) among others.

Self organinsing migration algorithm (SOMA), was presented in Zelinka (2002, 2006), as a novel tool for real optimization problem. This basically implies that SOMA can effectively solve real domain problems involving continuous values.

However, a separate branch of optimization problem exits; namely NP hard problems, of which flow shop is an example, which still presents considerable challenge.

The aim of this paper is to introduce the first *permuatative* SOMA heuristic which is then applied to the flow shop scheduling problem.

The paper is divided in the following sections: the next section introduces flow shop scheduling, the following section to that presents SOMA and its permutative refinements. This is followed by the results and analysis part and finally the work is concluded.

FLOW SHOP SCHEDULING

In many manufacturing and assembly facilities a number of operations have to be done on every job. Often, these operations have to be done on all jobs in the same order, which implies that the jobs have to follow the same route. The machines are assumed to be set up and the environment is referred to as flow shop (Pinedo 1995). The flow shop can be formatted generally by the sequencing on *n* jobs on *m* machines under the precedence condition. The general constraints that are assessed for a flow shop system is the time required to finish all jobs or makespan, minimizing of average flow time, and the maximizing the number of tardy jobs.

The minimization of completion time for a flow shop schedule is equivalent to minimizing the objective function

$$\Im = \sum_{j=1}^{n} C_{m,j} \qquad (1)$$

where
$C_{m,j}$ is the completion time of job *j*. To calculate $C_{m,j}$ the recursive procedure is followed for any i^{th} machine j^{th} job as follows:

$$C_{m,j} = \max\left(C_{i-1,j}, C_{i,j-1}\right) + P_{i,j} \qquad (2)$$

where $C_{i,j} = k$ (any given value) and

$$C_{i,j} = \sum_{k=1}^{j} C_{1,k}; C_{i,j} = \sum_{k=1}^{i} C_{k,1}$$ which represents *i* as the

machine number, j as the job in the sequence and $p_{i,j}$ as the processing time of job j on machine i.

SELF ORGANISING MIGRATING ALGORITHM

SOMA is based on the competitive-cooperative behaviour of intelligent creatures solving a common problem.

In SOMA, individual solutions reside in the optimized model's hyperspace, looking for the best solution. It can be said, that this kind of behaviour of intelligent individuals allows SOMA to realize very successful searches.

Because SOMA uses the philosophy of competition and cooperation, the variants of SOMA are called strategies. They differ in the way how the individuals affect all others. The basic strategy is called 'AllToOne' and consists of the following steps:

1. *Definition of parameters*. Before starting the algorithm, the SOMA parameters (popSize, Dim, PathLength, Step, PRT, Migrations, MinDiv, see Table 1) has to be defined. The user must also create the specimen and the cost function that will be optimized. Cost function is a wrapper for the real model and must return a scalar value, which is used as gauge of the position fitness.
2. *Creating of population*. New population with PopSize individuals is randomly generated.
3. *Migration loop*.
3.1. Each individual is evaluated by cost function and the leader (individual with best fitness) is chosen for the current migration loop.
3.2. For each individual the PRT Vector is created.
3.3. All individuals except the leader perform their run towards the leader according to equation (3). The movement consists of jumps determined by the Step parameter until the individual reaches the final position given by the PathLength parameter. For each step, the cost function for the actual position is evaluated and the best value is saved. Then, the individual returns to the position, where it found the best cost value on its trajectory.
3.4. The new leader is chosen.

If the difference in cost values between leader and the worst individual is lower than value of the MinDiv parameter or the maximum of migration loops has been reached, the run of SOMA is terminated and the best position (the best set of parameters) is returned. In other case the algorithm continues in step 3.

SOMA Parameters

SOMA, like other evolutionary algorithms, is controlled by two groups of parameters, which have to be defined

before running SOMA. The first group is used to stop the search process when one of two criterions is fulfilled, whereas the second group of parameters is responsible for the quality of optimization process results. They are presented in Table 1.

Table 1: SOMA Parameters

SOMA Parameter		
Name	**Range**	**Type**
PopSize	10+	Control
Dim	Problem Specific	Control
PathLength	$(1.1 - 3)$	Control
StepSize	$(0.11 - PathLength)$	Control
PRT	$(0 - 1)$	Control
Migration	10+	Termination
MinDiv	User specified	Termination

Discrete Population

The canonical SOMA creates a random population with implied MinDiv. This is not possible for the permutative population. In the permutative population, discrete solutions are created of size *Dim* to fill the population of size *PopSize*.

Mutation

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 however is applied differently 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. It is defined in the range [0, 1] and is used to create a perturbation vector (PRT Vector) as follows:

$$if\ rnd_j < PRT\ then\ PRTVector_j = 1$$
$$else\ 0\ ,\quad j = 1,\ldots,n_{param} \tag{3}$$

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.

Generating New Candidate Solutions

In standard ES, the *Crossover* operator usually creates new individuals based on information from the previous generation. Geometrically speaking, new positions are selected from an N dimensional hyper-plane. In SOMA,

which is based on the simulation of cooperative behaviour of intelligent beings, sequences of new positions in the *N*-dimensional hyperplane are generated. They can be thought of as a series of new individuals obtained by the special crossover operation. This crossover operation determines the behaviour of SOMA. The movement of an individual is thus given as follows:

$$\vec{r} = \vec{r_0} + \vec{m} \, t \, \vec{PRTVector} \qquad (4)$$

where:

$\vec{r}$:	new candidate solution
$\vec{r_0}$:	original individual
m :	difference between leader and start position of individual
t :	$\in [0$, Path length$]$
PRTVector :	control vector for perturbation

It can be observed from Equation (4) that the PRT vector causes an individual to move toward the leading individual (the one with the best fitness) in *N–k* dimensional space. If all *N* elements of the PRT vector are set to 1, then the search process is carried out in an *N* dimensional hyperplane (i.e. on a *N+1* fitness landscape). If some elements of the PRT vector are set to 0 then the second terms on the right-hand side of Equation (4) equal 0. This means those parameters of an individual that are related to 0 in the PRT vector are not changed during the search. The number of frozen parameters, *k*, is simply the number of dimensions that are not taking part in the actual search process. Therefore, the search process takes place in an *N–k* dimensional subspace.

Solution Conversion

During SOMA mutation and crossover, the discrete values are changed into real values. This is an approximation of the direction of the search space. The solution obtained is simply rounded to the nearest integer using the Discrete Set Handling (DSH) routine.

$$r_i = \text{int}[\vec{r_i}] \qquad (4)$$

All bound offending values are dragged to the offending boundary.

$$r = \begin{cases} Dim \text{ if } r > Dim \\ 1 \text{ if } r < 1 \\ r \text{ otherwise} \end{cases} \qquad (5)$$

Repairment

On occassions, there may evolve solutions which are non-viable, implying that multiple singular values are present in the solution. In order to repair the solution, a random repairment procedure is applied.

This procedure is given as:
1. Check the solution for repeated values, and store

the values and its index in an array.
2. Check the solution for missing values, and store these values in another array.
3. Randomly select a value from the missing value array and then insert it in the position randomly indexed from the repeated solution array.
4. Delete the value from the missing value array and delete the index from the repeated solution array.
5. Iterate until no values remain in the missing value array.

Once the solution is repaired, it is evaluated for the objective function.

EXPERIMENT AND ANALYSIS

Validification of SOMA was done on the benchmark problem sets of FSS. Two different sets were selected from literature and available on the OR Library (2006).

The experimental phase was divided into two parts. The first section delas with the parameter setting for SOMA, and the second section outlines the experimentational results.

Parameter Settings

SOMA utilises a number of parameter setting for operation, alongside its four separate strategies. The strategy All-To-One was primarily selected because of its lower operational time.

The other values obtained through parameter testing is given in Table 2.

Table 2: SOMA Operational Values

Parameters	Values
Migrations	300
Path Length	3.0
Step Size	0.23
PopSize	200

Benchmark problems of Car, Rec and Hel

The first set of benchmark problems to be evaluated were the Car (Carlier 1978), Rec (Reeves 1995) and Hel (Heller 1960) benchmark sets. A total of 31 instances were evaluated. The results are presented in Table 3.

Comparison of SOMA was done with the Improved Genetic Algorithm (IGA) and Multiagent Evolutionary Algorithm (MAEA) of Hu *et. Al.* (2006) and the Hybrid Genetic Algorithm (H-GA) and Othogonal Genetic Algorithm (OGA) of Tseng and Lin (2006).

As shown in Table 3, SOMA was highly competitive with the other heuritics for the defined problems. In summation, SOMA was able to find the optimal solution for all the Car problems, as well as seven

optimal solutions for the Rec problem. In addition, it also obtained the optimal solution for the Hel02 instance. Overall, SOMA was able to get the best solution in 27 instances out of the 31 total instances. It was the most successful heuristic in the Rec instance class, outperforming the other heuritics in problem instances where the optimal soltuion was not obtained by any heuristic (Rec 19-41).

Table 3: SOMA comparison with Car, Rec and Hel Problems.

Name	n x m	Cost	H-GA	OGA	IGA	MAEA	SOMA
Car1	11x5	7038	**0**	**0**	**0**	**0**	**0**
Car2	13x4	7166	**0**	**0**	**0**	**0**	**0**
Car3	12x5	7312	**0**	**0**	**0**	**0**	**0**
Car4	14x4	8003	**0**	**0**	**0**	**0**	**0**
Car5	10x6	7720	**0**	**0**	**0**	**0**	**0**
Car6	8x9	8505	**0**	**0**	**0**	**0**	**0**
Car7	7x7	6590	**0**	**0**	**0**	**0**	**0**
Car8	8x8	8366	**0**	**0**	**0**	**0**	**0**
Rec01	20x5	1247	**0**	0.04	**0**	**0**	**0**
Rec03	20x5	1109	**0**	**0**	**0**	**0**	**0**
Rec05	20x5	1242	0.08	0.21	**0**	**0**	0.002
Rec07	20x10	1566	**0**	0.79	**0**	**0**	0.01
Rec09	20x10	1537	**0**	0.35	**0**	**0**	**0**
Rec11	20x10	1431	**0**	0.91	**0**	**0**	**0**
Rec13	20x15	1930	0.52	1.08	0.62	**0**	**0**
Rec15	20x15	1950	0.92	1.23	0.46	**0**	0.01
Rec17	20x15	1902	1.26	2.08	1.73	**0**	0.02
Rec19	30x10	2093	0.38	1.76	1.09	0.28	**0.02**
Rec21	30x10	2017	0.89	1.64	1.44	0.44	**0.02**
Rec23	30x10	2011	0.45	1.90	0.45	0.44	**0.03**
Rec25	30x15	2513	1.03	2.67	1.63	0.43	**0.03**
Rec27	30x15	2373	1.18	2.09	0.80	0.56	**0.01**
Rec29	30x15	2287	1.05	3.28	1.53	0.78	**0.0**
Rec31	50x10	3045	0.56	1.49	0.49	0.10	**0.04**
Rec33	50x10	3114	**0**	1.87	0.13	**0**	**0**
Rec35	50x10	3277	**0**	**0**	**0**	**0**	**0**
Rec37	75x20	4951	2.54	3.41	2.26	2.72	**0.09**
Rec39	75x20	5087	1.79	2.28	1.14	1.61	**0.06**
Rec41	75x20	4960	2.82	3.43	3.27	2.70	**0.09**
Hel01	100x10	513	-	-	-	0.38	**0.02**
Hel02	20x10	135	-	-	-	0	**0**

Comparison with Taillard Benchmark Problem Sets

The second set of experimentation module is referenced from Thaillard (1993). These sets of problems have been extensively evaluated (see Nowicki *et al.* 1996 and Reeves *et al.* 1998). This benchmark set contains 100 particularly hard instances of 10 different sizes, selected from a large number of randomly generated problems.

The results are tablulated in Table 4 as quality solutions with the percentage relative increase in makespan with respect to the upper bound provided by Thaillard (1993). To be specific the formulation is given as:

$$\Delta_{avg} = \frac{(H - U) \times 100}{U} \qquad (8)$$

where *H* denotes the value of the makespan that is produced by the SOMA algorithm and *U* is the upper bound or the lower bound as computed.

The results obtained are compared with those produced by GA, Particle Swarm Optimization (PSO$_{spv}$) DE (DE$_{spv}$) and DE with local search (DE$_{spv+exchange}$) as in Tasgetiren *et al.* (2004).

Upon analysis of the results, it can be seen that SOMA performs competitvely with the other heuristics. On four instance classes of 20x*XX* and 50x5, it is the best performing heuristic. On three other instances of 50x20, 100x5 and 100x20 it is the second best performing heuritic behind DE$_{spv+exchange}$.

Additionally, on five classes of problems, SOMA obtaines the best average standard deviation. This displays the ability of SOMA to find good solution consistantly.

Table 4: SOMA comparison with Taillard benchmark problem sets.

	GA		PSO$_{spv}$		DE$_{spv}$		DE$_{spv+exchange}$		SOMA	
	Δ_{avg}	Δ_{std}	Δ_{avg}	Δ_{std}	Δ_{avg}	Δ_{std}	Δ_{avg}	Δ_{std}	Δ_{avg}	Δ_{std}
20x5	3.13	1.86	1.71	1.25	2.25	1.37	0.69	0.64	**0.42**	*0.48*
20x10	5.42	1.72	3.28	1.19	3.71	1.24	2.01	0.93	**1.29**	*0.45*
20x20	4.22	1.31	2.84	1.15	3.03	0.98	1.85	0.87	**1.09**	*0.34*
50x5	1.69	0.79	1.15	0.70	0.88	0.52	0.41	0.37	**0.41**	*0.34*
50x10	5.61	1.41	4.83	1.16	4.12	1.10	**2.41**	*0.90*	4.8	1.0
50x20	6.95	1.09	6.68	1.35	5.56	1.22	**3.59**	0.78	3.9	*0.69*
100x5	0.81	0.39	0.59	0.34	0.44	0.29	**0.21**	*0.21*	0.4	0.24
100x10	3.12	0.95	3.26	1.04	2.28	0.75	**1.41**	*0.57*	3.14	1.4
100x20	6.32	0.89	7.19	0.99	6.78	1.12	**3.11**	*0.55*	4.96	0.65
200x10	2.08	0.45	2.47	0.71	1.88	0.69	**1.06**	*0.35*	2.4	1.1

CONCLUSION

SOMA has proven to be highly successful in solving the flow shop problem. Two classes of problems were attempted and both the results validated the permutative approach of SOMA. SOMA was the best performing heuritsic in comparison with the other heuristics in the Car, Rec and Hel problems and the overall the second most effective heuristic for the Taillard problem set.

As far as it can be acertained by the authors, this is the first attempted research of applying SOMA to permutative or NP problems.

Further reseach will entail the application SOMA to other NP problems like Traveling Salesman etc.

ACKNOWLEDGEMENT

REFERENCES

Campbell, H. G., Durek, R.A., and Smith, M.L. 1970. "A heuristic algorithm for the *n* job *m* machine sequencing problem", *Management Science,* 16, (B) 630-637.

Carlier, J. 1978. "Ordonnancements a Contraintes Disjonctives". *RAIRO. Operations Research* 12, 333-351

Heller, J. 1960. "Some Numerical Experiments for an M×J Flow Shop and its Decision- Theoretical aspects". *Operations Research* 8, 178-184

Hu, K., Li, J., Liu, J. and Jiao, L. 2006, "Permutation Flow-Shop Scheduling Based on Multiagent Evolutionary Algorithm". In: *AI 2006: Advances in Artificial Intelligence,* A. Sattar and B.H. Kang (Eds.):, Springer-Verlag ,Berlin, 917-921.

Nawaz, M., Enscore, E. E. Jr, and Ham, I., 1983. "A heuristic algorithm for the *m*-machine, *n*-job flow-shop sequencing problem", *OMEGA International Journal of Management Science* 11, 91-95.

Nowicki, E., Smutnicki, C. 1996. "A fast tabu search algorithm for the permutative flow shop problem". *European Journal of Operations Research*, 91, 160-175

Onwubolu, G. C. and Mutingi, M. 1999. "Genetic algorithm for minimizing tardiness in flow-shop scheduling", *Production Planning and Control* 10 (5) 462-471.

Onwubolu G. C. 2002. *Emerging Optimization Techniques in Production Planning and Control*. Imperial Collage Press, London.

OR Library, 2006 : http://mscmga.ms.ic.ac.uk/info.html

Pinedo, M. 1995. *Scheduling: theory, algorithms and systems*, Prentice Hall, Inc., New Jersey.

Ponnambalam, S. G., Aravindan, P. and Chandrasekhar, S. 2001. "Constructive and improvement flow shop scheduling heuristic: an extensive evaluation", *Production Planning and Control* 12**,** 335-344.

Reeves, C. 1995. "A Genetic Algorithm for Flowshop Sequencing". *Computers and Operations Research*. 22, 5-13

Reeves, C. and Yamada, T. 1998. "Genetic Algorithms, path relinking and flowshop sequencing problem". *Evolutionary Computation* 6, 45-60.

Taillard, E. 1993. "Benchmarks for basic scheduling problems". *European Journal of Operations Research*, 64, 278-285.

Tasgetiren, M. F., Sevkli, M. Liang, Y-C., and Gencyilmaz, G. 2004. "Particle swamp optimization algorithm for permutative flowshops sequencing problems", *4th International Workshops on Ant Algorithms and Swamp Intelligence*, ANTS2004, LNCS 3127, Brussel, Belgium. (Sept) 5-8, 389-390.

Tasgetiren, M. F., Liang, Y-C., Sevkli, M. and Gencyilmaz, G. 2004. "Differential Evolution Algorithm for Permutative Flowshops Sequencing Problem with Makespan Criterion*:, 4th International Symposium on Intelligent Manufacturing Systems*, IMS2004, Sakaraya, Turkey. (Sept) 5-8, 442-452.

Tseng, L. and Lin, T. 2006. "A Hybrid Genetic Algorithm for the Flow-Shop Scheduling Problem", *Lecture Notes in Computer Science*, Springer Berlin, 218-227.

Zelinka, I. 2004. "SOMA – Self Organizing Migrating Algorithm". In: *New Optimization Techniques in Engineering*, Onwubolu, G., Babu, B., (Eds.), Springer Verlag, Germany,

Zelinka, I. 2002. "Artificial Intelligence in the Problems of Global Optimization*". BEN – technicka literatura*, Praha.

DONALD DAVENDRA is a Master of Science graduate in Optimization Techniques from the University of the South Pacific, Fiji Islands. Currently he is a doctoral candidate in Technical Cyberrnatics at the Tomas Bata University in Zlin, Czech Republic. His email address is davendra@fai.utb.cz

IVAN ZELINKA holds the position of Associate Professor**,** Head of Department of Applied Informatics and Vice Dean of Research at the Faculty of Applied Informatics at the Tomas Bata University in Zlin. He is the inventor of Analytic Programming and SOMA, which won him the Siemens Award for outstanding doctoral dissertation in Europe in 2001. His email address is zelinka@fai.utb.cz

Distributed Self-Organizing Migrating Algorithm and Evolutionary Scanning

Pavel Varacha and Ivan Zelinka
Department of Applied Informatics
Tomas Bata University in Zlin
Nad Stranemi 4511, Zlin, 760 05, Czech Republic
E-mail: {varacha,zelinka}@fai.utb.cz

KEYWORDS

neural network, symbolic regression, analytic programming, evolutionary searching, evolutionary scanning, evolutionary algorithms, distributed computing

ABSTRACT

This paper presents an idea of new algorithm combining advantages of evolutionary algorithm and simple distributed computing to perform tasks which required many re-runs of the same program. Computing time is shorted due to elementary distribution within a number of common computers via the Internet. Progressive .NET Framework technology allowing this algorithm to run effectively and examples of possible usage are also described.

The algorithm deals with a problem of synthesis of the artificial neural networks using the evolutional scanning method. The basic task to be solved is to create a symbolic regression algorithm on principles of analytic programming, which will be capable of performing a convenient neural network synthesis. The main motivation here is the computerization of such synthesis and discovering so far unknown solutions.

INTRODUCTION

Distributed Self-Organizing Migrating Algorithm (DISOMA) is a highly effective evolutionary optimization heuristic combining eminent computational performance with attainable technical requirements. DISOMA is derived from the Self-Organizing Migrating Algorithm (SOMA) (Zelinka 2002,2004) which has proven to be very useful for optimization of many hard-solvable problems. However there exist some tasks that cannot be solved by a small band of SOMA executes, for example, a task that is deduced from the model containing a number of stochastic based variables. In such cases huge set of SOMA cycles has to be run and final solution is interpolated through all obtained possibilities via statistic procedures. This technique can be extremely demanding in terms of computing time. The mainframes capable of computing such exacting assignments are not always available. DISOMA reduces such demand by distributing separate SOMA executes to bigger group of independent computers. The Internet connection and .NET Framework presence are the only conditions required and that each computer is to be part of the process. This paper introduces DISOMA as a new method of functional optimization, bringing a utile tool for the users who can not get access to expensive mainframes but have at their disposal large group of personal computers, for example lab computers. Various DISOMA aspects are discussed including installation, structure, methods of remote control and communication, operation system requirements and examples of use. The paper concludes by explaining the importance of DISOMA for connected research of Analytic Programming (Zelinka, Oplatkova 2003) and also for research connected with SOMA itself.

METHODS

Combination of many times proved SOMA and modern .NET Framework architecture is what makes DISOMA such powerful. Main techniques used to construct DISOMA are described in this chapter.

SOMA

SOMA is an effectual artificial intelligence method of functional optimization. Together with Genetic Algorithms, Simulated Annealing and Differential Evolution, SOMA can be treated as an evolutionary algorithm. Set of hypothetic solutions is as a group of individuals moving towards Leader through their environment (N dimensional hyper-plane) to reach the best living condition (functional minimum). Owing to this quality SOMA is able to work out the best (or very good) solution for many problems which are unsolvable by rigorous mathematical techniques. Following table of parameters has to be set SOMA can be run.

Table 1. SOMA parameter

Parametr name:	Description:
CostFunction	function defining the problem it the way of founding minimal value
PathLength	how far an individual goes on its way toward Leader
Step	how many times individual stops during its path
PRT	degree of randomness during individual moves
PopSize	number of individuals within the population
Migrations	how many times population will be migrating toward the Leader
MinDiv	terminal vertical distance between the best and the worst individual

Windows Services

Windows Services are applications that can be automatically started when the operating system boots. They can run without having an interactive user logged on the system. Any Functionality can be built in the service such as scanning for files to do backup or a virus check, or starting a .NET Remoting server, for example. (Robinson 2004)

.NET Remoting

Many applications are not standalone applications that are running on a single system, but they use some network communication technologies to invoke methods on a remote server. This is what .NET Remoting is good for. .NET Remoting can be used for accessing objects in another application domain. Instead of sending data, it can invoke methods across the network across the network. (Robinson 2004)

DISOMA

DISOMA works on ordinary client - server architecture however unconventionally there are many servers and only one client. Within DISOMA structure every server is called "Solver" and the client is called "Scheduler". DISOMA itself can be defined as a set of one Scheduler and altering figure of Solvers. Large number of Solvers communicating with Scheduler (more than 10 recommended) is crucial for DISOMA capability to solve more extensive tasks. Main attributes of Solver and Scheduler are described further.

Solver

Solver is designed as Windows Service. This conception brings basically automated method how to install Solver via Windows Service Installer. Every computer running on Windows XP or Windows 2000 can be easily equipped by one instance of Solver. After installation Solver is running together with Windows no matter if any user is logged in or not. Every solver communicates with Scheduler only and knows nothing about possible existence of other Solvers. Computer with Solver instance has to be connected with Internet to allowed Solver participating on DISOMA.

Solver contains two main parts, communication routines and SOMA. After Windows is switched on Solver sends message to Scheduler that it is ready to participate on calculations and what is its IP address. Than this message is sent periodically till Windows are not switched off. If an order comes from Scheduler, Solver starts computing one instance of SOMA. After an adequate solution is found its coordinate are send back to Scheduler and Solver is ready for another task. If the solution was not found within specified time an error warning is sent to Scheduler and calculation is stopped.

Scheduler

Scheduler is a simple WinForm application connected to SQL database via ADO .NET technology. Scheduler itself can be run without installation on computer equipped by .NET Framework however Microsoft SQL Server has to be present in order to help Scheduler administrating SQL database. Also Scheduler position within the Internet is limited due to its exact IP address with has been specified firmly within Solvers code.

Scheduler has to handle thee main databases - defined tasks, found solutions and connected Solvers. The user creates "Defined tasks database" manually. This definition can be originated directly via SQL server or by the help of Scheduler window. Every defined task has to contain vector of SOMA parameters and Cost Function (see Table 1) also number of repetition can be included. "Connected Solvers database" contains all Solvers connected to Scheduler and has to be periodically updated in order to remain relevant. Every included Solver assumes ready, calculating or error state. Scheduler sends tasks to Solvers which are ready - always one task (the oldest item from Defined task database) to one Solver. When Scheduler receives coordinates of successfully found solution form Solver the solution is stored into "Found solutions database" together with task parameters and solved task is deleted from Defined task database. If any Solver does not return awaited solution within specified time or stops responding its status is changed to error and its task is redirected to another ready Solver.

APLICATION

Class of problems which can be effectively solved by DISOMA includes problems which require solving large number of single SOMA tasks during evaluation

Figure1: DISOMA diagram

which make it ideal tool for Evolutionary Scanning. For more details see (Varacha, Zelinka 2007a).

Successfulness of SOMA sensitively depends on parameters setting (see Table 1). For statistic determination of ideal SOMA parameters many runs of SOMA have to be examines on large representative set of functions. (Zelinka 2004)

EVOLUTIONARY SCANNING

Clause: *Let there be a set of all neural networks with forward running propagation ANN_{all} = {ANN_1, ANN_2, ..., ANN_i, ...} and a set of all functions F_{all} = {f_1, f_2, ...,*

f_k, ...}. Then for each $ANN_i \in ANN_{all}$ exists a function $f_k \in F_{all}$, alternatively a set of functions $F_k \subset F_{all}$ such, that holds $ANN_i \Leftrightarrow f_k$, alternatively $ANN_i \Leftrightarrow F_k$.

Kolmonogorov theorem (Bose and Liang 1996) further shows also validity of the inverse clause: *For every continuous function $f_k \in F_{all}$ exists $ANN_i \in ANN_{all}$ such, that holds $f_k \Leftrightarrow ANN_i$.*

Task: Design an algorithm, which will by the means of the symbolic regression methods, evolutionary scan a set $\mathbf{F_{all}}$ in order to find:

a) $\mathbf{f_k} \Leftrightarrow ANN_i$

b) $\mathbf{f_k}$, whose at least some subfunctions $\{\mathbf{f_1}, \mathbf{f_2}, ...\}$ $\Leftrightarrow \{ANN_n, ANN_m, ...\}$

which solve the particular problem P with global error $E_T < \xi$, where ξ is the user defined biased tolerance threshold.
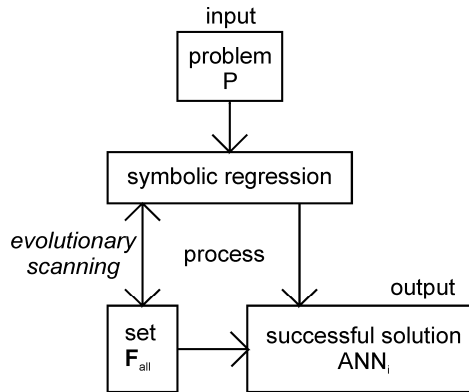
Figures 2: Principle of the Evolutionary scanning

ANALYTIC PROGRAMMING

Algorithm referred to as analytic programming (AP) proved itself by providing the solution to a variety of problems of the symbolic regression as: synthesis of trigonometric functions (Zelinka 2005a), polynomial functions (Zelinka and Oplatkova 2003), functions of boolean parity (Zelinka and Oplatkova 2004), functions of boolean symmetry (Zelinka at al. 2004), solution of differential equations (Zelinka 2002b) and optimization of the artificial ant tour (Zelinka and Oplatkova 2006). Those accomplishments have led to the motivation to utilise an AP also in the ANN synthesis.

AP has been built-up on the sets of functions, operators and so called terminals, which are mostly invariables or independent variables. These mathematical objects form a set, out of which the AP strives to synthesize a suitable solution – so called *General Function Set (*GFS). Members of **GFS** are then the principal functions **gf**. (That is, we can easily understand the

operators and terminals as functions.) **GFS** = {$\mathbf{gf}_1$, ... $\mathbf{gf}i$, ... $\mathbf{gf}n$}.

The basic principle of AP is based on the handling with Discrete Sets (*Discrete Set Handling*, DSH). DSH forms the interface between the Evolutionary Algorithm (*Evolutionary Algorithm*, EA) (Zelinka, 2004), a problem, which should be solved symbolically.

Thanks to that, almost any EA can be applied in the AP. In such a way the EA in effect works as an engine actuating the AP. From the EA nature further results to be rational to define the Cost Function (**CF**), whose value is necessary to be minimized.

Note: AP does not use as terminals particular, however, general constants K1 to Kn, whose particular value is determined by means of nonlinear interlacing as closely as possible near the assessment of the relevant CF.

SYNTHESIS OF NEURAL NETWORKS

To accomplish by the means of an AP the successfully synthesize of the ANN, it is necessary at first to appropriately choose the elements of **GFS** and correctly define the **CF**, applicable for the problem being solved. To fulfill both those steps is, however, necessary to comprehend the relationship of equivalence between the ANN and functions – namely an AP itself does not understand the ANN conception, it is capable to synthesise only functions, eventually algorithms.

The relation between an ANN and mathematical functions

The connection between an ANN and mathematical functions is often neglected in the professional literature, at the same time each ANN can be expressed as to its equivalent class of functions **f***. The ANN synthesis process is thus a process of searching of a suitable **f* in** the set of class of functions **F**, which contains the classes accrued by combinations of (basic) base functions **b** of a set **B**. The constitution of elements of **B** then differs according to individual types of ANN. The process of ANN education can then be expressed as searching for convenient values of invariables, which differentiate particular functions **f** in **f*** from each other. The resulting function **f** is then equivalent to the searched for ANN.
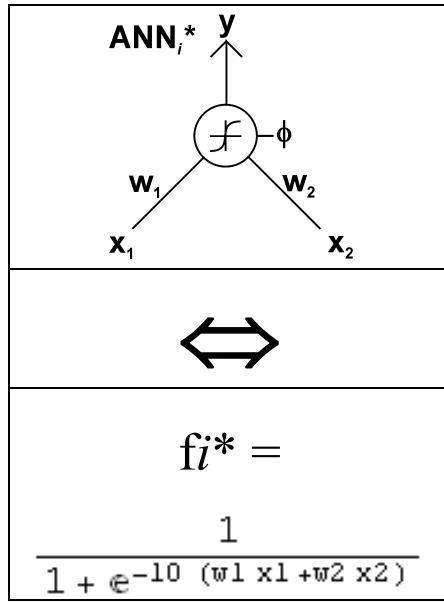
Figures 3: Equivalence between ANNi* and fi*

That previous described feature of ANN together with aforementioned stated characteristics of an AP, makes from the AP an ideal algorithm for the ANN synthesis. The most important task to solve, is to select a suitable **B**, which will be used as a **GFS.**

Conception of used GFS

From the preceding text it is clear, that suitable selection of individual **gf** by using **GFS**, is an absolutely crucial question. As a fundamental base stone - function **gf1** – is synthesised ANN, and the choice then becomes **sigmoid** – the transfer function of two input arguments *arg1*, *arg2*:

$$gf1 = \frac{1}{1 + e^{-\lambda\,(arg1 + arg2)}}$$

Sigmoid function is in the world of ANN the often used transfer function (Master 1993), which is derived from behavior of real neurons in human brain. For a parameter λ the invariable value 10 was selected, so that the sigmoid nears rather to the unipolar binary function, which is suitable for solutions of classification problems. Against the binary function it , however, has the advantage, that there exists a derivation of it in its whole interval and that it is possible to apply the algorithm dependent on such derivation of function, as it is , for example, just the nonlinear interlacing used in the AP, on the structures, which are thus being created.

Beyond **gf1** there must be an integral part of, as constituted by this research, **GFS** also input independent variables x1 and x2, which represent coordinates of individual points on the classified plane, and then invariables K (K1 to Kn), which shall figure the thresholds ϕ and weights w of the network. To be

able to tie up properly the weights and thresholds to the network, it is necessary to add into **GFS** next two **gf,** + (plus) and × (times).

By that one finally arrives at **GFS1** = {**gf1**, +, ×, x1, x2, K}. From the point of view of the problem it handles about so-called **complete GFS,** consequently **GFS1** has by the means of AP guaranteed the potential of solving the ANN syntheses – for the set **B,** through combination of which rises already known solution ANN of the particular problem, is valid **B** $\subseteq$ **GFS1** - so it comprises all elements, which build and compose the relevant **f**.
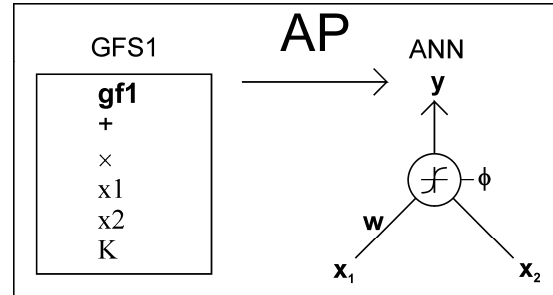
Figures 4: Principle of ANN synthesis from **GFS1**

Such definition of **GFS1** is an attractive one because an AP can, thanks to it synthesize not only classical neural structures, known from the rigorous theory of ANN, however, also the networks of pseudo-neural character, eventually quite non-neural functions. These properties from the world of neural functions shall, thanks to AP involuntarily combine with further mathematical transformations.

Effect of GFS1 on the synthesized ANN topology

The sub-functions can arise from the set **GFS1** under favorable conditions during cultivation of an individual in AP, which will compose the winner individual, and on which can be looked on as neurons in ANN. Examples of such arrangement of individual **gf** $\in$ **GFS1** into convenient subfunctions will be introduced in future studies.

Definition of CF

The correct definition of **CF** is unique for every problem to be solved. **CF** should in itself namely include the whole testing set of the synthesized ANN. **CF** works in such cases on the basis of testing the ANN responses on particular elements of the testing set. What, however, remains always same is the chosen philosophy, under which the relevant **CF** is created.

As a fundamental model of CF1, the staircase conception **CF** was selected for the synthesis. Its operation is, that at the beginning of scoring the **CF** of particular individual (ANN) is valid **CF** = 0. Step by step all elements of the tested set are passed through. If

there shows at any tested element, that the response of ANN is divergent to the requirements of the testing set an increment of **CF = CF + 1** is performed. By that way the CF1 can gather the discrete values {1,2, ..., n}, where n is number of elements in the testing set. CF1 thus calculates on how many elements of testing set replied ANN with the inadmissible reply.

By all of that it is assumed, that at the output of ANN there is inserted one more neuron, which standardizes the replies greater than 0.8 to 1 and replies less than 0.2 damps to 0.

CONCLUSION

Applied on described class of problems DISOMA can compensate expensive mainframes inside organizations which cannot afford such advanced accessories but manages number of standard computer. This appears to be reasonable way how to save unnecessary investments. Presently DISOMA goes trough the stadium of debugging however this algorithm is going to be expose on the Internet for a public use. During the progress many completive ideas, such as creating remote control of Scheduler via ASP .NET services, are proposing and waiting for integration into DISOMA. For more information see (Varacha, Zelinka 2007b).

REFERENCE

BOSE, B.K., LIANG, P. 1996. *Neural Network Fundamentals with Graphs, Algorithms, and Aplications*. McGraw-Hill Series in Electrical and Computer Engineering, 478 p. ISBN 0-07 006618-3.

MASTER T., 1993, Practical Neural Networks Recipes in C++, London : Academi Press, 1993, 490 s. ISBN 0-12-479040-2.

ROBINSON, R., AT AL. 2004 *Professional C# 3rd Edition*, Wiley Publishing, Inc., Indianapolis , ISBN 0-7645-5759-9.

ZELINKA I. 2002a. Analytic programming by Means of Soma Algorithm. Mendel '02, *In: Proc. 8th International Conference on Soft Computing Mendel'02*, Brno, Czech Republic, 93-101., ISBN 80-214-2135-5.

ZELINKA I. 2002b. Analytic programming by Means of Soma Algorithm. ICICIS'02, *First International Conference on Intelligent Computing and Information Systems*, Egypt, Cairo, ISBN 977-237-172-3

ZELINKA I., OPLATKOVA Z. 2003. Analytic programming − Comparative Study. *CIRAS'03, The second International Conference on Computational Intelligence, Robotics, and Autonomous Systems*, Singapore, ISSN 0219-6131.

ZELINKA I., OPLATKOVA Z. 2004. Boolean Parity Function Synthesis by Means of Arbitrarry Evolutionary Algorithms - Comparative Study, *In: 8th World Multicon-ference on Systemics, Cybernetics and Informatics (SCI 2004)*, Orlando, USA, in July 18-21.

ZELINKA I., OPLATKOVA Z., NOLLE L. 2004. Boolean Symmetry Function Synthesis by Means of Arbitrarry Evolutionary Algorithms - Comparative Study, *In: 18th European Simulation Multiconference (ESM 2004)*, Magdeburg, Germany, June 13-16, 2004, ISBN 3-936150-35-4.

ZELINKA I. 2004. *SOMA* - Self Organizing Migrating Algoritm, kap. 7, str. 33, in B.V. Batu, G. Onwubolu (eds), *New Optimization Techniques in Engineering*, Springer-Verlag.

ZELINKA I, OPLATKOVA 2006. Z. Investigation on Artificial Ant using Analytic Programming, *GECCO 2006*, Seatle, WA, USA

VARACHA P., ZELINKA I. 2007a. Distributed Self-Organizing Migrating Algorithm (DISOMA). *ICCC 2007, 8th International Carpathian Control Conference, Štrbkké Pleso*, Slovak Republic, ISBN 978-80-8073-805-1

VARACHA P., ZELINKA I. 2007b. Synthesis of artificial neural networks by the means of evolutionary scanning - preliminary study. *ECMS 2007, 21st European Conference on Modelling and Simulation*, Praha, Czech Republic, ISBN 978-0-9553018-2-7

MODELLING OF STOCHASTIC PARAMETERS FOR CONTROL OF CITY ELECTRIC TRANSPORT SYSTEMS USING EVOLUTIONARY ALGORITHM

Mikhail Gorobetz,
Anatoly Levchenkov
Institute of Industrial Electronics and Electrotechnics,
Riga Technical University
1 Kronvalda blvd., Riga, LV-1010, Latvia
E-mail: mgorobetz@latnet.lv, levas@latnet.lv

KEYWORDS

Modelling, evolutionary algorithm, electric transport

ABSTRACT

This work is based on research in a field of intelligent agent systems, negotiation algorithm solving tasks of energy saving, optimal electric vehicle control and transport flow control in traffic jam.

Main goal of research is energy saving for public electric transport. Mathematical model and evolutionary algorithm is proposed in the paper to solve multi-criteria optimization task minimizing idle time and electric energy used by public electric transport and maximize average speed of the flow in traffic jam.

Paper presents a computer experiment to test proposed mathematical model and workability of evolutionary algorithm. The specific dynamic model with stochastic parameters of city transport system is created and results of evolutionary optimization are simulated.

INTRODUCTION

Authors propose to use electronic intelligent agent system to control traffic lights corresponding to the transport flow. The target is to minimize of total electric energy usage by electric transport and total idle time for all transport flow participants using evolutionary algorithm for intelligent agents of traffic lights [1].

Paper describes the problem of decision-making for city transport control. The decision making modeling for transport problems is described, using methods of decision making theory [2] and process optimization in intelligent transport systems.

As the transport system is a cooperative system, where behavior of one participant depends on other. That is why the negotiation process for intelligent agent is necessary. Each agent controls his own object sending data to the superagent, which is responsible for optimization and coordination of negotiation process.

PROBLEM FORMULATION

Transport flow in the city is growing and traffic jams become the common problem for most of big cities.

Public electric transport should have higher priority than private cars, by criterion of transported passenger number, electric energy consumption and service level evaluated by schedule fulfillment. Public electric transport, such as trams and especially trolleybuses, which are more sensible to traffic jams, uses more electric energy during frequent acceleration and braking in traffic jam and infringe scheduled time. Also traffic lights are not synchronized and working independently from transport flow.

Public electric transport, such as trams and especially trolleybuses, uses more electric energy during frequent acceleration and braking in traffic jam. That is why, traffic light's green light should be synchronized with the schedule of public electric transport. For all other participants of the transport flow the minimization of idle time in traffic jam is needed.

It is multi-criteria optimization task with following criterions: minimization of electric energy and idle time and maximization of average flow speed.

The purpose of this paper is to define fitness function for evolutionary algorithm. Evolutionary algorithm is realized for intelligent superagent to optimize green and red light time. Fitness function should include normalized value of idle time and average speed for all transport participants in traffic jam and electric energy consumption.

METHOD OF SOLUTION

Authors propose to use intelligent agent system 3 for task solution.

Each intelligent agent is responsible only for his own object. The intelligent agent has all necessary information about the object and has possibility to make decisions for the object.

Intelligent agents – the elements of artificial intelligence 4 - are incorporated in the electric transport control system. This system is managed and controlled by the superagent. The superagent is responsible for intelligent agent negotiation and to cooperate the work of autonomous agents to achieve common goals of the system. A superagent is an intelligent agent, which is not responsible for any object, but for processes where more then one object's intelligent agent participate.

Intelligent agents input and process of the information on the physical transport unit as well as send it to intelligent superagent for the analysis of the situation of the transport system in general to realize the decision-making procedure. Alias the superagent realizes negotiation of the intelligent agents, defining a negotiation set. It plays a role of the arbitration judge, which nominates what players have to do proceeding from results of negotiation.
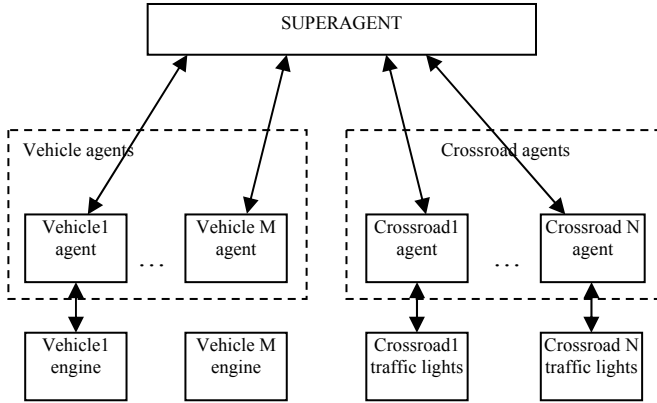

Figure 1: Intelligent agent system

Each electrical transport unit has own intelligent agent. An intelligent agent is a software package that is responsible for an object (fig. 1.) or a process.
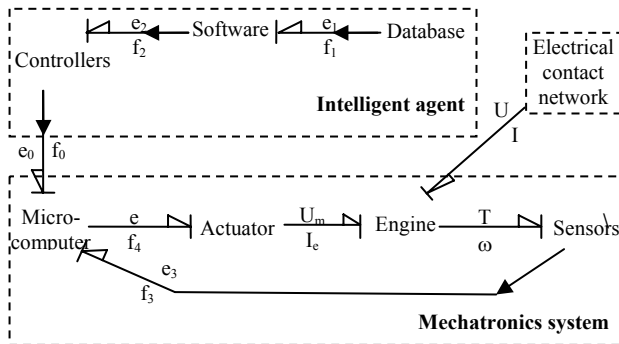

Figure 2: Intelligent agent for mechantronics system control

A intelligent agent consists of three necessary components: database, software, which based on artificial intelligence structures and algorithms, and electronic devices-controllers.

All information about an object such as technical characteristics and parameters, strategies, schedules, calculated data and so on is incorporated in intelligent agent's database.

The software is used to process this data with evolutionary algorithms.

Electronic devices are used to realize intelligent control on machine level.

Each public electric transport unit has own intelligent agent, which is responsible only for the current unit. Physically each transport unit is equipped with the

telemetric system (sensors) that collects data for agents database 5.

MATHEMATICAL MODEL FOR INTELLIGENT AGENT

This part presents mathematical model for task solution. Mathematical model of public electric transport system has a set of dynamic parameters. Such parameters can be number of cars waiting on the crossroads, number of passengers waiting on passenger stops. Also number of lanes, which is very significant parameter for traffic jam problem solution, may be changed in case of breakdowns and crash of cars. All these variables are changing continuously.

That is why evolutionary algorithm is necessary to adopt optimal solution to current input dynamic parameters.

Dynamic parameters has stochastic behaviour. For instance, driver's reaction time, distance between cars, acceleration time and moving speed dpends on human factors, that's why it should be modeled as stochastic parameter.

Static Input Data

- set of crossroads – K
- length of a car - c_i;

Stochastic Dynamic Input Data

- number of transport units – n_{jk}, where j – direction, k - crossroad;
- average speed of moving car (without stop) – v;
- flow rate of crossroad (number of lanes) – l;

Parameters for i^{th} transport unit

- driver's average reaction time – r_i;
- acceleration time for one transport unit – a_i;
- distance between cars in traffic jam - d_i;
- position in traffic jam - p_i.

Variables

- green light time – x_k;
- red light time – y_k;

where k – crossroad.

Target functions

- Average waiting time – $T = \Sigma t_i / \Sigma n \rightarrow min$
- Average flow speed - $V = v_i \rightarrow max$
- Electric energy - $E = \Sigma e_i \rightarrow min$

Auxiliary functions

- time of moving from position to the crossroad -
$$t_i = \frac{(p_i - 1) \cdot (c_i + d_i)}{v} + (r_i + a_i) \cdot p_i$$

- flow rate of traffic light:
$$f = \frac{v \cdot x - c_i - d_i}{c_i + d_i + (r_i + a_i) \cdot v} \cdot l; \quad f \in \aleph$$

- red light times (green light missing):

$$rt = p_i / f; \quad f \in \aleph$$

- red light waiting time: $t_{ir} = rt \cdot y;$

- position before passing crossroad:

$$p'_i = p_i - t_{ir} \cdot (r_i + a_i); \quad f \in \aleph$$

Fitness function for optimization

$$T = \sum_{i=1}^{n} t_i(x,y) = \qquad (1)$$
$$= \left(\sum_{i=0}^{n} \frac{(p'_i - 1) \cdot (c_i + d_i)}{v} + (r_i + a_i) \cdot p'_i + t_{ir} + rt \cdot x \right) \to \min$$

Normalization functions

Let us assume, that maximal values of target functions are:

- T_{max} = 3600 s;
- V_{max} = 50 kmh;
- E_{max} = 1000 kWh.

According to this values normalization is:

$$T' = \frac{T_{max} - T}{T_{max}} = \frac{3600 - T}{3600};$$

$$V' = \frac{V}{V_{max}} = \frac{V}{50};$$

$$E' = \frac{E_{max} - E}{E_{max}} = \frac{1000 - E}{1000}.$$

Let us assume the priorities for optimization parameters are

- α_T = 0.3;
- α_V = 0.3;
- α_E = 0.4.

Fitness function for optimization:

$$F = \alpha_T \, T' + \alpha_V \, V' + \alpha_E \, E' \to \max \quad (2)$$

EVOLUTIONARY ALGORITHM FOR INTELLIGENT AGENT

From evolutionary point of view the parameters of the problem are presented as genes of chromosome. Each chromosome presents problem solution. Each solution should be evaluated by fitness function. The goal of genetic algorithm is to find the chromosome with optimal value of fitness function.

In our case the chromosome contains green light time and red light time of each traffic light:

$$s = (x_1, y_1, x_2, y_2, ..., x_k, y_k)$$

Population in the task is presented as a set of traffic light working modes for all analyzed crossroads:

$$S = \{s_1, s_2, ..., s_p\}$$

Each state s is evaluated by fitness function (2):

$$V_i = F(s_i)$$

General evolutionary algorithm for task solution is following:

Step 1: Initialize population:

$$S = \{s_1, s_2, ..., s_p\};$$

Step 2: Evaluate population:

$$V^S = \{F(s_1), F(s_2), ..., F(s_p)\};$$

Step 3: Arrange population by evaluation:

$$\overline{S} = \{\overline{s}_1, \overline{s}_2, ..., \overline{s}_p\}, \quad F(\overline{s}_1) = \max(V^S);$$

Step 4: Select best parents to elite

$$S_E \subset \overline{S};$$

Step 5: Select other parents for crossover

$$S_C = \overline{S};$$

Step 6: Apply crossover, where

$$\overline{s}_i \Pi \overline{s}_j \to s'_i = s_{ij}; s'_j = s_{ji}, \quad i, j = \overline{1, p};$$

Step 7: Apply mutation, where

$$x_j^{s'_i} = x_j^{s'_i} + 1, \quad s'_i \in S',$$
$$j = rand(\overline{1,k}), \quad i = rand(\overline{1,p}) \; ;$$

Step 8: Evaluate population:

$$V^{S'} = \{F(s'_1), F(s'_2), ..., F(s'_p)\};$$

Step 9: Arrange new population by evaluation:

$$\overline{S}' = \{\overline{s}'_1, \overline{s}'_2, ..., \overline{s}'_p\}, \quad F(\overline{s}'_1) = \max(V^{S'});$$

Step 10: Combining with elite:

$$S = (S_E + \overline{S}');$$

Step 11: Delete last:

$$S = S / \{s_{p+1}, s_{p+2|}, ...\};$$

Step 12: IF $F(s_1) <> F(s_p)$

THEN Goto Step 2
ELSE End of Evolutionary

Algorithm with solution s_1.

PRACTICAL EXAMPLE

An abstract crossroad is selected for a computer experiment (Fig.3). Function (1) is optimized in this experiments with following parameters:
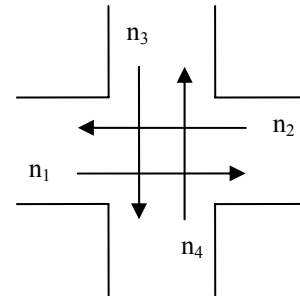

Figure 3: Schema of crossroad for computer experiment.

Let us assume,
- number of vehicles on the first street:

$$n_1 + n_2 = 100$$

- number of vehicles on the second street:
$$n_3 + n_4 = 200$$
- number of lanes: $l = 2$
- average speed of moving vehicle: $v = 25$;
- driver's average reaction time: $r_i = 1$;
- acceleration time – $a_i = 3$;
- length of a car: $c_i = 4$;
- distance between cars in traffic jam: $d_i = 0,5$;

Target is to minimize average waiting time for all vehicles by optimal traffic light regulation (1).

$$T(x, y) \rightarrow \min$$

Input data for evolutionary algorithm are presented on Figures 4-6.

Figure 4: Setting limits for traffic light optimization algorithm

Figure 5: Input parameters for crossroad

Figure 6: Input parameters for evolutionary algorithm

Results of evolutionary algorithm are presented on figure 7.

Figure 7: Results of evolutionary algorithm

SIMULATION OF CITY TRANSPORTATION SYSTEM

The specific modelling environment is developed by authors for computer experiment 6.

For the practical example object the tram system in Riga centre is taken shown on the figure 8.

The part of the route selected for test is the way from the passenger stop "Centraltirgus" to the stop "Grecinieku iela" for one tram.

All simulation dynamics parameters of a tram are taken from the real tram T3-A.

Authors results prove, that the application of proposed algorithm for green wave organization can be very useful for solving energetics and electrotechnology problems in public electric transport systems.
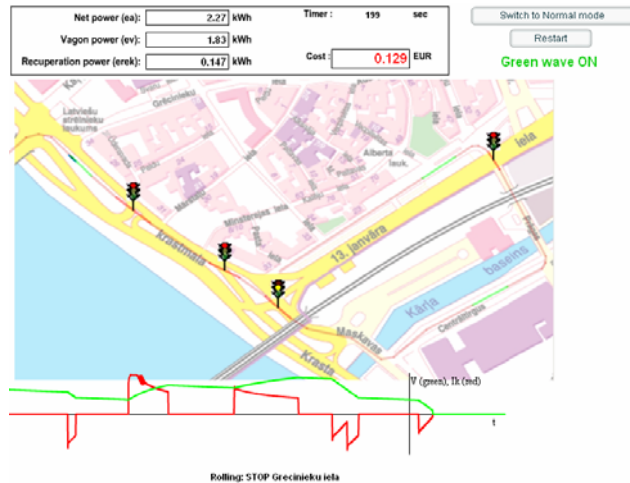
Figure 8: Simulation of intelligent traffic light control

The main advantage of proposed algorithm is energy saving by more than 20% at least and time saving by more than 40%. That means, the results of using intelligent agents in public electric transport motion control allow to reduce running costs and to cut down load on traffic network by decreasing time spent in the city centre as well as offer of much faster service.

This research presents results of traffic control using evolutionary optimization of traffic light working time.
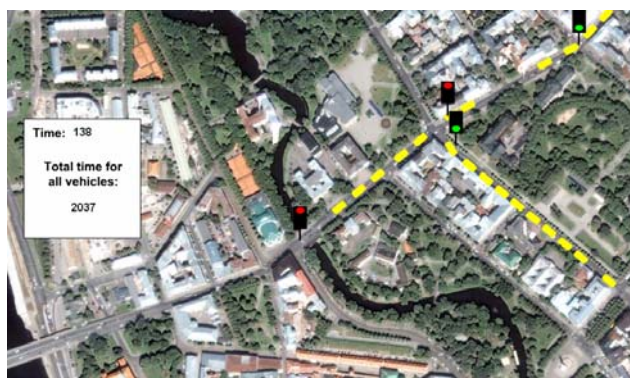
Figure 9: Computer model for traffic control

Figure 9 presents the model of traffic jam without evolutionary algorithm. Model shows current situation in Riga, where traffic jam is very actual problem.

Results of evolutionary algorithm show the necessity to set up maximal allowed time for green light.

Figure 10 show total waiting time for usual traffic light working mode.
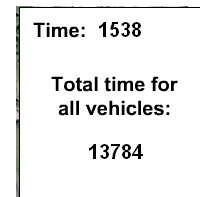
Figure 10: Results of simulation without optimization

Figure 11 provides total waiting time with evolutionary algorithm results.
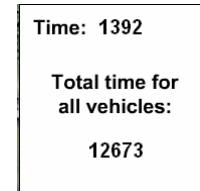
Figure 11: Results of simulation using evolutionary algorithm

On current stage of research 8% of time is saved by using of evolutionary algorithm results.

ACKNOWLEDGMENT

CONCLUSION

Results of practical examples on this stage of research prove that evolutionary algorithm is useful for traffic control task solution.

Advantage of using evolutionary algorithm is the possibility of continuous optimization of traffic control.

Evolutionary algorithm is very sensitive to its parameters. That is why is necessary to find out the most suitable algorithm of selection, crossover as well as number of bits, population size, number of loops and mutation.

Also minimal and maximal limits of variables x and y is very important. Experiment shows that in case of high maximal limit of green light, algorithm tries to switch on green light as long as possible to allow all vehicles to pass the crossroad. Otherwise, the minimal limit of green light is optimal.

REFERENCES

1. J. Branke. Evolutionary Optimization in Dynamic Environments, Kluwer Academic Publishers, 2002, 224 p.
2. M. Gorobetz, A. Levchenkovs, N. Kunicina, L. Ribickis. Modelling of Decision Support System for Intelligent Public Electric Transport. //In Proceedings of International Conference on

Industrial Engineering and Systems Management, Bejing, China, 2007, 334 p .

3. S. J. Russel, P. Norvig. Artificial Intelligence. A Modern Approach, 2nd edition. Prentice Hall, 2006, 1408 p.
4. G. F. Luger. Artificial Intelligence. Structures and Strategies for Complex Problem Solving, Williams, 2003
5. J. Greivulis, I. Raņķis. Iekārtu vadības elektroniskie elementi un mezgli. - Rīga: Avots, 2005. – 288 lpp.
6. M.Gorobetz, A.Levchenkov, L.Ribickis. Intelligent Electric Vehicle Motion And Crossroad Control. //In Proceedings of 12th International Power Electronics and Motion Control Conference. Portoroz, Slovenia, 2006 – 1239 - 1246 p

AUTHOR BIOGRAPHIES

ANATOLY LEVCHENKOV is Professor at Riga Technical University, Faculty of Power And Electrical Engineering, Institute of Industrial Electronics and Electrotechnics from 2006. 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.

MIKHAIL GOROBETZ is graduated Master of Information Technology at Riga Technical University Computer Science and Information Technology Faculty Information Technology Institute in 2005. At present he is a researcher and last year PhD student at Riga Technical University Institute of Industrial Electronics and Electrotechnics .
.

TIME SERIES ANALYSIS OF HEAT DEMAND

Bronislav Chramcov
Jaroslav Baláté
Tomas Bata University in Zlín
Faculty of Applied Informatics
Nad Stráněmi 4511, 760 05 Zlín, Czech Republic
E-mail: chramcov@fai.utb.cz

KEYWORDS

Prediction, District Heating Control, Box-Jenkins, Control algorithms, Time series analysis.

ABSTRACT

The paper deals with an analysis of time series of heat demand course. The course of heat demand and heat consumption can be demonstrated by means of heat demand diagrams. The most important one is the Daily Diagram of Heat Supply (DDHS) which demonstrates the course of requisite heat output during the day. This analysis is utilized for building up a prediction model of the DDHS. Forecast of this heat demand course is significant for short-term and long-term planning of heat production. This forecast is very important for technical and economic considerations. The solved prediction is especially determined for the qualitative-quantitative method of hot-water piping heat output control, i.e. the Baláté System which enables to eliminate any transport delay. Its importance is increased especially for controlled systems having great time constants and great transport delay. In this paper we have focused on determination of difference order, autoregressive process order and moving average process order.

INTRODUCTION

Improvement of technological process control level can be achieved by the time series analysis in order to predict their future behaviour. We can find applications of this prediction also in the control of the Centralized Heat Supply System (CHSS), especially for the control of hot water piping heat output. Knowledge of heat demand is the base for input data for the operation preparation of CHSS. The term "heat demand" means an instantaneous heat output demanded or instantaneous heat output consumed by consumers. The term "heat demand" relates to the term "heat consumption". It expresses heat energy which is supplied to the customer in a specific time interval (generally a day or a year). The course of heat demand and heat consumption can be demonstrated by means of heat demand diagrams. The most important ones are:

- **daily diagram of heat supply** which demonstrates the course of requisite heat output during the day. (See Figure 1)
- **duration heat demand diagram** - Y-coordinates represent heat demand and distance from zero

represents duration of corresponding heat demand. Daily and yearly duration heat demand diagrams are currently known.
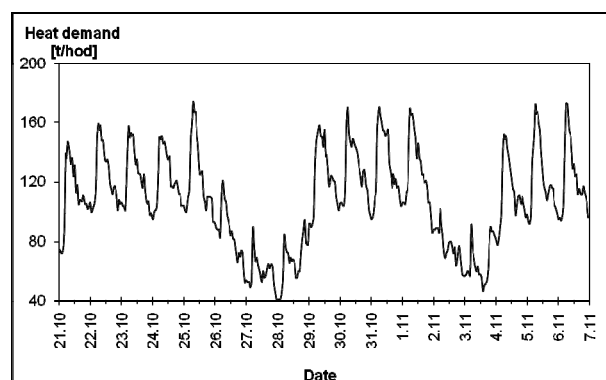
Figure 1: Time series of DDHS from the concrete locality

These diagrams are of essential importance for technical and economic considerations. Therefore forecast of the diagrams course is significant for short-term and long-term planning of heat production. It is possible to judge the question of peak sources and particularly the question of optimal load distribution between the cooperative production sources and production units inside these sources according to the time course of heat demand. The forecast of DDHS is used in this case (Baláté 1982).

FORECAST MODEL FOR CONCRETE TIME SERIES OF HEAT DEMAND

It is possible to use different solution methods for the calculation of time series forecast. (For example: solution by means of linear models, solution by means of non-linear models, spectral analysis method, neural networks etc.). In the past, lots of works were created which solved the prediction of DDHS and its use for controlling the District Heating System (DHS). Most of these works are based on mass data processing. Nevertheless, these methods have a big disadvantage that may result in out-of-date real data. From this point of view it is suitable to use the forecast methods according to the Box –Jenkins methodology (Box and Jenkins 1976). This method works with a fixed number of values which are updated for each sampling period.

This methodology is based on the correlation analysis of time series and works with stochastic models which enable to give a true picture of trend components and also that of periodic components. As this method achieves very good results in practice, it was chosen for the calculation of DDHS forecast.

Identification of time series model parameters is the most important and the most difficult phase in the time series analysis. This paper is dealing with the identification of a model of concrete time series of the DDHS. We have particularly focused on determination of difference degree as well as on obtaining a suitable order of autoregressive process and order of moving average process.

DETERMINATION OF DEGREE OF DIFFERENCING

Many observed non-stationary time series exhibit certain homogeneity and can be accounted for by a simple modification of the ARMA model, the autoregressive integrated moving average (ARIMA) model. Determination of a degree of differencing d is the main problem of ARIMA model building. In practice, it seldom appears necessary to difference more than twice. That means that stationary time series are produced by means of the first or second differencing. A number of possibilities for determination of difference degree exists.

1. It is possible to use a plot of the time series, for visual inspection of its stationarity. In case of doubts, the plot of the first or second differencing of time series is drawn. Then we review stationarity of these series.
2. Investigation of estimated autocorrelation function (ACF) of time series is a more objective method. If the values of ACF have a gentle linear decline (not rapid geometric decline), an autoregressive zero is approaching 1 and it is necessary to difference.
3. Anderson (Anderson 1976) prefers to use the behaviour of the variances of successive differenced series as a criterion for taking a decision on the difference degree required. The difference degree d is given in accordance with the minimum value of variances $\sigma_z^2, \sigma_{\nabla z}^2, \sigma_{\nabla^2 z}^2 \dots$.

An example of the determination of the difference degree for our time series of DDHS is shown in this part of paper. The course of time series of DDHS contains two periodic components (daily and weekly periods). The daily period presents increase and decrease in heat demand during the day. Heat demand drop at the weekend forms the weekly period of DDHS. The general model according to Box-Jenkins enables to describe only one periodic component. From this point of view we will further consider only time series of the DDHS without Saturday and Sunday values. This time series (see Figure 2) exhibits an evident non-stationarity. It is necessary to difference. The course of time series of the first differencing is shown in Fig. 3. The differenced series looks stationary now.
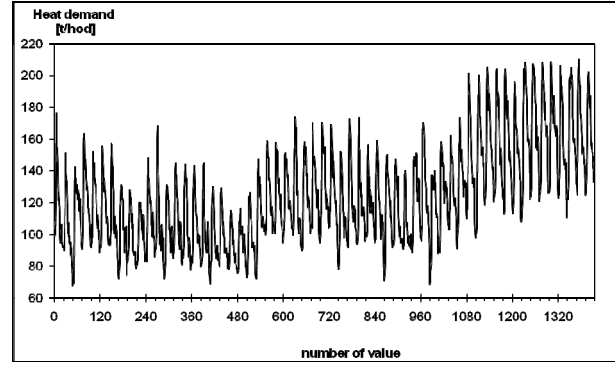

Figure 2: The course of DDHS without values of Saturday and Sunday
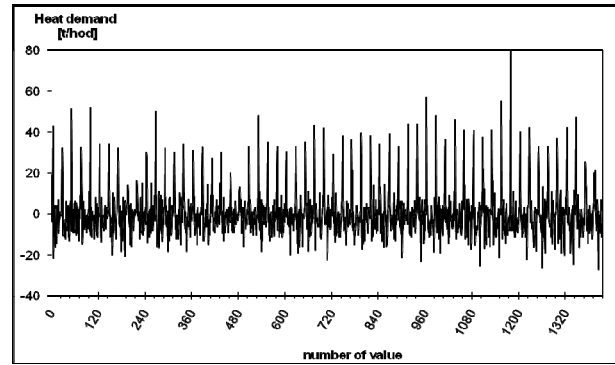

Figure 3: The course of DDHS after the first differencing
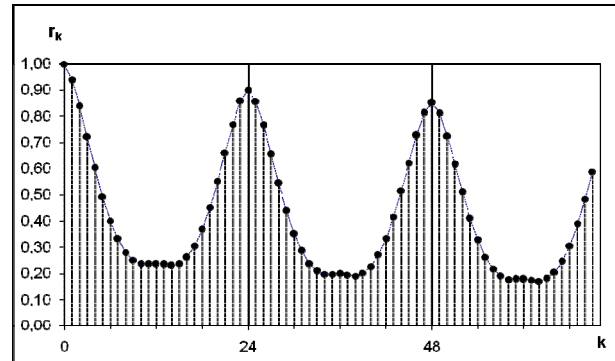

Figure 4: The course of estimated ACF of DDHS
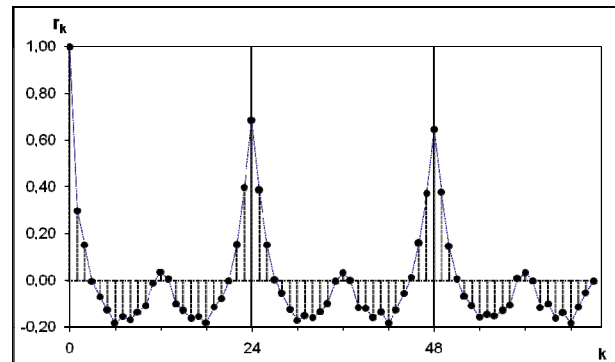
Figure 5: The course of estimated ACF of the DDHS after the first differencing

This fact is confirmed by the sample ACF (see Figure 4 and Figure 5). It is also possible to use the estimated variance of non-differenced and differenced series for making a comparison. In our case we have obtained the following results (1):

$$\hat{\sigma}_z^2 = 967.7; \ \hat{\sigma}_{\nabla z}^2 = 119.9; \ \hat{\sigma}_{\nabla^2 z}^2 = 168.2. \quad (1)$$

From these results and Figures above, it is suitable to choose the first differencing of the DDHS (*d*=1).

In practice, many time series have seasonal components. These series exhibit periodic behaviour with a period *s*. Therefore it is necessary to determine a degree of seasonal differencing - *D*. The seasonal differencing is marked by ∇_s^D. In seasonal models, necessity of differencing more than once occurs very seldom. That means *D*=0 or *D*=1. It is possible to decide on the degree of seasonal differencing on the basis of investigation of sample ACF. If the values of ACF at lags *k*s* achieve the local maximum, it is necessary to make the first seasonal differencing (*D*=1) in the form $\nabla_s z_t$.

As our time series of the DDHS exhibits an obvious seasonal pattern, it is necessary to make an analysis of the seasonal differencing of our time series, as well. The course of the ACF sample evidences the seasonal pattern (see Figure 4 or Figure 5). These functions have their local maxima at lags 24, 48, ... That represents a seasonal period of 24 hours by a sampling period of 1 hour. On the basis of the executed analysis, it is necessary to make the first differencing and also the first seasonal differencing of the DDHS in the form (2).

$$\nabla\nabla_{24}z_t = z_t - z_{t-1} - z_{t-24} + z_{t-25} \quad (2)$$

The course of differenced DDHS in the form (2) is shown in Figure 6.
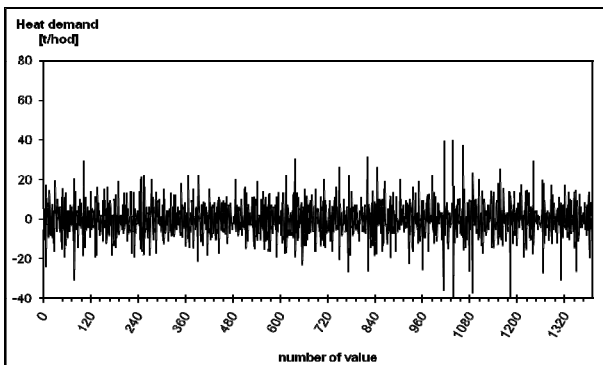
Figure 6: The course of DDHS after the first differencing and after the first seasonal differencing

For the comparison, it is possible to calculate the variances of time series and differenced series according to (Anderson 1976). The results are shown in the form (3).

$$\hat{\sigma}_z^2 = 967.7; \ \hat{\sigma}_{\nabla z}^2 = 119.9; \ \hat{\sigma}_{\nabla^2 z}^2 = 168.2;$$
$$\hat{\sigma}_{\nabla_{24}z}^2 = 150.5; \ \hat{\sigma}_{\nabla\nabla_{24}z}^2 = 72.7 \quad (3)$$

DETERMINATION OF AUTOREGRESSIVE PROCESS ORDER AND MOVING AVERAGE PROCESS ORDER

After differencing the time series, we have to identify the order of autoregressive process *AR(p)* and order of moving average process *MA(q)*. The traditional method consists in comparing the observed patterns of the sample autocorrelation and partial autocorrelation functions with the theoretical autocorrelation and partial autocorrelation function patterns. These theoretical patterns are shown in Table 1.

Table 1: Behaviour of theoretical autocorrelation and partial autocorrelation function

Model	ACF	PACF
AR(p)	Tails off	Cuts off after *p*
MA(q)	Cuts off after *q*	Tails off
ARMA(p,q)	Tails off	Tails off

The expression *Tails off* in Table 1 means that the function decreases in an exponential, sinusoidal or geometric fashion, approximately, with a relatively large number of nonzero values. Conversely, *Cuts off* implies that the function truncates abruptly with only a very few nonzero values. The standard errors of the ACF and PACF samples are useful in identifying nonzero values. As a general rule, we would assume an autocorrelation or partial autocorrelation coefficient to be zero if the absolute value of its estimate is less than twice its standard error. Order of autoregressive operator *p* and order of moving average operator *q* are not usually high; therefore only 20 values of ACF and PACF will be enough to compute. The ACF and PACF samples for the time series of the DDHS are shown in Figure 7 and Figure 8. For assistance in interpreting these functions, two-standard-error limits are plotted in the graphs. We can see from Figure 7 that the ACF sample cuts off after one lag, while the PACF sample tails off with a sinusoidal decline. Thus, we would tentatively identify the underlying model of our series to be a stationary MA(1) model.
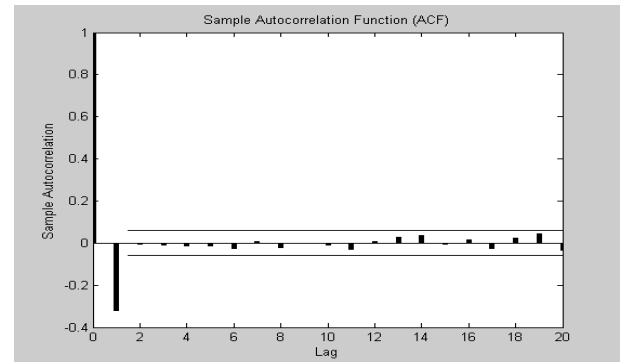
Figure 7: The course of ACF of differencing DDHS
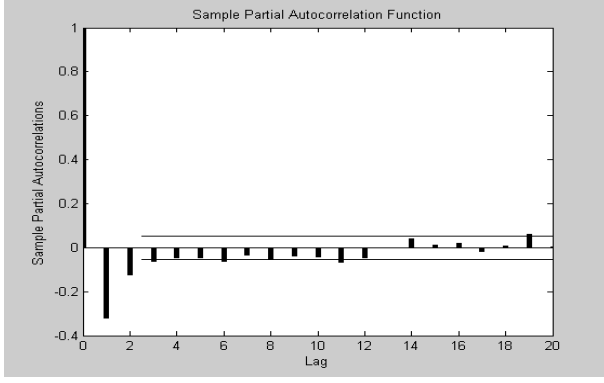
Figure 8: The course of PACF of differencing DDHS

The order of model is usually difficult to determine on the basis of the ACF and PACF. This method of identification requires a lot of experience in building up models. From this point of view it is more suitable to use the objective methods for the tests of the model order.

Criteria for model order

A number of procedures and methods exists for testing the model order (Cromwell 1994). These methods are based on the comparison of the residuals of various models by means of special statistics. In this paper, the Akaike Information Criterion (AIC) in the form (4), Bayesian Information Criterion (BIC) (5) and Schwarz test in the form (6) are used for testing.

$$AIC(p,q) = n \cdot ln\,\hat{\sigma}_a^2 + 2(p+q) \qquad (4)$$

$$BIC(p,q) = n \cdot ln\,\hat{\sigma}_a^2 - (n-p-q)\,ln(1-\tfrac{p+q}{n})$$
$$+ (p+q)\,ln\,n + (p+q)\,ln\left(\frac{\frac{\hat{\sigma}_z^2}{\hat{\sigma}_a^2}-1}{p+q}\right) \qquad (5)$$

$$SC(p,q) = n \cdot ln\,\hat{\sigma}_a^2 + (p+q) \cdot ln\,n \qquad (6)$$

Where n is a number of residuals, $\hat{\sigma}_a^2$ is an estimate of the residual variance, $\hat{\sigma}_z^2$ is the variance associated with z_t.

The test results of the model order of the DDHS are included in the next tables (see Tables 2, 3 and Table 4). The tables represent the values of the particular criteria in dependence on model order (p, q).
The minimum value of BIC and SC is for p=0 and q=1, while the minimum value of AIC is for p=2 and q=1. AIC for p=0 and q=1 is very close to the minimum value. According to these results and on the basis of the general theorem (small values of model order), it is obvious that we would tentatively identify our time series as the MA(1) process. This fact is also confirmed by the analysis of ACF and PACF samples described above. Adequacy of this tentative model may be examined by means of Portmanteau test.

Table 2: Values of Akaike Information Criterion

q \ p	0	1	2	3	4	5	6	7
0	1688	1647.657	1649.6	1649.2	1652.2	1650.2	1652.2	1654.2
1	1653.1	1649.6	4357.9	1651.7	2530.5	1652.3	1654.2	4409.1
2	1651.6	1647.655	1651.1	1792.6	1906.5	1654.2	4389.6	3979.6
3	1651.4	1652.4	2032.4	1942.2	2222.5	1655.6	1657.3	2262.8
4	1653.1	2391.6	1704.9	1659.4	1656	1657.3	3175.7	2304.2
5	1652.6	1654.1	1656.8	1675.3	1656.3	2486.3	1934.2	2043.5
6	1653.3	1655.9	1926.3	1749	2412.5	2049.4	2024.3	2984.5
7	1654.9	1657	1659	2469.5	2493.4	2027	3869	N/A

Note: Value *N/A* in the tables means that the criterion is not available.

Table 3: Values of Bayesian Information Criterion

q \ p	0	1	2	3	4	5	6	7
0	N/A	1654.7	1662.4	1667.2	1675.1	1677.7	1684.1	1690.4
1	1660.2	1662.4	4369.4	1674.6	2530.9	1684.2	1690.4	4431.7
2	1664.4	1665.6	1674	1818	1933.7	1690.4	4412.2	4003.9
3	1669.4	1675.3	2053.8	1968.7	2244.6	1695.9	1701.6	2289
4	1676	2403.4	1735.9	1695.4	1696.3	1701.5	3199.9	2329.6
5	1680.2	1686	1692.9	1715.2	1700.6	2490.4	1976.7	2084.3
6	1685.3	1692.1	1959.9	1790.8	2428.1	2087.4	2065.9	3010.4
7	1691.1	1697.3	1703.3	2477.3	2495.1	2068.5	3899.2	N/A

Table 4: Values of Schwarz Criterion Test

q \ p	0	1	2	3	4	5	6	7
0	1688	1651.5	1657.5	1661	1667.9	1669.8	1675.8	1681.7
1	1657	1657.5	4369.7	1667.4	2550.2	1675.8	1681.7	4440.5
2	1659.4	1659.3	1666.8	1812.3	1930.1	1681.7	4421	4015
3	1663.2	1668.1	2052.1	1965.8	2250	1687	1692.7	2302.1
4	1668.9	2411.2	1728.5	1686.9	1687.5	1692.6	3215	2347.4
5	1672.3	1677.7	1684.3	1706.8	1691.7	2525.6	1977.4	2090.7
6	1676.9	1683.4	1957.8	1784.4	2451.8	2092.6	2071.5	3035.6
7	1682.5	1688.4	1694.4	2508.8	2536.6	2074.2	3920.1	N/A

Diagnostic checking – Portmanteau test

If the fitted model is adequate, it should transform the observations to a white noise process. Thus, a logical method of diagnostic checking is to compute the residuals and then estimate and examine their autocorrelation function. If the model is appropriate, then the residual sample autocorrelation function should not differ significantly from zero for all lags greater than one. Adequate model can be tested by means of Portmanteau test. We may obtain an indication of whether the first *K* residual autocorrelation considered together indicate adequacy of the model. This indication may be obtained through an approximate chi-square test of model adequacy. We consider the test statistic in the form (7), which is approximately distributed as chi-

square with *K-p-q* degrees of freedom if the model is appropriate.

$$Q = (n - d - s) \cdot \sum_{k=1}^{K} r_k^2 (\hat{a}) \qquad (7)$$

Where *n* is a number of residuals, d is the difference degree, s is the season period, $r_k^2(\hat{a})$ is value of ACF sample of residuals.

If the model is inadequate, the calculated value of *Q* will be too large. Thus we should reject the hypothesis of model adequacy if *Q* exceeds an appropriately small upper tail point of the chi-square distribution with *K-p-q* degrees of freedom. Further details of this test are in (Box and Pierce, 1970).

The sample autocorrelation function of the residuals for the model MA(1) for our time series of DDHS is shown in Figure 9. None of the 20 autocorrelations exceeds twice their standard errors. Furthermore, the chi-square statistic applied to the first 20 autocorrelations (*K=20*) is *Q=20.1995*, with 19 degrees of freedom. Comparing this value with a 5 percent value chi-square variable with 19 degrees of freedom, we find out that $\chi^2_{0.05,19} = 30.14$ and so we would conclude that there is no strong evidence to reject the model.
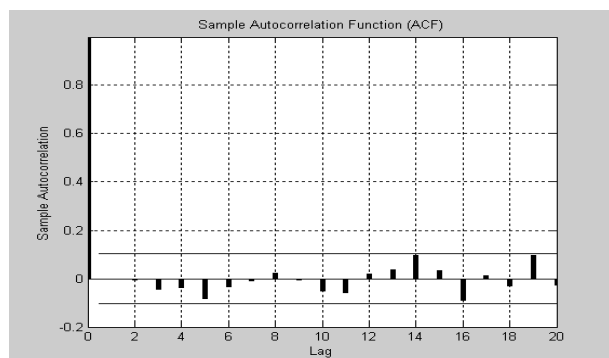
Figure 9: The ACF sample of residuals of DDHS

CONCLUSION

This paper presents the method for building up the model of time series of DDHS. The model is used for prediction of heat demand. This prediction of DDHS is necessary for the control in the Centralized Heat Supply System (CHSS), especially for the qualitative-quantitative control method of hot-water piping heat output – the Baláte System (Baláte 1982). The modelling is based on the Box-Jenkins methodology. The time series analysis was made for the DDHS from the concrete locality.

REFERENCES

Anderson, O. D. 1976. "*Time Series Analysis and Forecasting – The Box-Jenkins approach*". London and Boston: Butterworths, 180 p., ISBN: 0-408-70926-X.

Baláte, J. 1982. "*Design of Automated Control System of Centralized Heat Supply*". Thesis of DrSc (Doctor of Science) Work. Faculty of Mechanical Engineering of TU Brno, Brno.

Box G.E.P. and G.M. Jenkins. 1976. "*Time Series Analysis-Forecasting and Control*". USA: Holden Day, 575 p., ISBN 0-8162-1104-3.

Box G.E.P. and D.A. Pierce. 1970. "Distribution of Residual Autocorrelations in Autoregressive-Integrated Moving Average Time Series Models". *Journal of the American Statistical Association*, vol. 64.

Cromwell, J.B., W.S. Labys and M. Terraza. 1994. "*Univariate Tests for Time Series Models*". SAGE Publication, Thousand Oaks, California, ISBN 0-8039-4991-X.

ACKNOWLEDGEMENT

AUTHOR BIOGRAPHIES

BRONISLAV CHRAMCOV was born in Uherské Hradiště, Czech Republic, in 1975. He studied Automatization and control technology at the Faculty of Technology in Zlin of the University of Technology in Brno, and he took his degree in 1998. In 2006 he graduated his doctoral degree from the Faculty of Applied Informatics of Thomas Bata University in Zlin. His doctoral thesis was focused on the utilization of time series prediction for control of technological process. He is working now as a lecturer at the Faculty of Applied Informatics of Thomas Bata University in Zlin. His research activities are focused on Control Algorithms for District Heating Systems, Time Series Forecast in Energy or Using of Fuzzy Logic for Time Series Forecast. His email address is: chramcov@fai.utb.cz.

JAROSLAV BALÁTĚ was born in Kolín, Czech Republic in 1931. He graduated from the Faculty of Mechanical Engineering of Slovak Technical University Bratislava. He worked in Power and Heating Plant Brno for 12 years as a process engineer, head of automation and measurement department. In 1966 he started to work at TU Brno-Faculty of Mechanical Engineering gradually as an Assistant Professor and a Professor (he became Professor of Technical Cybernetics at the Czech Technical University Prague, Faculty of Mechanical Engineering in 1986). He gradually managed the Department of Hydraulic Machines and Equipment and the Department of Automation. At present, he is working at the Faculty of Applied Informatics of Thomas Bata University in Zlin. His research activities are focused on Technical cybernetics, Theory of automatic control, Application of automatic control on the energy processes, District heating control for large heat supply systems. His email address is: balate@fai.utb.cz.

EVOLUTIONARY OPTIMIZATION OF CHAOS CONTROL – A NEW APPROACH

Roman Senkerik
Ivan Zelinka
Department of Applied Informatics
Faculty of Applied Informatics, Tomas Bata University in Zlin
Nad Stranemi 4511, Zlin 762 72, Czech Republic
E-mail: { senkerik, zelinka }@fai.utb.cz

KEYWORDS

Chaos, control, optimization, evolutionary algorithms, parameter estimation

ABSTRACT

This research deals with optimization of the control of chaos by means of evolutionary algorithms. The main aim of this work is to show that evolutionary algorithms are capable of the optimization of chaos control and to show a new approach of solving this problem and constructing new cost functions operating in "blackbox mode" without previous exact mathematical analysis of the system, thus without knowledge of stabilizing target state. As a model of deterministic chaotic system, the two dimensional Henon map was used. The optimizations was realized in several ways, each one for another desired periodic orbit. The evolutionary algorithm Self-Organizing Migrating Algorithm (SOMA) was used in four versions. For each version, repeated simulations were conducted to outline the effectiveness and robustness of used method and cost function.

INTRODUCTION

Currently, evolutionary algorithms (EA) are known as powerful tools for almost any difficult and complex optimization problem. But the quality of obtained through the optimizations mostly depends on proper design of the used cost function, especially when the EAs are used for optimization of chaos control. It is well known that deterministic chaos in general and also any technique to control of chaos are sensitive to parameter setting, initial conditions and in the case of optimization they are also extremely sensitive to the construction of used cost function.

This research utilized the Pyragas's delayed feedback control technique (Just 1999, Pyragas 1992). Unlike the original OGY control method (Ott, *et al.* 1990) it can be simply considered as a targeting and stabilizing algorithm together in one package (Kwon 1999). Another big advantage of Pyragas method is the amount of accessible control parameters. This is very advantageous for successful use of optimization of parameters setting by means of EA, leading to

improvement of system behavior and better and faster stabilization to the desired periodic orbits. Some research in this field has been recently done using the evolutionary algorithms for optimization of local control of chaos (Richter and Reinschke, 2000).

This research is concerned with the investigation of the design of the "blackbox mode" cost functions securing the stabilization to desired UPO (unstable periodic orbit) without previous mathematic analysis of system, thus without knowledge of exact UPOs position. The control law is based on Pyragas method: Extended delay feedback control – ETDAS (Pyragas 1995). This research is a continuation of previous experiments with application of EA to chaos control (Zelinka 2007, Senkerik 2007, Senkerik 2006).

PROBLEM DESIGN

Problem selection and case studies

The chosen example of a chaotic system was the two dimensional Henon map in form (1).

$$\begin{aligned} x_{n+1} &= a - x_n^2 + b y_n \\ y_{n+1} &= x_n \end{aligned} \tag{1}$$

This work primarily consists of three case studies. All of them are focused on an estimation of three accessible control parameters for EDTAS control method to stabilize desired UPO and a comparison of obtained results for used cost function. Desired UPOs are following: p-1 (a fixed point) in the first case, p-2 in the second case and p-4 in the last case. All simulations were 50 times repeated for each EA version. The control method – ETDAS in the discrete form suitable for two dimensional Henon map has the form (2).

$$\begin{aligned} x_{n+1} &= a - x_n^2 + b y_n + F_n \\ F_n &= K\big[(1-R)S_{n-m} - x_n\big] \\ S_n &= x_n + R S_{n-m} \end{aligned} \tag{2}$$

where K and R are adjustable constants, F is the perturbation, S is given by a delay equation utilizing previous states of the system and m is the period of m-periodic orbit to be stabilized. The perturbation F_n in

equations (2) may have an arbitrarily large value, which can cause diverging of the system outside the interval {-1.5, 1.5}. Therefore, F_n should have a value between $-F_{max}$, F_{max} and EA should find an appropriate value of this limitation to avoid the diverging of system.

The cost function

Several unique designs of cost functions (CF) were developed and tested for stabilization of p-1 orbit (fixed point), p-2 orbit and p-4 orbit. In previous research the CF had been calculated in general from the distance between desired state (desired UPO) and actual system output on simulation interval $-\tau$. The minimal value of this cost function giving the best solution is zero. The aim of all the simulations was to find the best solution that returns the cost function value as close as possible to zero. This CF is given by (3).

$$CF = \sum_{t=0}^{\tau} |TS_t - AS_t| \qquad (3)$$

where: TS - target state, AS - actual state

Other cost functions (CF1 CF2 and CF4) had to be used for stabilizing of the chaotic system in "blackbox mode" ie. without exact numerical value of target state. In this case it is not possible to use the simple rule of minimizing the area created by the difference between the required and actual state on the whole simulation interval $-\tau$.

Our noval approach is based on searching for periodic orbits in chaotic attractor and stabilizing the system on these periodic orbits by means of applying the optimal value of controller adjustable constants (K, F_{max}, R). It means that this new CF did not take into consideration any numerical target state, but the selected target behavior of system. Therefore the new CFs are based on searching for optimal values of controller parameters securing the stabilization on any type of selected UPO (p-1 orbit – stable state, p-2 orbit – oscillating between two values etc.). The slight disadvantage of this approach is that for each UPO (i.e. different behavior) a different CF is needed.

The proposal of CF1 used for in the first case (p-1 orbit) is based on the following simple rule. In discrete systems, the iteration $y(n)$ and $y(n+1)$ of output value must be the same. The idea was to minimize the area created by the difference between the n and $n+1$ output iteration on the whole simulation interval $-\tau$, thus at the same time this proposal of CF should secure fast targeting into the close neighborhood of p-1 orbit and its stabilization. The CF_1 has form (4).

$$CF_1 = p + \sum_{t=0}^{\tau} |y(n+1) - y(n)| \qquad (4)$$

where: p = *penalization*

The next proposal of CF2 used for in the second case (p-2 orbit) is based on following rule. The iteration $y(n)$ and $y(n+2)$ must have the same value. But this rule is also valid for the previous case of – p-1 orbit. Thus another condition had to be added. It says that in the case of p-2 orbit there must be some difference between the n and $n+1$ output iteration. Considering the fact of minimizing the CF value this condition had to be rewritten into this suitable form (5)

$$\frac{1}{|y(n+1) - y(n)| + c} \qquad (5)$$

where: c – small constant 1.10^{-16} which was added to prevent the evolutionary optimization from crashing because upon finding the suboptimal solution stabilized at p-1 orbit it returns the division by zero.

The CF_2 has form (6).

$$CF_2 = p + \sum_{t=0}^{\tau} |y(n+2) - y(n)| + \frac{1}{|y(n+1) - y(n)| + c} \qquad (6)$$

where: p = *penalization*

The last CF used in the third case (p-4 orbit) is based on the following design. The iteration $y(n)$ and $y(n+4)$ must have the same value. But this is also valid for the two previous cases – p-1 and p-2 orbit.
Also in this case another condition had to be added. It states that in case of p-4 orbit there must be some difference between the $n+1$ and $n+3$ output iteration.

The CF_4 has form (7).

$$CF_4 = p + \sum_{t=0}^{\tau} |y(n+4) - y(n)| + \frac{1}{|y(n+3) - y(n+2)| + c} \qquad (7)$$

where: p = *penalization*

In all proposed CFs there had to be included penalization, which should avoid finding of solutions, where the stabilization on saturation boundary values -1.5, 1.5 or oscillation between them (i.e. artificial p-2 orbit) occurs. This penalization was calculated as the sum of the number of iterations, where the system output reaches the saturation boundary value.

These three "blackbox" modes CFs (6), (6) and (7) are tested in this work. The difference between them can be clearly seen in Fig. 1 - 3, which shows the dependence of CF values on the adjustable parameters K and F_{max}.
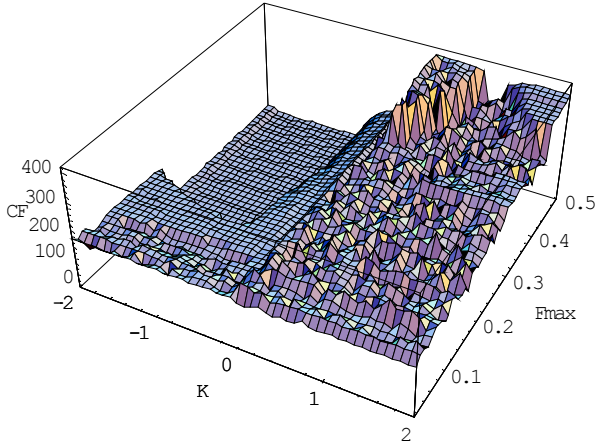
Fig. 1. Dependence of CF value on parameters K and F_{max} for p-1 orbit, $x_{initial}$ = 0.7, CF$_1$, R = 0.4426
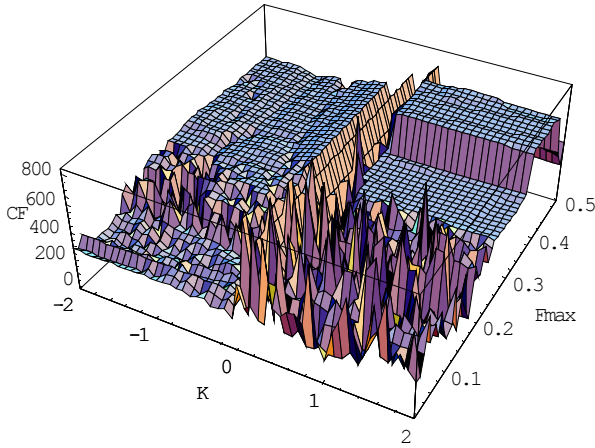
Fig. 2. Dependence of CF value on parameters K and F_{max} for p-2 orbit, $x_{initial}$ = 0.7, CF$_2$, R = 0.4387
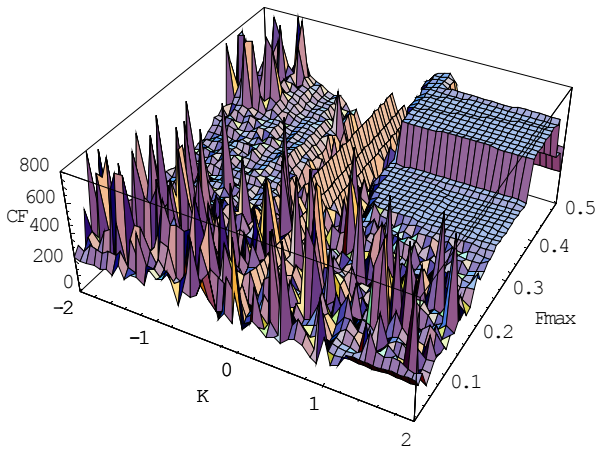
Fig. 3. Dependence of CF value on parameters K and F_{max} for p-4 orbit, $x_{initial}$ = 0.7, CF$_4$, R = 0.4574

Optimization algorithms

For the experiments described here, stochastic optimization algorithm SOMA (Zelinka, 2004), has been used. It was chosen because it has been proven that this algorithm has the ability to converge towards the global optimum.

EXPERIMENTAL RESULTS

Four versions of SOMA were used for all simulations. See Table 1 for relation between each version and index corresponding to Tables. Parameters for the optimizing algorithm were set up in such a way in order to reach the same value of maximal CF evaluations for all used versions. Each version of SOMA has been applied 50 times in order to find the actual optimum.

The primary aim here is not to show which version is better or worse but to show that the EA can in reality be used for deterministic chaos control when the "blackbox mode" cost functions are properly defined.

All results are shown only for variable x of two dimensional Henon map because of its form (1), where the variable y has the same values as variable x but it is only phase shifted.

The ranges of all estimated parameters were these:
$$-2 \leq K \leq 2 \, , \, 0 \leq F_{max} \leq 0.5 \text{ and } 0 \leq R \leq 0.5$$

Table 1: Used versions of SOMA

Index	Algorithm / Version
1	SOMA AllToOne
2	SOMA AllToRandom
3	SOMA AllToAll
4	SOMA AllToAllAdaptive

The best solution for each version of SOMA are shown in Tables 2 – 4 together with other optimization results like Average IStab value for 50 repeated simulations (Avg. IStab).

Control of chaos, p-1 orbit, CF$_1$

The first case is focused on the stabilization of p-1 orbit. From previous experiments and numerical analysis of this system, it follows that unperturbed Henon map has this p-1 orbit: x_F = 0.8

Each SOMA version gave almost identical result of CF value for the best solution. See Fig. 4 for the best individual solution with the lowest CF value (SOMA ATAA) and Fig. 5 for the simulation of this best solution with initial conditions uniformly distributed in the region of $0 < x_{initial} < 1$ See also Fig. 6 for the simulation of all 200 solutions obtained by all versions of EA. Based on obtained results, it may be stated that the control parameters estimated in the optimizations ensured fast reaching of a desired state without any

knowledge about its exact position for all 200 simulations. On average, about 90 iterations are required.

Table 2: Results for p-1 orbit, CF_1

EA	1	2	3	4
K	-0,6691	-0,6630	-0,7571	-0,6603
F_{max}	0,3172	0,3070	0,3471	0,3195
R	0,4300	0,3961	0,4824	0,4426
CF Val.	**0,3459**	**0,3459**	**0,3483**	**0,3459**
Avg. IStab	93	99	94	96
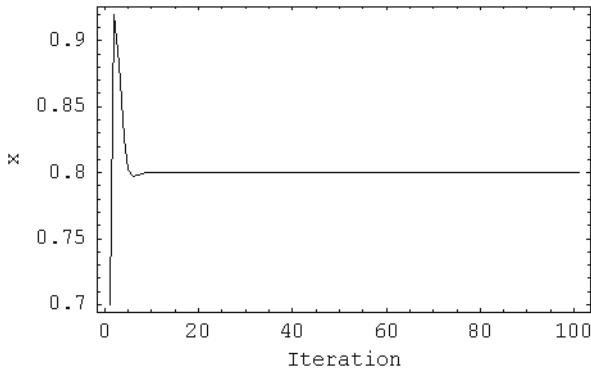
Fig. 4. Best ind. solution: p-1 orbit, CF_1, SOMA ATAA
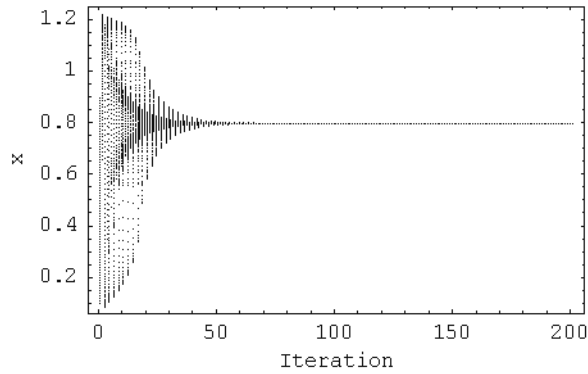
Fig. 5. Best solution: p-1 orbit, CF_1, SOMA ATAA, simulation with $0 < x_{initial} < 1$.
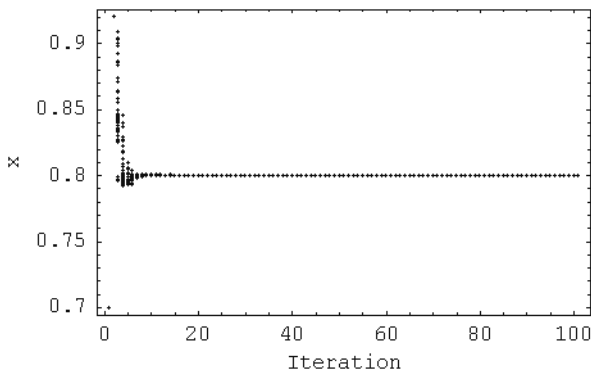
Fig. 6. Simulation of all 200 solutions, p-1 orbit, CF_1

Control of chaos, p-2 orbit, CF_2

This case is focused on the stabilization of 2p-orbit. Unperturbed Henon map has this p-2 orbit: $x_1 = -0.56$, $x_2 = 1.26$.

The best results of each SOMA version are given in Table 3. The simulations of the best solution are depicted in Fig. 7 and Fig. 8. As can be seen from Table 3, three SOMA versions give similar results of the best solution; SOMA ATR has found the lowest CF value. From comparison with previous cases and from Figs. 8 and 9 it follows that the EDTAS control method is not able to reach "exact" stabilization of p-2 orbit. This design of CF secures very fast reaching of desired state but with lower quality of stabilization. More about this problem is written in the special section. For stabilization, on average, about 130 iterations are required.

Table 3: Results for p-2 orbit, CF_2

EA	1	2	3	4
K	0,3941	-1,4353	-1,4917	-1,4823
F_{max}	0,1345	0,2209	0,2473	0,2622
R	0,1406	0,4387	0,4977	0,4932
CF Val.	**62,3066**	**48,4768**	**49,9408**	**49,8355**
Avg. IStab	137	134	137	128
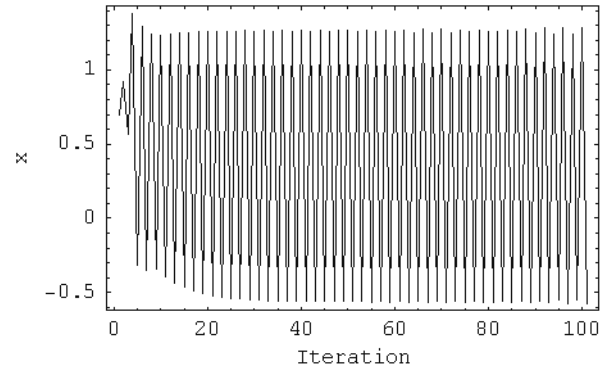
Fig. 7. Best ind. solution: p-2 orbit, CF_2, SOMA ATR
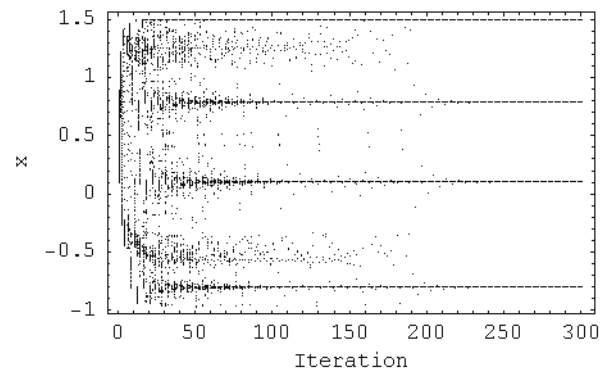
Fig. 8. Best solution: p-2 orbit, CF_2, SOMA ATR, simulation with $0 < x_{initial} < 1$.
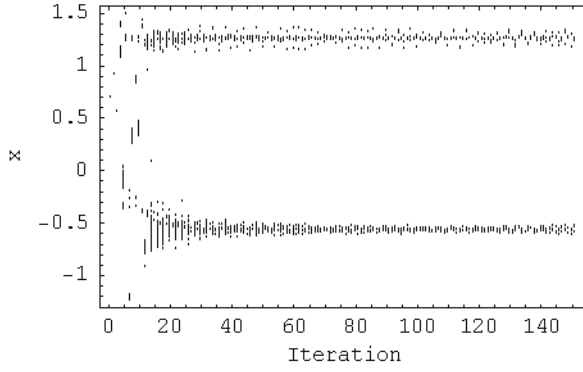
Fig. 9. Simulation of all 200 solutions, p-2 orbit, CF_2

Control of chaos, p-4 orbit, CF_4

The last case is focused on the stabilization of p-4 orbit. Unperturbed Henon map has the following p-4 orbit: $x_1 = 0.139$, $x_2 = 1.4495$, $x_3 = -0.8595$, $x_4 = 0.8962$.

See Table 4 for the results of this optimization. The simulations of the best individual solution are depicted in Figs. 10. and 11. As a conclusion of this case study it is possible to say that all SOMA versions gave almost the same results of the best solution, SOMA ATAA has found the slightly lowest value of CF. From Figs. 10 and 12 follows that the stabilization was reached very quickly and precisely. But this fact is valid only for limited circle of initial conditions (see Fig. 11). For successful stabilization of p-4 orbit, on average, about 145 iterations are needed

Table 4: Results for p-4 orbit, CF_4

EA	1	2	3	4
K	-0,3757	-0,4877	-0,3785	-0,3692
F_{max}	0,1152	0,1154	0,1145	0,1163
R	0,4617	0,5074	0,4632	0,4574
CF Val.	**19,2853**	**19,3006**	**19,2896**	**19,2836**
Avg. IStab	143	142	143	147
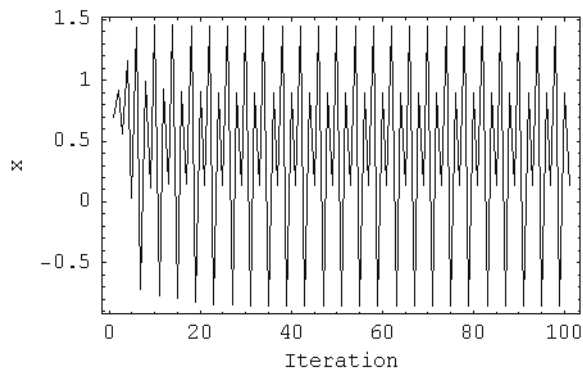
Fig. 10. Best ind. solution: p-4 orbit, CF_4, SOMA ATAA
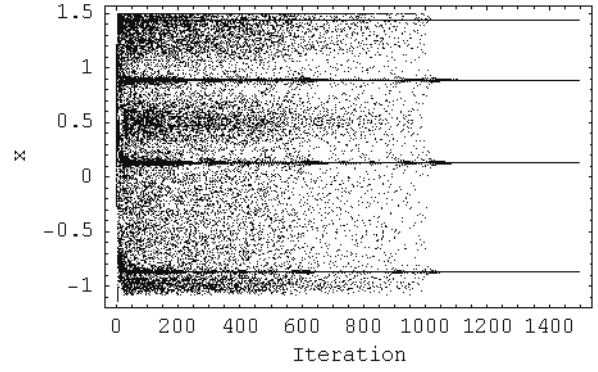
Fig. 11. Best solution: p-4 orbit, CF_4, SOMA ATAA, simulation with $0 < x_{initial} < 1$.
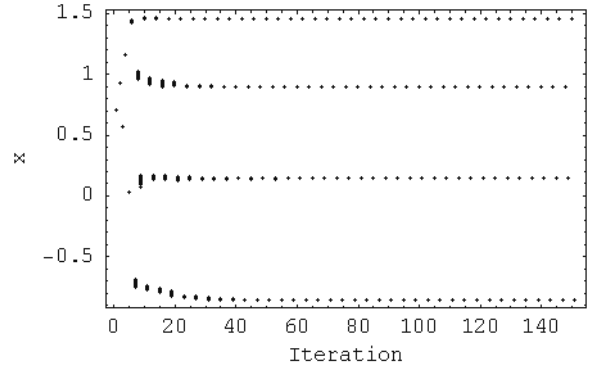
Fig. 12. Simulation of all 200 solutions, p-4 orbit, CF_4

Investigation on results of p-2 orbit stabilization

In the first and last case study, the desired p-orbit was reached rapidly and precisely, but in the second case the EA had found solutions which did not secure such precise stabilization. The answer for this problem can be clearly seen from Figs 13 – 15. The first one shows the dependence of CF value on parameter K (F_{max} = 0.2209, R = 0.4387). On the left part of this picture there is global optimum CF Value = 48.5 (see Fig, 14) and on the right positive part there are many local optimums CF value = approx. 62 (see Fig, 15). The global optimum represents solution with very fast but not precise stabilization, whereas the solution given by local optimum secures slow but very precise stabilization on desired p-2 orbit. This problem is caused by including of initial chaotic part into CF value before stabilization. The interesting fact is that version SOMA ATO did not find global optimum while in one or two cases from all 50 runs of other versions the global optimum was found. This can be solved by change of CF design and implementing the experiences with complex CF designed for higher periodic orbits from previous research.
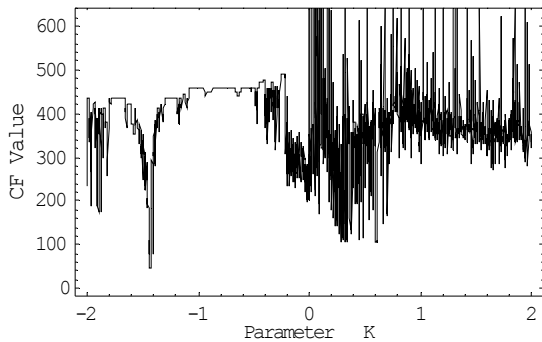
Fig. 13. Dependence of CF Value on K, p-2 orbit
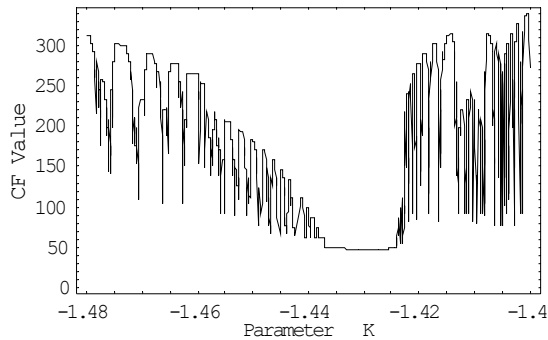
Fig. 14. Dependence of CF Value on K, p-2 orbit
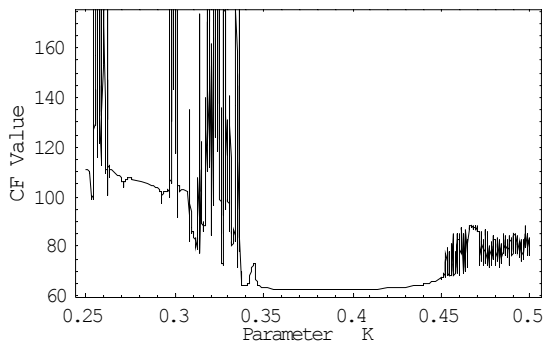
Fig. 15. Dependence of CF Value on K, p-2 orbit

CONCLUSION

Based on obtained results, it may be claimed that all simulations gave promising results and thus EAs are capable of solving this "blackbox mode" problem. In this research three different CFs were introduced and tested in the task of fast targeting and stabilization of desired periodic orbits. As can be seen from the optimization results presented here, they are extremely sensitive to the construction of the used CF. Any small change in the design of CF can cause radical improvement of system behavior, but on the other hand can cause worsening of any other observed parameter, as in the case of CF_2 for p-2 orbit.

According to all results shown here it is planned that the main activities will be focused on testing more complex cost functions and other criteria for precision distinguishing of stabilized UPO, together with searching for better settings of EA. These promising results gives the possibility for "blackbox mode" evolutionary searching for controller parameters in the

task of continuous time systems (Lorenz, Rössler), where the exact mathematical analysis of UPOs position in the chaotic attractor is very demanding.

REFERENCES

Just W., 1999, Principles of Time Delayed Feedback Control, In: Schuster H.G., *Handbook of Chaos Control*, Wiley-Vch, ISBN 3-527-29436-8.

Kwon O.-J., 1999, Targeting and stabilizing chaotic trajectories in the standard map*, Physics Letters A*, 258, 229-236.

Ott E., C. Greboki, J.A. Yorke, 1990, Controlling Chaos, *Phys. Rev. Lett*. 64, 1196-1199.

Pyragas K., 1992, Continuous control of chaos by self-controlling feedback*, Physics Letters A*, 170, 421-428.

Pyragas K., 1995, Control of chaos via extended delay feedback*, Physics Letters A*, 206, 323-330.

Richter H. and K. J. Reinschke, 2000, Optimization of local control of chaos by an evolutionary algorithm*, Physica D*, 144, 309-334.

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

Zelinka I., Senkerik R., Navratil E, 2007, *Investigation on Evolutionary Optimitazion of Chaos Control*, CHAOS, SOLITONS & FRACTALS, doi:10.1016/j.chaos.2007.07.045

Senkerik R., Zelinka I., Navratil E., 2006, Optimization of feedback control of chaos by evolutionary algorithms, in *proc 1st IFAC Conference on analysis and control of chaotic systems*, Reims, France.

Senkerik R., Zelinka I., Navratil E., 2007, Design of Targeting Cost Function for Evolutionary Optimization of chaos Control, in proc. *ECMS 2007*, Prague, Czech Republic.

AUTHORS BIOGRAPHIES
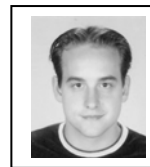
ROMAN SENKERIK was born in the Czech Republic, and went to the Tomas Bata University in Zlin, where he studied Technical Cybernetics and obtained his degree in 2004. He is now a Ph.D. student and lecturer at the same university. Email address: senkerik@fai.utb.cz

IVAN ZELINKA was born in the 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 professor (Artificial Intelligence, Theory of Information) and head of department. Email address: zelinka@fai.utb.cz .

Vision and Visualization

3D VISUALIZATION OF SIMULATION DATA IN A NETWORK CONTEXT: A CASE STUDY FROM SYSTEMS BIOLOGY

Peter Droste, Frieder Hadlich,
and Wolfgang Wiechert
Simulation Group, Inst. of Systems Engineering
University of Siegen
57068 Siegen, Germany
Email: droste@simtec.mb.uni-siegen.de

Ermir Qeli
Functional Genomics Center
Y32 H66, Winterthurerstrasse 190,
8057 Zurich, Switzerland

KEYWORDS

systems biology, 3D network visualization, isotope labeling networks, network thermodynamics, metabolic regulation

ABSTRACT

Methods for 3D visualization are well established in all disciplines where simulation preprocessing data (e.g. FE meshes) or simulation output data is directly coupled to the geometry of the real system. The situation is quite different, when this connection is missing because the system is conceptually modeled by abstract structures. The complex interconnected biochemical reaction networks in living cells which are studied in Systems Biology are a typical example. Here, it is a challenge to design visual representations of these networks showing all wanted structural features combined with quantitative data from simulation runs or experiments. This paper presents and evaluates three case studies that illustrate new visualization approaches in 3D to display structural and quantitative information related to biochemical networks and discusses the potential of interactive 3D network visualization in context of Systems Biology.

INTRODUCTION

Visualizing Simulation Data

Information visualization techniques are well established for systems with a direct spatial relation in 2D or 3D space. This does not only hold for all kinds of partial differential equation applications (continuum mechanics, fluid dynamics, electro magnetism etc.) but also to discrete systems with spatial compartmentalization (multi body dynamics, material flow, traffic etc.) (Hege and Polhier 1997). In each case powerful visualization techniques have been developed to present the simulation results and animate time dependent data. Moreover, simulation preprocessing is supported, for example by visualizing computational grids or domain decompositions. Likewise, derived simulation data like sensitivity results or eigenfunctions can be visualized.

The situation is quite different when models with a conceptual (i.e. non spatial) compartment are considered. This holds for example for chemical reaction networks where the chemical species all occupy the same space but are conceptually represented by a reac-

tion network. In this situation the visualization of simulation results by the time courses of all state variables is still predominating. However, this is not a satisfying solution because the network context is lost. Clearly, it would be more appealing if all data is directly represented in association with the network structure. A particular problem of biochemical networks here is their size, heterogeneity, and complexity of networks.

For this reason, the present contribution explores some new methods to visualize simulation related data on networks. Particularly, the potential of 3D visualization is explored. The examples stem from the field of biochemical network modeling in Systems Biology. Three case studies concerned with direct simulation data, sensitivity related information and simulation preprocessing are briefly discussed.

Biochemical Networks

Biochemical networks consist of nodes and edges (cf. Fig. 1) wheras the nodes represent chemical substances and reactions and edges represent interactions between these components (Gottschalk 1986; Michal 1999).
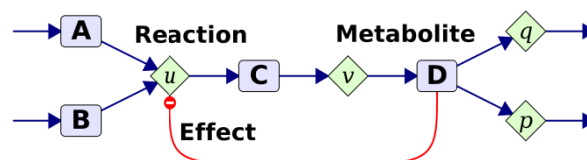
Figure 1: A typical biochemical reaction network with nodes for metabolites/reactions and edges for material flow/effector relationships (activation or inhibition).

In contrast to electrical networks, where edges always represent a flow of electrons, biochemical networks do not only contain flow edges representing chemical reactions, but also regulatory edges defining an influence of a chemical substance on a chemical reaction (Fig. 1) (Noack, Wahl et al. 2007). The direction of an edge either defines the sign of a material flow or the role of two connected nodes (source-target, effector-influenced reaction etc.).

As opposed to electrical networks for which planarity is of great technical importance, biochemical networks are usually not planar even for simple pathways. Moreover, because biochemical networks are frequently the result of systems analysis activities identifying the most im-

portant interactions in a real system, they are often enriched by a functional decomposition of the network (Kremling, Jahreis et al. 2000; Palsson 2006). This leads to the concepts of metabolic pathways, spatial compartmentation, functional units or motifs, hierarchical network decompositions, genetic operons and regulons, or networks with metabolic/genetic/signal transduction levels.
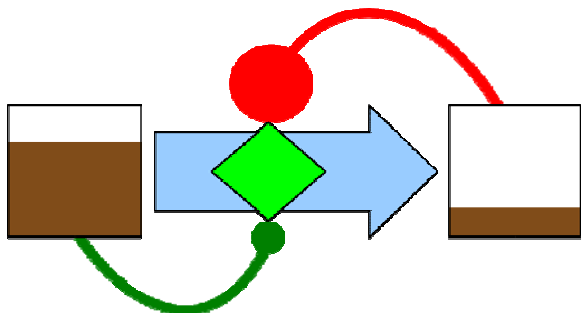
Figure 2: 2D representation of fluxes (arrow), pool sizes (partly filled rectangles) and regulatory interactions (edges with circle arrowhead) used in MetVis (Noack, Wahl et al. 2007). Sequential or time dependent data can be interactively visualized by using a slider.

Networks, which are enriched in that way by semantic structural information, demand for a graphical representation reflecting the underlying structural systems analysis. This results in the problem of designing a biochemical network in such a way that certain structural information is visually preserved in the layout. Three-dimensional (3D) network visualization is one possible way to reach this goal.

Established Visualization Methods

The aforementioned design problem becomes even stronger when biochemical networks are augmented by quantitative data from biochemical systems simulations or network wide data available from experiments. These data are associated to the network nodes and edges. A rather simple but frequently applied kind of representing quantitative data in networks is to annotate nodes and edges with numbers, bars, time courses (Junker, Klukas et al. 2006; Klamt, J. et al. 2003; Klamt, Saez-Rodriguez et al. 2007). A more direct visualization of data can be obtained by mapping numerical values or states to visual parameters of the graphical items like size, color, transparency, orientation, shape, filling etc. All these methods have been established in 2D. The reader is referred to (Qeli, Wahl et al. 2003; Rost and Kummer 2004; Sudermann and Hallett 2007) for more details.

Typically, these data are also time dependent or sequential (i.e. taken from a series of experiments). One way to visualize such serial data is to produce animations with changing graphical items over time. Interactive exploring of single time slots can be offered by using a slider as described in (Qeli, Wahl et al. 2003).

A novel development has been presented in (Noack, Wahl et al. 2007) where not only the fluxes and pool sizes but also the regulatory interaction between substance nodes and reaction nodes has been given by a quantitative measure (Fig. 2). The 2D visualization tool MetVis (Noack, Wahl et al. 2007) simultaneously visualizes metabolic fluxes, metabolite pool sizes and regulatory strength as they change over time.

3D Visualization

Because 3D representations of networks are rather seldom encountered in the literature – particularly with applications to biochemical networks – 3D visualization methods are still under development (Sudermann and Hallett 2007).

Switching to a 3D context, quantitative data associated with networks can be visualized in even more ways. However, the big question is whether this can improve rapid information perception and intuitive understanding. 3D network visualization, particularly in the biochemical network context, has not been in the main focus of the graph visualization community even in recent years (Sudermann and Hallett 2007). Several ways of 3D network representations have been suggested in the literature (Cohen, Eades et al. 1995; Dickerson, Yang et al. 2004; Frati and Battista 2007; Ho and Hong 2006; Koike 1993). Another way is the 2.5D approach (Brandes, Dwyer et al. 2003; Dwyer, Rolletschek et al. 2004) that stacks different planar data visualizations in the third dimension in order to facilitate direct comparison of time series.

Briefly, the general advantages of a 3D network representation are

- the representation of non-planar networks without edge intersections
- the option to represent modular networks and hierarchically organized structures in a more intuitive way (e.g. by top-down arrangements of subnetworks)
- the exploitation of human 3D perception capabilities to enlarge the amount of information that can be simultaneously transferred.

Some of these advantages turn into disadvantages from a technical viewpoint:

- 3D representation require a special graphics hardware (e.g. shutter glasses, caves, animation) to make the third dimension perceptible
- 3D networks loose all their expressiveness when printed on a sheet of paper
- The navigation within a complex 3D network is even more difficult than in 2D networks.

However, these are technical problems and, thus, it is a challenging problem to invent new methods and tools for 3D visualization and to compare them with their 2D counterparts. Aiming at this goal, an ongoing research project explores different concepts for 3D visualization

related to biochemical networks. The following sections present some first results and discuss central problems.

CASE STUDY 1: NETWORK THERMODYNAMICS

Foundations

Like any other process in nature biochemical reaction steps are governed by energetic principles. A reaction can only proceed in its nominal direction if the sum of the energetic reaction potentials of the reaction sources is larger than the sum of these of the reaction products. Here it should be noticed that thermodynamics does only make a statement on possible reaction directions but not on their quantitative magnitude. Even if a reaction step has a large energy gradient there might be some regulatory interaction controlling the amount of flux through this reaction step (cf. Fig. 1 and case study 2).

The free Gibbs energy potential of a chemical substance with concentration (or, more precisely, activity) x is given by

$$\Delta G = \Delta G^0 + RT \ln(x) . \qquad (1)$$

Here the term ΔG^0 is free Gibbs energy potential under standard conditions. R is the Boltzmann constant and T is temperature (Beard, Babson et al. 2004; Kümmel, Panke et al. 2006; Maskow and Stockar 2005; Qian and Beard 2005). If a chemical reaction step has sources A, B,… and targets U, V,… then it must consequently hold

$$\Delta G_A + \Delta G_B + ... > \Delta G_U + \Delta G_V + ... \qquad (2)$$

for the reaction to proceed in forward direction. The set of thermodynamic constraints obtained in this way by collecting the respective inequalities for every reaction step imposes a multidimensional relation between possible substance concentrations, free Gibbs energies under standard conditions and flux directions.

Because Eq. (2) holds both for the steady state and for dynamic transients, the display of thermodynamic potentials is an interesting complement to other simulation data like concentrations or fluxes.

3D Visualization

Using the third dimension is an appealing method to represent energy levels in a metabolic network. Here the analogy between reaction flow and hydrodynamic flow helps to understand the thermodynamic concept more intuitively.

For this purpose we developed ThermoVis, a three dimensional thermodynamic visualization tool that visualizes the energies of metabolites and reactions on a two dimensional network using the third dimension. For fast and intuitive conceiving of the network features Ther-

moVis uses color. Reactions proceeding in the feasible direction are indicated with green edges whereas reactions proceeding in the non-expected way are indicated with red edges. Furthermore the tool supports interactive navigation in the network moving, rotating and zooming in the scene and also by focusing single reaction nodes. Fig. 3 shows a three-dimensional representation of some part of central metabolism with certain measured substance concentrations.
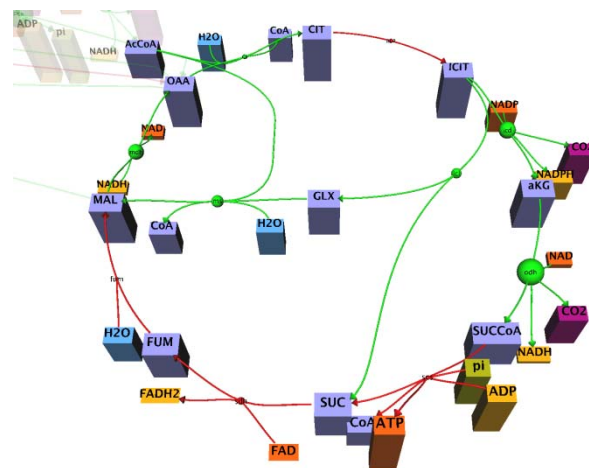
Figure 3: Snapshot of a thermodynamic visualization. Here the citric acid cycle is shown.

Many reacting metabolite pools have almost the same Gibbs energy level and, of course, are not really distinguishable. Because reactions between such pools cannot be classified in their direction ThermoVis visualizes the energy gradient of a reaction as sphere whose diameter indicates the size of this gradient (see Figs. 3 and 4). Fig. 4a shows two reactions which can be easily directed by the help of reactions energy level.
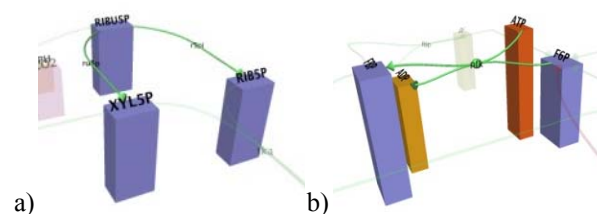
Figure 4: ThermoVis snapshots of (interesting) single reactions

Reaction steps can procees in the direction of an upward energy step when they are driven by a second downward reaction that supplies the required energy. This situation is shown in Fig. 4b.

CASE STUDY 2: SENSITIVITY RELATED DATA

Foundations

Generally, thermodynamic driving forces determine the direction of fluxes but not their magnitude. Typically, in biological networks those reaction steps with large

energy gradients are highly regulated on an enzymatic level. In the hydrodynamic analogon this corresponds to controlled valves. Several metabolite pools can have a regulatory influence on one reaction flux (Fig. 1). These influences are quite similar to sensitivities of reaction rates w.r.t. several influencing factors. For this reason regulatory dependencies are a good example for sensitivity visualization in a network context.

2D Visualization

A first approach visualizing the quantities describing the metabolic data (metabolites and fluxes) inside the metabolic network is published in (Qeli, Wahl et al. 2003), called MetVis (Metabolic Visualizer). MetVis is able to animate metabolic networks visualizations based on data generated by simulation.

To indicate the concentration level of metabolites in the network, they are drawn as variable filled rectangles like illustrated in Fig. 2. The size of the filling level is restricted to a certain range and its minimum and maximum value is understandable in every moment because they correspond to the size of the rectangle representing the metabolite. The strength of reaction flows is indicated by the width of the corresponding flux edges. Inhibition is represented with red edges and activation with green ones. The influence of an effector to a reaction is indicated by the radius of the circle-arrows (cf. Fig. 2).

3D Visualization

2D visualizations are very helpful for analyzing both topological and other simulation related properties of metabolic networks. However, 2D views have some restrictions which could possibly be avoided by using 3D visualizations. Thus, some pathways, as for example the pentose phosphate pathway, cannot be drawn in 2D without line intersections. A much more difficult problem occurs when metabolic cofactors like ATP or NADH are involved. They are coupled to almost all central metabolic reaction steps and induce a strong network connection resulting in many line crossings. In the case of metabolic networks, the crossing problem increases additionally when activation/inhibition effects are considered. The crossings issue can be treated by using 3D visualization techniques to displace edges in different planes.

In (Qeli 2007), a three dimensional approach of visualizing time series of data in metabolic network as a 3D extension of the two dimensional MetVis tool is presented. Basically, the 3D visualization is a transformation from 2D into the 3D space following certain rules. Therefore, a generated 3D network is similar to its 2D counterpart, without creating totally different layouts which do not conform to biochemical conventions. The 2D rectangle representation of metabolites in MetVis is mapped to 3D cubes and the two dimensional Bézier curves representing edges now are mapped to tubes that have the shape of a three dimensional Bezier curve and with a certain diameter. The 3D view is intended to be used as a complementary part to the 2D view, allowing the elimination of edge crossings in 2D. To reach this aim, the inner control points of the Bézier curves are displaced in different z-planes. Fig. 5 presents a comparison between a 2D network with several edge crossings and its transformation in 3D, where the intersecting edges are displaced in the z-plane.

a)   b)
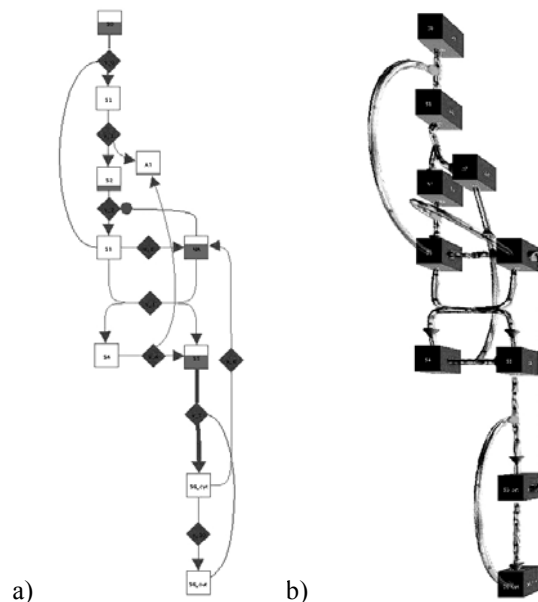
Figure 5: Comparison between 2D and 3D representation of a small metabolic network containing edge intersections.

To animate the 3D views, the raw data taken from a simulation is converted into relative percentages of the respective metabolite or flow. Two types of items are animated:

- The z-dimension of the cubes representing the metabolites varies according to the scaled concentration of the corresponding metabolite.
- Tubes represent flows for both material flux and effector influence. They are drawn semitransparent due to spheres moving within the tubes represent the material flow for the corresponding reaction. For the reaction flows, these spheres are colored with a nuance of blue, for inhibition they are colored in red and for activation they are colored in green. The speed of flow of these spheres depends on the scaled rate of the respective reaction.

Depending on the point of time, the height of cubes will show the measure of metabolite concentrations, and the velocity of spheres in the tubes indicate the reaction flow respectively the effector influence. Thus, whereas in the 2D animation one must be concentrated to view the changes in different flows, in 3D animation the speed of the movement of spheres makes it directly clear which part of the metabolic network is more active.

CASE STUDY 3: SIMULATION PREPROCESSING

Foundations

Isotope labeling experiments are an important tool to determine intracellular fluxes in a living organism. In this context not the metabolite network has to be visualized but a related network that shows the flow of ^{13}C carbon atoms or groups of such carbon atoms.

Isotope labeling networks associated to metabolic networks have an extremely high dimension. Details on these networks can be found in (Weitzel, Nöh et al. 2007). Essentially, they describe the fate of isotope labeled substances metabolized by the cell. To account for the fate of each differently labeled molecule each metabolite pool has to be divided into so-called cumomer pools (different labeled/unlabeled states).

If a metabolite has n atoms which are accessible by isotope labeling then there are 2^n different cumomers associated to this substance. These cumomers are connected by cumomer reactions which are analogous to the underlying edges in the metabolic network. Interestingly, it could be proven (Weitzel, Nöh et al. 2007) that cumomer networks can always be decomposed into smaller subnetworks. The exploitation of this decomposition has a strong impact on the performance of simulation algorithms, measurement evaluation or experimental design (Wiechert 1999).

Consequently, this is a good example for network visualization in the phase of simulation preprocessing analogous to the visualization of finite element nets in other simulation fields.

3D Visualization

Cumomer networks of even small metabolic networks can have quite a large number of nodes which makes it extremely difficult to visually disentangle the complicated structure of the network. However, this structured knowledge is important for required computing time, interpretation of simulation results or experimental design.

This immediately suggests a three dimensional representation of cumomer networks where the metabolic network is represented in a planar way on the "ground" level and the cumomers are stacked on top of the metabolite nodes using the third dimension. Fig. 6 shows the cascaded structure of a cumomer network subdivided in several levels that contain cumomers of common weight. The arrangement of nodes and edges in every level is based on the underlying planar metabolic network layout. Clearly, it is a big challenge to visualize these 3D networks in such a way that the user can recognize important structural features of the network.

The CumoVis tool has been implemented to visualize this structural information. Here, the primary challenge was to facilitate the navigation within the network, the sequential decomposition of networks and the tracing of reaction sequences. CumoVis takes a metabolic network layout and automatically arranges the cumomers on top

of each metabolite node. Moreover, CumoVis takes up the structural analysis results of a graph analysis algorithm (Weitzel, Nöh et al. 2007) that decomposes the network into the aforementioned weight levels, connected and strongly connected components.
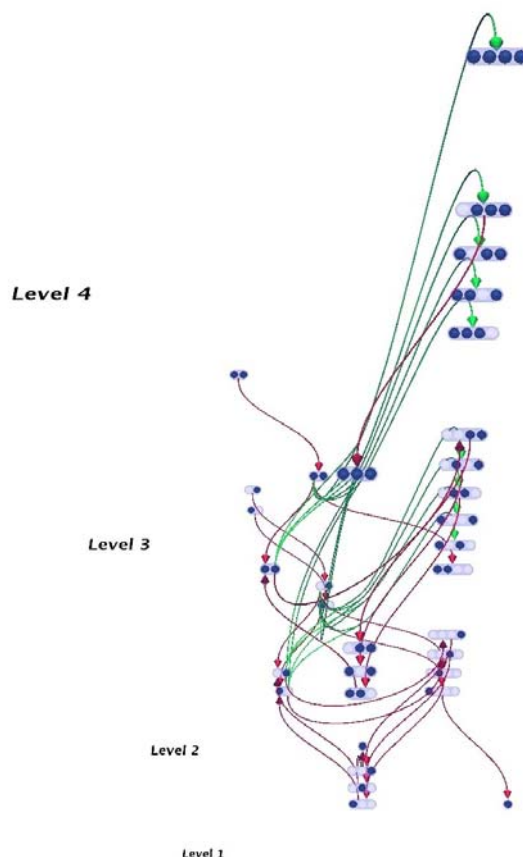
Figure 6: Cumomer visualization of a simple example network with 7 metabolites and 5 reactions.

The following interaction techniques have been implemented in the cumomers tool:

- A rough understanding of the overall complexity and structure of isotope labeling networks can be achieved by visualizing the whole network.
- More detailed impressions can be obtained by interactively rotating the network and zooming into certain parts of it. Here, navigation is made easy by the stacked network organization.
- The user can now hierarchically break up the network into smaller sub networks.
- To understand the labeling dynamics in an experiment the search for cyclic paths is very important because, essentially, these cycles make the quantitative determination of isotope enrichment in a network a non-trivial task. For this reason, cyclic structures are automatically detected and displayed on demand.
- The fate of a labeled particle can be studied by path tracing in forward and backward direction. The respective reaction paths are graphically highlighted.

This is one of the most time consuming operations that are frequently performed by experimentalists (but manually).

DISCUSSION

Experiences

Communication processes between experimentally working biologists and modelers (frequently with a mathematics or engineering background) are central activities in any interdisciplinary research project in Systems Biology. The transfer of results from systems analysis and simulation studies can be greatly facilitated by using visualization techniques. However, the investigated networks are becoming more and more complex and, consequently, are demanding for more advanced and sophisticated visualization tools. 3D representations are an option here, although they are more difficult to handle than their 2D counterparts.

Several of the tools mentioned before have already been successfully used in project meetings and teaching. Here, the built in interaction features turned out to be of great benefit because specific aspects of the networks could be explained by pointing, zooming, and highlighting. Generally, 3D network visualization makes little sense without interactive features, animation or even the use of 3D graphics facilities.

On the other hand, as also becomes clear from the figures shown above, 3D networks loose much of their expressiveness when printed on sheet of paper. This problem can be solved by publishing the visualization tool together with the paper.

Future Developments

Further work will concentrate on the 3D visualization of more complex networks involving metabolic and genetic processes. Here, constraint based 3D graph layout from given network components will also be an issue. Finally, more advanced graphics hardware (shutter glasses or a cave) will be used and evaluated.

Clearly, at the present state, the mentioned visualization tools are rather meant as prototypes to explore the potential of 3D visualization for biochemical network structure and simulation/measurement data. The greatest challenge now is to identify some general concepts which will help to unify the different approaches and still leave the full flexibility.

ACKNOWLEDGEMENTS

REFERENCES

Beard, D. A., E. Babson, E. Curtis, et al. 2004. "Thermodynamic constraints for biochemical networks." *J.Theor. Biol.* 228: 327–333.

Brandes, U., T. Dwyer and F. Schreiber. 2003. Visualizing related metabolic pathways in two and a half dimensions. *11. Intl. Symposium on Graph Drawing*, Springer LNCS.111-122.

Cohen, R. F., P. Eades, T. Lin, et al. 1995. "Three-dimensional graph drawing." *In Tamassia, Roberto and Tollis, Ioannis G., Eds. Proceedings Graph Drawing*: pages pp. 1-11.

Dickerson, J. A., Y. Yang, K. Blom, et al. 2004. Using Virtual Reality to Understand Complex Metabolic Networks. *Atlantic Symposium Comp Biol Genomic Info Systems Technol.*950-953.

Dwyer, T., H. Rolletschek and F. Schreiber. 2004. Representing experimental biological data in metabolic networks. *2nd Asia Pacific Bioinformatics Conference*

Frati, F. and G. D. Battista. 2007. "Three Dimensional Drawings of Bounded Degree Trees." *In Kaufmann, Michael and Wagner, Dorothea, Eds. Proceedings Graph Drawing*: pages pp. 89-94.

Gottschalk, G. 1986. *Bacterial Metabolism*, Cambridge University Press.

Hege, H.-C. and K. Polhier. 1997. *Visualisation and Mathematics - Experiments, Simulations and Environments*. Berlin/Heidelberg, Springer.

Ho, J. and S.-H. Hong. 2006. "Drawing Clustered Graphs in Three Dimensions." *In Healy, Patrick and Nikolov, Nikola S., Eds. Proceedings Graph Drawing*: pages pp. 492-502.

Junker, B. H., C. Klukas and F. Schreiber. 2006. "VANTED: A system for advanced data analysis and visualization in the context of biological networks." *BMC Bioinformatics* 7: 109.

Klamt, S., S. J., M. Ginkel, et al. 2003. "FluxAnalyzer: exploring structure, pathways, and flux distributions in metabolic networks on interactive flux maps." *Bioinformatics* 19(2): 261-269.

Klamt, S., J. Saez-Rodriguez and E. D. Gilles. 2007. "Structural and functional analysis of cellular networks with CellNetAnalyzer." *BMC Systems Biology* 1:2.

Koike, H. 1993. "The role of another spatial dimension in software visualization." *ACM Trans. Inf. Syst.* 11(3): 266-286.

Kremling, A., K. Jahreis, J. W. Lengeler, et al. 2000. "The organization of metabolic reaction networks: A signal-oriented approach to cellular models." *Metabolic Engineering* 2(3): 190-200.

Kümmel, A., S. Panke and M. Heinemann. 2006. "Putative regulatory sites unraveled by network-embedded thermodynamic analysis of metabolome data." *Mol. Syst. Biol.* doi: 10: 1038/msb4100074.

Maskow, T. and U. v. Stockar. 2005. "How Reliable are Thermodynamic Feasibility Statements of Biochemical Pathways?" *Biotechn. Bioeng.*: 223-30.

Michal, G. 1999. *Biochemical Pathways - An Atlas of Biochemistry and Molecular Biology*, Wiley.

Noack, S., A. Wahl, E. Qeli, et al. 2007. "Visualizing Regulatory Interactions in Metabolic Networks." *BMC Biology*: 5:46.

Palsson, B. O. 2006. *Systems Biology: Properties of Reconstructed Networks. Properties of Reconstructed Networks*. Cambridge.

Qeli, E. (2007). Information Visualization Techniques for Metabolic Engineering. Computer Science, University of Marburg.

Qeli, E., A. Wahl, D. Degenring, et al. 2003. MetVis: A tool for designing and animating metabolic networks. *Proceedings of the 2003 European Simulation and Modelling Conference*, (Naples, Italy), Eurosis Press.333-338.

Qian, H. and D. A. Beard. 2005. "Thermodynamics of stoichiometric biochemical networks in living systems far from equilibrium." *Biophysical Chemistry* 114: 213– 220.

Rost, U. and U. Kummer. 2004. "Visualisation of biochemical network simulations with SimWiz." *IEE Systems Biology* 1: 184-189.

Sudermann, M. and M. Hallett. 2007. "Tools for visually exploring biological networks." *BMC Bioinformatics* 23(20): 2651-2659.

Weitzel, M., K. Nöh and W. Wiechert. 2007. "The topology of metabolic carbon labeling networks." *BMC Bioinformatics*: 8:315.

Wiechert, W., Möllney, M., Isermann, N., Wurzel, M., de Graaf, A.A. 1999. "Bidirectional Reaction Steps in Metabolic Networks. Part III: Explicit Solution and Analysis of Isotopomer Labeling Systems." *Biotechnology and Bioengineering* 66(2): 69-85.

AUTHOR BIBLIOGRAPHIES

PETER DROSTE studied computer science with computer graphics at the University of Siegen. In 2006, he obtained his diploma degree and joined the Institute of Systems Engineering at the University of Siegen. Currently, he is working at his doctoral thesis about visualization in context of metabolic engineering and Systems Biology.

FRIEDER HADLICH studied multimedia technologies at the University of Wismar, germany, until 2005. Afterwards, he came to the University of Siegen to study in applied computer science. Since 2006, he is working at his doctoral thesis concerning the simulation, simplification and analysis of multi-dimensional omics-data in the department of modeling and simulation.
His e-mail address is:
frieder.hadlich@uni-siegen.de

ERMIR QELI studied Computer Science at the University of Tirana, Albania. From 2000 to 2007 he worked at the University of Siegen and Marburg, Germany, to reach the PhD degree. Since April 2007 he is working for FGCZ/C-MOP at the University of Zurich, Switzerland. His research interests include Information Visualization in Metabolic Engineering.
His e-mail address is:
ermir.qeli@fgcz.ethz.ch

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 Centre, where he earned his postdoctoral lecturer qualification. Since 1996, he has been professor for simulation at the Institute of Systems Engineering at the University of Siegen.
His e-mail address is:
wolfgang.wiechert@uni-siegen.de

MULTI-TECHNIQUE DATA TREATMENT FOR MULTI-SPECTRAL IMAGE VISUALISATION

Chukalina Marina
Institute of Microelectronics
Technology RAS
142432 Chernogolovka
Moscow District, Russia
E-mail: marina@ipmt-hpm.ac.ru

Andrea Somogyi
DiffAbs beamline

Synchrotron Soleil, Saint-Aubin - 48
Gif-sur-Yvette Cedex, France
E-mail: somogyi@synchrotron-soleil.fr

Nikolaev Dmitry
Institute for Information Transmission
Problems RAS
Bol'shoj Karetnyj lane, 19
101447, Moscow, Russia
E-mail: dimonstr@iitp.ru

Gerald Schaefer
School of Engineering
Aston University
Aston Triangle
Bimingham B4 7ET, U.K.
E-mail: g.schaefer@aston.ac.uk

KEYWORDS

Multi-technique datasets, microprobe techniques, mathematical model, point spread function, automatic matching, multispectral image, visualisation.

ABSTRACT

The 'in situ' investigation of different biological, chemical, physical, and environmental processes often requires the knowledge of these sample characteristics with high - (sub)micro-meter - resolution as a valuable complement to the average bulk information. Different microprobe techniques can meet these requirements and their possible combinations would offer the opportunity of thorough study of a given sample or a scientific problem.

Although some of the existing software packages allow for multivariate statistical treatment of data-sets obtained by a given experimental technique, none of them permits a simultaneous statistical treatment of datasets obtained by different experimental techniques, e.g. by simultaneous scanning μ-XRF (X-ray fluorescence) and μ-XRD (X-ray diffraction), or measured at several set-ups/beamlines/synchrotrons /laboratories (e.g. combined μ-IR and μ-XRF spectroscopy). The goal of this paper is to describe in detail a sequence of operations which can be applied to multi-technique datasets for obtaining a complete set of sample characteristics and to show that at present this can be realised automatically.

INTRODUCTION

The idea of joint analysis of results of polytypic experiments can be found in the literature (Ehmann *et al.* 1993; Bishop et al. 2004, Denecke et al. 2007). Also, commercial software products are available, which support this analysis for certain experimental techniques. Here we refer to CytoSpec as an example. This program is designed specifically for the analysis of vibrational spectroscopic (IR and Raman) imaging data sets. The software permits standard spectral manipulations, such as expansion, smoothing, scaling, normalisation, etc. Univariate methods of analysis consist of various mapping displays of hyperspectral data. The user may select band and integrated intensities, frequencies, intensity ratios, etc. to construct false color maps of the spectral data, which can be considered as slices through the hyperspectral data cube. The multivariate methods of data analysis including Principal Component Analysis (PCA), and unsupervised and supervised methods of Cluster Analysis (HCA), create spectral correlations and maps by including not only one intensity or frequency point of a spectrum, but by utilising the entire spectral information. 3D-Fourier self deconvolution (FSD) is included which allows Fourier self-deconvolution both in two spatial dimensions (x and y) and the spectral/frequency dimensions (z) to be performed at the same time. The method, adapted from the 1-D algorithm described in (Lasch and Naumann 2006), can be applied to enhance the spectral and/or spatial resolution and to increase the image contrast. Despite the numerous advantages, this software package has an essential restriction: the data series (spectra) to be evaluated should be measured with the same equipment. In this paper we describe a sequence of operations, which could be applied to the evaluation of multi-technique datasets. If this system is built from "black boxes" and each "box" performs a well-defined, independent operation, then the only task to do in order to perform the sequence of operation on the multitechnique data-set is to "connect" these boxes correctly.

The following black boxes should be included into this system: "superhigh resolution", "inverse problem", "image matching", and "multispectral visualisation" box. The question of the software interface is omitted from our consideration as we concentrate on the box content itself.

MAIN STEPS

Each measurement technique gives unique information about a sample's properties. Based on the measured signal formation, a mathematical expression is sought that relates a measured value *S(x,y)* (e.g. a number of quanta) with the sample characteristics *O(x,y)* (e.g. element concentration) for ideal experimental conditions (fine beam etc.). To solve the direct problem means to calculate *S(x,y)*, if *O(x,y)* is known. As a rule, the set-up used for data collection distorts the image by smearing the details. Consequently the so called "apparatus function" or "point spread function" (PSF) has to be taken into account and the model modified through convolution with the PSF. To solve the inverse problem means to reconstruct *O(x,y)* from a measured set of data *S(x,y)* which can be achieved in two steps. The first is known as the PSF deconvolution problem and the second is the reconstruction of *O(x,y)* by solving the equation (Chukalina and Watjen 2001). This operation should be done for each set of data. The next subsequent operation should then be automatic matching of the noisy images of the same sample in the same spectral range with different shifts, orientations and clippings. Finally, the last stage is the calculation of a final image that presents an overview of the images obtained.

Superhigh resolution

The point spread function (PSF) describes the response of an imaging system to a point source or a point object. The degree of spreading (blurring) of a point object is a measure indicating the quality of an imaging system. The point spread function may be independent of the position in the object plane, in which case it is called shift invariant. If the image plane coordinates are linearly related to the object plane coordinates and the PSF is shift-invariant, then the convolution expression describes the image formation

$$S(x_0, y_0) = \int O(x, y) PSF(x - x_0, y - y_0). \quad (1)$$

Where $S(x_0, y_0)$ is the recorded signal and $O(x, y)$ is the signal that we wish to recover. If we know the PSF, or at least know the form of PSF, we can perform deterministic deconvolution. However, if the PSD is not know in advance, then it has to be estimated. In physical measurements, the situation is usually closer to

$$S(x_0, y_0) = \int O(x, y) PSF(x - x_0, y - y_0) + \varepsilon \quad (2)$$

In this case, ε is the noise that has entered the recorded signal. If we assume a noisy signal to be noiseless when we try to make a statistical estimate of PSF, the estimate and consequently also the estimate of *O* will be incorrect. That is the reason why usual inverse signal filtering is not a good solution. However, if we have at least some knowledge of the type of noise in the data we may be able to improve the estimate of O through Superhigh Resolution Computer-Aided Measuring Systems (SHR CAMS) (Pyt'ev and Chulichkov 1998). The underlying concept of SHR CAMS is a transformation of recorded data into a form they would have if the measurement were performed by an ideal instrument. In order to achieve this, it is necessary to have a complete mathematical model of the data formation process. Since the model will contain errors caused by an approximate description of the real processes, CAMS concepts should incorporates a reliability of the model and a reliability of interpretation. The reliability of the model characterises the consistency of the model for experimental data. The reliability of interpretation characterises the possibility of estimating the object parameters with a guaranteed precision by using the model. The task "from S to O" is known as "Inverse problem. Part I".

Inverse problem. Part II

This task is reserved for data interpretation. As an example, let us consider the interpretation of m-XRF measurements. A recorded spectrum shows how many quanta of a given energy *E* were generated in a specimen volume illuminated by a microbeam. Unfortunately, the relation between the number of quanta and the concentration of a given element having this specific fluorescence line is not linear. Several factors should be included into spectrum evaluation and quantification (He and van Espen 1991). To use the data of XRF experiments jointly with diffraction data or with that of some other experimental techniques, it is necessary to know quantitative physical values such as concentration, weight fraction etc. Switching from the data recorded by a receiver to the real physical ones is known as "Inverse problem. Part II". There are several software packages developed for the XRF spectra interpretation (Espen and He. 1989, Abbott and Adams. 1997, QXAS. 2005). Thus, one image or a set of recorded images represent the input to the box and the image or a set of images, where the physical values are presented, its output.

Automatic image matching

Matching is defined as making or finding the equal or the like of (Heipke 1996). For our task, matching (or registration) can be defined as establishing a correspondence between noisy images of the same sample in the same spectral range with different lateral resolutions, shifts, orientations and clippings. Each of the mentioned factors makes the problem more difficult.

Adding to the complexity is the fact that often different spectral ranges images can have little in common in which case additional knowledge about the images, e.g. the comparison of physical models, might have to be considered.

In general there are two main approaches to image registration: landmark based and intensity based techniques (Brown 1992). While landmark based algorithms rely on the definition of stable corresponding points across the different modalities, intensity based registration utilises all available data to find the best match.

Visualization of multi-spectral images

The final images, which are to be presented to the user, have to include the most important features from the images obtained and processing in previous steps (after convertation the receiver output maps to device-independent data and after matching). An example of a powerful visualisation system is MultiSpec (http://cobweb.ecn.purdue.edu/biehl/MultiSpec) which is a processing system for interactively analysing Earth observational multispectral image data such as that produced by the Landsat series of Earth satellites. New images may be the result of principal component analysis or feature extraction transformations of existing ones, or they may result from the ratio of a linear combination of existing bands divided by a different linear combination of bands. In addition. visualisation algorithms that preserve object boundaries (Nikolaev and Karpenko 2006) will prove useful in this scenario.

CONCLUSIONS

The possibility of obtaining information about diverse sample characteristics such as morphology, crystalline structure, elemental and/or molecular composition and chemical speciation in a non-destructive manner is a general requirement of several research fields such as biology, material science, environmental science, geology, and archaeology. Different microprobe techniques can meet these requirements and their possible combinations would offer the opportunity of a thorough study of a given sample or a scientific problem.

Although existing software packages such as Cytospec (for μ-IR spectroscopy) or AXIL (for X-ray fluorescence), allow for multivariate statistical treatment of data-sets obtained by a given experimental technique, none of them permits a simultaneous statistical treatment of datasets obtained by different experimental techniques (e.g. by simultaneous scanning μ-XRF and μ-X-ray diffraction) or measured at several set-ups/beamlines/synchrotrons/laboratories (e.g. combined μ-IR and μ-XRF spectroscopy). In the paper we have argue the needs for such a system and have highlighted the problems which can arise during system development.

REFERENCES

Abbott P.H. and M. J. Adams. 1997. "AXIS: Automated XRF Interpretation of Spectra." *X-Ray Spectrometry* 26, No. 3, 125-131.

Bishop J.L., E. Muradb, M.D. Lanec and R.L. Mancinellia. 2004. "Multiple techniques for mineral identification on Mars: a study of hydrothermal rocks as potential analogues for astrobiology sites on Mars." *Icarus* 169, Issue 2 (June 2004), 311-323.

Brown L.G. 1992. "A survey of image registration techniques." *ACM Computing Surveys* 24, No. 4, 325-376.

Chukalina M. and U. Watjen. 2001. "Mathematical reconstruction of sample microstructures obtained from PIXE elemental maps." *Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms* 181, 249-253.

Denecke M.A., A. Somogyi, K. Janssens, R. Simon, K. Dardenne and U. Noseck. 2007. "Microanalysis (Micro-XRF, Micro-XANES, and Micro-XRD) of a Tertiary Sediment Using Microfocused Synchrotron Radiation." *Microscopy and Microanalysis* 13, 165-172.

Ehmann W.D., X.X. Ding, S.S. Khare, M.A. Lovell, B.F. Ni, L. Tandon, D.E. Vance and D.E. Wenstrup. 1993. "Activation analysis in a multitechnique study of trace element imbalances in age-related neurological diseases." *Journal of Radioanalytical and Nuclear Chemistry* 168, No. 1, 223-231.

Espen P.V. and F. He. 1989. AXIL: "Quantitative X-Ray Fluorescence Analysis." Technical Note, Wilrijk, Antwerp, Belgium.

He F. and P. van Espen. 1991. "General approach for quantitative energy dispersive X-ray fluorescence analysis based on fundamental parameters." *Analitical Chemistry* 63, 2237-2244.

Heipke C. 1996. "Overview of Image Matching Techniques ." Technische Universität München.

Lasch P. and D. Naumann. 2006. "Spatial Resolution in Infrared Microspectroscopic Imaging of Tissues." *Biochim Biophys Acta (BBA) - Biomembranes 2006* 1758, No. 7, 814-829.

Nikolaev D. and S. Karpenko. 2006. "Color-To-Grayscale Image Transformation Preserving The Gradient Structure." Proceedings of the 20th European Conference on Modelling and Simulation. (Bonn, Germany) 427-430.

Pyt'ev Yu. and A.I. Chulichkov. 1998. "Foundations for a theory of computer assisted superhigh resolution measurement systems." *Measurement Techniques* 41, Number 2 / February, 111-121.

QXAS. 2005. "Quantative X-ray Analysis System." IAEA.

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/.

DMITRY P. NIKOLAEV was born in Moscow, Russia. He studied physics, obtained his Master degree in 2000 and Ph.D. degree in 2004 from Moscow State University. Since 2007 he has been a head of the sector 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 and his Web page can be found at http://chat.ddt.ru/Lace/pub.html.

ANDREA SOMOGYI received her PhD and degree of Habilitation in physics at the University of Debrecen, Hungary. She was working at the ID22 hard X-ray microprobe end-station of the European Synchrotron Radiation Facility (ESRF, Grenoble, France) between 1999-2004. Recently, she is working at Synchrotron Soleil (St Aubin, France) as beamline scientist. Her interests include the development and application of X-ray Microscopy techniques in the field of environmental and material sciences.

GERALD SCHAEFER his PhD in Computer Vision from the University of East Anglia. He worked at the Colour & Imaging Institute, University of Derby as a Research Associate (1997-1999), as Senior Research Fellow at the School of Information Systems, University of East Anglia (2000-2001), and as Senior Lecturer in Computing at the School of Computing and Informaticcs at Nottingham Trent University (2001-2006). In September 2006 he joined the School of Engineering and Applied Science at Aston University. His research interests include colour image analysis, physics-based vision, image retrieval, and medical imaging. He has over 100 scientific publications in these areas.

HOUGH TRANSFORM: UNDERESTIMATED TOOL IN THE COMPUTER VISION FIELD

Dmitry P. Nikolaev
Institute for Information
Transmission Problems, RAS
127994, Moscow, Russia
E-mail: dimonstr@iitp.ru

Simon M. Karpenko
Institute for Information
Transmission Problems, RAS
127994, Moscow, Russia
E-mail: simon.karpenko@gmail.com

Ilia P. Nikolaev
Department of Physics
Moscow State University
119991, Moscow, Russia
E-mail: nikolaev@optics.ru

Petr P. Nikolayev
Institute for Information
Transmission Problems, RAS
127994, Moscow, Russia
E-mail: nikol@iitp.ru

KEYWORDS

Hough transform, fast Hough transform, SIMD-optimized algorithms, document layout analysis, vanishing point detection, ICR

ABSTRACT

We discuss Hough Transform, some of its key properties, a scheme of fast and complete calculation of the Hough Transform (similar to the Fast Fourier Transform), and an efficient implementation of this scheme on SIMD processors. We also demonstrate an application of the Fast Hough Transform in computer vision by the example of an automatic page orientation detection unit incorporated in an intelligent character recognition system. Both 2D (scanner) and 3D (camera) cases of page acquisition are considered.

INTRODUCTION

Almost every modern textbook on image processing and computer vision (Pratt 2001; Gonzalez and Woods 2002; Ballard and Brown 1982; Forsyth and Ponce 2003) pays attention to Hough Transform (HT) (Hough 1959), the transformation of summation along various lines. In those textbooks containing no "HT" reference in the index, you will most likely find the description of Radon Transform, the transformation of integration along various lines in a continuous space. So, HT is a discretization of Radon Transform. This transform is traditionally used to solve the linear regression problem in the presence of a strong spike noise or the problem of decomposition of a distribution into linear submanifolds.

In Section 1, we discuss why HT is often not considered as an efficient tool to solve computer vision problems. In Section 2, a method for fast implementation of HT on modern computer platforms is suggested. In Section 3, we demonstrate how to use this method in an industrial ICR system for document page orientation detection. In Section 4, a more complex problem of vanishing point detection in perspective text views is considered.

1. WIDESPREAD MYTHS ABOUT THE HOUGH TRANSFORM

The HT is a well-known transform widely used in image processing and machine vision (Gonzalez and Woods 2002; Forsyth and Ponce 2003). Though many results have been obtained regarding its properties, these results remain surprisingly unknown to a variety of researchers. Let us discuss those myths about the HT, which are most frequently adopted in literature.
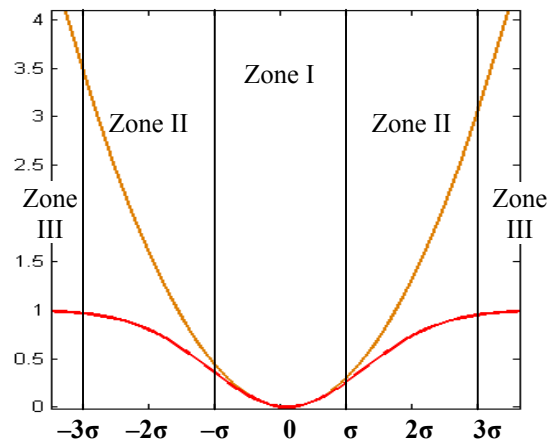
Figure 1: Quadric and Gaussian penalties

1) The HT is unstable against a normally distributed additive coordinate noise (Kiryati and Bruckstein, 2000).
This is not true. Consider the HT of data array pre-smoothed with a Gaussian filter. Evidently, the maximum in the Hough space corresponds to the line minimizing the Gaussian penalty (see Fig. 1). Such a penalty function can be conventionally split into three zones. Within Zone I, the Gaussian curve fits a parabola with a deviation not exceeding 20%, while within Zone III it is almost constant. Thus, if all the voting points belong to Zone I relatively to ideal line, solving the regression problem with the HT method yields almost the same results as obtained with the least squares method (LSM), which is known to be optimal in the

case of a normal noise. At the same time, the presence of voting points within Zone III does not affect the solution until these points form an additional line with greater effective weight (as opposed to the LSM being quite sensitive to a non-correlated spike noise – see Fig. 2a). Those nodes belonging to transient Zone II do affect the solution but weaker than those nodes belonging to central Zone I. Thus, if the smoothing parameter sigma is properly chosen, the HT method is suitable for solving the linear regression problem in the case of simultaneous presence of a normal additive and a spike coordinate noise.

Figure 2 demonstrates robustness of HT in the presence of spike noise. Fig 2a shows set of six data points (2-7) and a noise one (1). Grey line marks LSM solution. Fig 2b shows HT results for this data set, and grey line on the Fig 2c marks Hough solution.
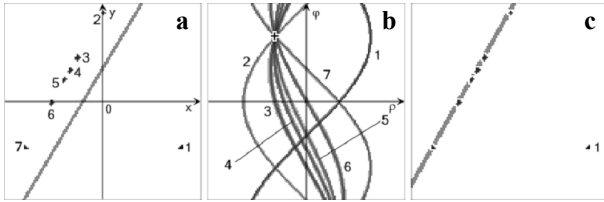
Figure 2: LSM and Hough Solutions for Linear Regression Problem

2) When applying the HT, it is important to guess the discretization scale (Forsyth and Ponce 2003).
Though this is not absolutely wrong, the problem is overstated. As it follows from the considerations of paragraph 1, the discretization scale can be fixed while the smoothing parameter is adjusted. Nevertheless, this does not erase the problem. Note that it is not too unusual for computer science that an algorithm contains a freely adjustable parameter. In such a case, it is only desirable that when there is an *a posteriori* quality criterion, the parameter adjustment process can be implemented fast enough. An iterative variation of the smoothing parameter followed by recalculation of HT does not belong to high-speed algorithms. However, it is known (Gindikin et al. 2003) that the HT is "permutable" with the Gaussian convolution! This means that after a single calculation of HT we become able to adjust the smoothing parameter in the initial space by smoothing the transformation result along ρ axis with 1D Gaussian kernel.

3) The HT is slow. Or: the HT can be accelerated only by approximate methods, including stochastic ones (Song et al. 2002).
This is not true. For instance, a method for calculating the fast HT (FHT) using the fast Fourier transform (FFT) was proposed almost 20 years ago (Lawton 1988).

4) The right way of calculating the FHT implies using the FFT (Lawton 1988).

This is not true as well. There are FHT calculation algorithms having the same asymptotic as FFT ($O(n^2 \cdot \log n)$, where n is the linear size of the array) but using only integer addition with no complex arithmetic (Brady 1998; Donoho and Hue 2000; Karpenko et al. 2004). In the next section, we shall demonstrate how to improve the complexity multiplier for one of these methods, using contemporary processor systems.

Thus, the HT can be calculated with $O(n^2 \cdot \log n)$ integer additions. To be precise, full FHT requires $4 \cdot n^2 \cdot \log n$ operations comparing to n^3 for brute-force HT. That means more than 25 times speed-up for 1024x1024 images. This allows one, provided that the smoothing parameter is properly adjusted, to suboptimally solve the linear regression problem with simultaneous presence of a normal additive and a spike coordinate noise. It is notable that the parameter adjustment iteration is reduced to a 1D Gaussian convolution of the transformation results.

2. SIMD-OPTIMIZED FHT CALCULATION ALGORITHM

Now consider a method for fast calculation of FHT. Apparently, it was first suggested by Brady (Brady 1998), and a few years later an in-place (buffer-free) version of this algorithm was published (Karpenko et al. 2004; Frederick et al. 2005).

To describe the FHT algorithm, let us introduce the following definitions. Let φ_l be the angle between the line l and the X-axis. Those lines for which $|tg\varphi_l| \leq 1$ will be called "mostly horizontal". Correspondingly, all the lines with $|ctg\varphi_l| \leq 1$ will be called "mostly vertical". Evidently, those lines parallel to quadrant diagonals belong to both classes.
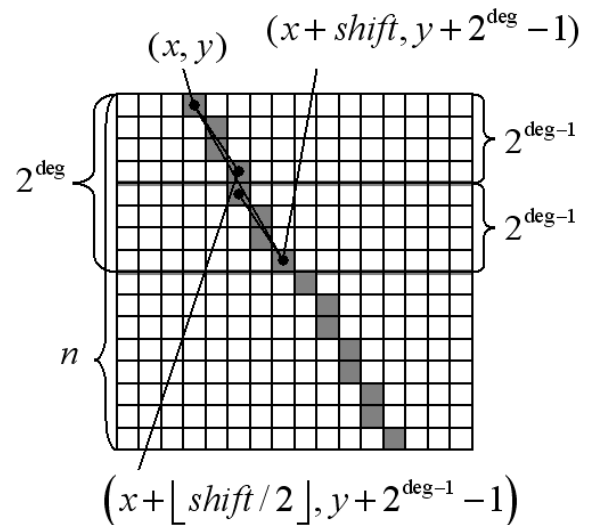
Figure 2: FHT Calculation Scheme

A discrete 8-connected line in a data array will be called an 8-connected chain of elements of this array, which approximates this line with a certain accuracy (i.e., the line passes through each element of the chain, while any 3 successive elements of the chain make an angle of at least 135˚). Examples of discrete 8-connected lines are chains built by the algorithm of Bresenham.

Consider the HT calculation algorithm for the case of "mostly vertical" discrete 8-connected lines, the case of "mostly horizontal" lines being completely analogous. Consider an input array with linear dimensions $m \times n = m \times 2^D$. A version of the algorithm, dealing with height different from a power of two, is described in the next section. Consider a horizontal strip, $m \times 2^{\deg}$ ($0 < \deg \le D$), with a vertical shift, $y = k \cdot 2^{\deg}$ ($0 \le k < 2^{D-\deg}$). Let us introduce the following recurrent definition of the 8-fold chain connecting the elements (x, y) and $(x + shift, y + 2^{\deg} - 1)$:

$$
\begin{aligned}
&\left[(x,y), (x+shift, y+2^{\deg}-1) \right] = \\
&= \left[(x,y), \left(x + \left\lfloor \frac{shift}{2} \right\rfloor, y + 2^{\deg-1} - 1 \right) \right] \cup \\
&\cup \left[\left(x + \left\lceil \frac{shift}{2} \right\rceil, y + 2^{\deg-1} \right), (x+shift, y+2^{\deg}-1) \right]
\end{aligned}
\quad (1)
$$

It is quite evident that the 8-connected chain defined by Eq. (1), is a discrete 8-fold line passing through the terminal points (Fig. 3), although it is neither Bresenham chain nor Donoho chain. Indeed, expression (1) is a method for building a line by dichotomy along the "mostly longitudinal" axis.

Instead of the classical HT parameterization (ρ, φ), we shall use the space $(x_0, shift)$, where $shift$ is the horizontal shift acquired by a "mostly vertical" line all along the array ($shift = \tan(\varphi) \cdot n$), and x_0 is the coordinate of the intersection of the parameterized line and the top row. To calculate HT, let us use Eq. (1) and fix the sign of $shift$ (not greater or not less than zero). At the iteration with number $\deg$ ($0 < \deg \le D$), "mostly vertical" HT is calculated for each of $2^{D-\deg}$ horizontal strips of height $2^{\deg}$ on the basis of the previous iteration results. Calculation of a single horizontal strip of the current iteration will be referred to as a "sub-iteration". Each sub-iteration involves into calculations two strips of the previous iteration (let us call these "sub-strips"). In each sub-strip, HT rows with $0 \le |shift| < 2^{\deg}$ are already calculated. As it follows from Eq. (1), the $|shift| = 2 \cdot s$ and $|shift| = 2 \cdot s + 1$ rows of the current strip may be calculated on the basis of the $|shift| = s$ rows of the sub-strip. Let us call this action a row operation of FHT. Since the input data of the row operation are no longer required for further calculations, the current result can be stored in-place. Unfortunately,

in such a case HT rows become "shuffled". To obtain the index of the $|shift| = s$ row in a strip of height $2^{\deg}$, it is necessary to write down the $\deg$ bits of the number s from right to left.

The described algorithm allows one to calculate HT for the chosen "primarily direction" of lines (horizontal or vertical) and the sign of $shift$, using $n \cdot \log(n)$ row operations ($4 \cdot n \cdot \log(n)$ operations for the complete HT). Now consider in more detail an elementary row operation of FHT. Let it process row A_U of the upper sub-strip and row A_D of the lower sub-strip. The simplest implementation of the FHT row operation uses a buffer array, A_T, of length $m+1$. Let us copy row A_D into the buffer array, cyclically left-shifting it by s elements. This requires two commands of block copying. The last element of the buffer array is filled in with the zero element. Then, for each $0 \le i < m$, we can calculate $A_D[i] = A_U[i] + A_T[i+1]$ and $A_U[i] = A_U[i] + A_T[i+1]$.

Now discuss a possibility of accelerating the suggested algorithm with the use of modern processor systems. Most general-purpose processors presently support the technology called SIMD (single instruction, multiple data). For x86-family processors, this is MMX (for integers), for Power PC – AltiVec (also known as Velocity Engine or VMX), for Cell/B.E. – SPE. All the above-mentioned technologies use 128-bit registers, which allows processing in parallel four 32-bit integers. (Legacy MMX instructions supports only 64-bit registers, but SSE2 technology includes MMX128 subset). However, inevitable vectorization overheads make the overall profit of this approach not very high. Using 16-bit integers is only possible when processing images with linear dimensions not exceeding 256 pixels (with 8-bit brightness). Otherwise, either saturation or overflow of a register may occur during summation.

Nevertheless, we have succeeded in increasing the degree of parallelism up to 16 operations per command with a minimum loss of accuracy. In the scheme described above, at each summation the summands are sub-sums of an equal number of elements. This feature allows us to replace the summation with averaging. All the above-mentioned SIMD technologies have corresponding commands for simultaneously averaging 16 pairs of 8-bit integers (MMX: *pagvb*, VMX: *vavgub*, SPE: *avgb*). With such a calculation scheme, we obtain, as a result, an estimate of the 8 higher bits of the exact HT result. Of course, such accuracy is not always enough. Nevertheless, this approach is worth paying careful attention to, because it demonstrates almost 10x acceleration of the HT calculation and 4x reduction of the required memory size. All the results mentioned in the next section are obtained with the use of the 8-bit calculation scheme.

3. AUTOMATIC ORIENTATION OF SCANNED DOCUMENT PAGE USING FHT

The classical image preprocessing sequence in OCR and ICR is the following: binarization, page orientation and layout analysis. However, for a mixed document flow it is often impossible to find a universal binarization algorithm. In such a case, it is desirable to perform layout analysis first. In turn, the layout analysis task becomes greatly simplified if you precisely know the internal coordinate system of the document. This fact makes it very important to robustly estimate the angle of page orientation on the basis of the original image of the page.

Consider various 1D projections of the brightness of text page pixels. It is quite evident that the projection taken along the text lines contains strongly pronounced maxima (projections of text gaps) and minima (projections of text lines). With any deviation from this particular direction, the extrema get blurred. Thus, a simple but efficient functional for detecting the optimum page orientation is the dispersion of brightness of projection pixels. Taking into account that the sum of the values contained in each row of HT is the same (and equals the brightness integrated over the image), let us reformulate the criterion: the angle of orientation of the text page is determined by the HT row showing the maximum sum of squared values.

Now consider some details of algorithm implementation. Suppose that the maximum page orientation angle is 30° (limited by the scanner used). Let us now skew the image vertically by 30° clockwise. Correspondingly, the expected range of text orientation angles is from 0° to 45° clockwise. For the image obtained, it is enough to calculate FHT within a single quadrant only: the direction is "mostly horizontal" and the shift is positive. To rescale the image width to the form 2^D, we used bilinear up-scaling of the image to the nearest power of 2. We should note that up-scaling and skewing were performed simultaneously.

The algorithm performance is illustrated in Fig. 4. Fig 4a shows source image, Fig 2b shows result of applying FHT to the skewed image and Fig 2c shows corrected image. White lines on the Fig 2b marks zero angle and angle of projection with maximal dispersion. The algorithm execution time is 60 msec for an A4 page scanned with a resolution of 200 dpi (1660 x 2340 pixels) and Pentium IV 2 GHz processor. For the sake of backward compatibility, we used FHT implementation with 8x parallelism (MMX technology, not MMX128). An additional acceleration might be obtained by preliminarily downscaling the image, but this was not supposed worthwhile. In such a form, the algorithm has been exploited for more than one year in the industrial document recognition system "Cognitive Forms", showing robust results for images of passports,

driving licenses, financial documents, and forms for surveys and tests.
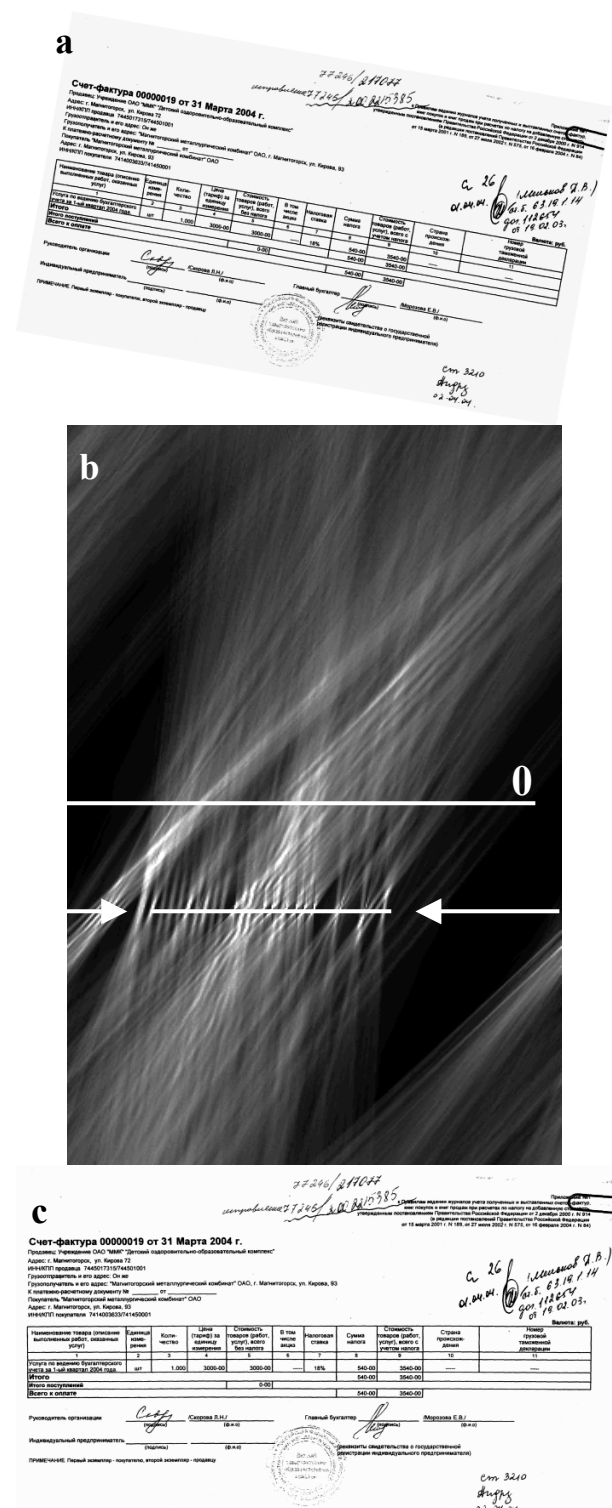
Figure 4: Automatic Page Orientation Using FHT

4. VANISHING POINT DETECTION FOR CAMERA ACQUIRED DOCUMENTS

Now consider a more complex case when the page to process is acquired in a perspective view. In this case, text strings are no longer parallel on the acquired image.

To normalize their orientation, it is necessary to detect the so-called "vanishing point". The use of HT for vanishing point detection is not a novel idea (Cantoni et al. 2001). A pencil of lines generates in the Hough space a series of maxima belonging to a 1D two-parametric manifold, such that the manifold parameters are unambiguously determined by the coordinates of the pencil center. In the space (ρ, φ), the pencil generates a sinusoid, which leads to computational difficulties on the stage of looking for the manifold that maximizes the quality functional.

a

b

c
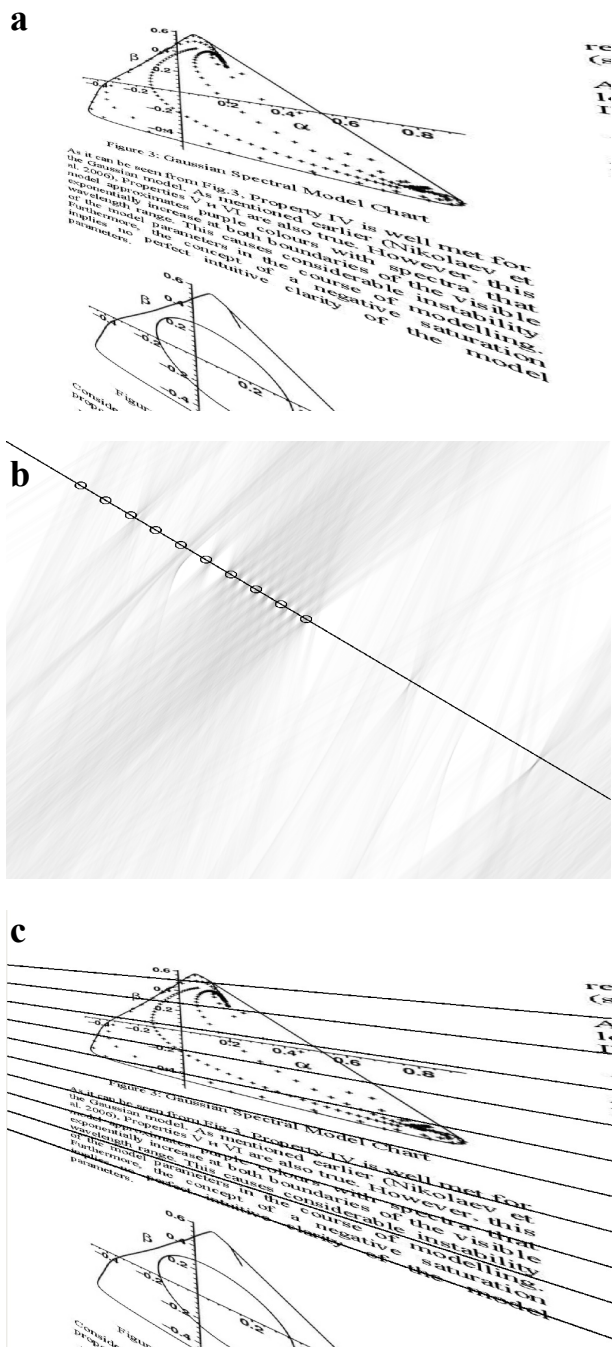
Figure 5: Vanishing Point Detection Using FHT

At the same time, in the space $(x_0, shift)$ the maxima corresponding to the lines forming the pencil lie on the same line. Using the results of the previous section, we can formulate the following criterion for searching for the vanishing point in a text page: the coordinates of the vanishing point are determined in the Hough space by the parameters of the line for which the dispersion of the values lying on it is maximum.

For lines arbitrarily oriented in the Hough space, the sum of the values is not a constant. However, numerical simulations show that the differences between such sums are negligible. We propose the following algorithm of determining the text orientation, based on using the SIMD-optimized FHT. First, apply HT to the original image. Then, calculate the value $\bar{v}$ averaged along a HT row. Then, build up a look up table (LUT) for the squared deviations:

$$LUT(v) = \min\left(255, \left[\alpha \cdot (v - \bar{v})^2\right]\right),$$ where α is a

scaling factor. Transform the result of the first HT using the LUT and perform FHT once more. The coordinates of the maximum in the resultant array unambiguously determine the coordinates of the vanishing point. The performance of the proposed algorithm is illustrated in Fig. 5. Fig 5a shows source image, Fig5b shows results of first FHT and line of maximal dispersion. Fig 5c visualizes pencil of lines with center in estimated vanishing point. As it can be seen from the figure, the dispersion maximum criterion is still efficient in the presence of pictures.

CONCLUSION

In our opinion, despite a large number of relevant papers are published, the potential of the Hough transform is not yet fully adopted in image processing and computer vision. Probably, this is due to a lack of efficient program realizations. In future, we are going to continue our research on using FHT as a tool for solving some problems of color segmentation (Nikolaev and Nikolayev, 2004) and color constancy (Finlayson and Schaefer, 2001; Nikolaev and Nikolayev, 2007).

REFERENCES

Ballard, D. and Brown, C. 1982. *Computer Vision.* Prentice-Hall.

Brady, M.L. 1998. "A fast discrete approximation algorithm for the Radon transform". In *SIAM J. Computing*, V. 27(1), 107-119.

Cantoni, V.; Lombardi, L.; Porta, M. and Sicard, N. 2001. "Vanishing point detection: representation analysis and newapproaches." In *Proceedings of 11th Int. Conference on Image Analysis and Processing*, 90-94.

Donoho, D. and Huo, X. 2000. "Beamlet pyramids: A new form of multiresolution analysis, suited for extracting lines, curves, and objects from very noisy image data". In *Proc. SPIE 2000*. V. 4119.

Finlayson, G. and Schaefer, G. 2001. "Solving for colour constancy using a constrained dichromatic reflection model". In *Int. J. Com. Vision*. V. 42, 127-144.

Forsyth, D.A. and Ponce, J. 2003. *Computer Vision: A Modern Approach.* Prentice Hall.

Frederick, M.T.; VanderHorn, N.A. and Somani, A.K. 2005. "Real-time H/W Implementation of the Approximate Discrete Radon Transform". In *Proc. of IEEE International Conference on Application-Specific Systems, Architecture Processors (ASAP'05)*. 399-404.

Gindikin, S.G.; Gelfand, I.M. and Graev, M.I. 2003. *Selected Topics in Integral Geometry.* Amer. Mathematical Society.

Gonzalez, R.C. and Woods, R.E. 2002. *Digital image processing (2nd ed.)* Prentice Hall.

Hough, P.V.C. 1959. "Machine Analysis of Bubble Chamber Pictures", In *Proc. of Int. Conf. High Energy Accelerators and Instrumentation*.

Karpenko, S.M.; Nikolaev, D.P.; Nikolayev, P.P.; Postnikov V.V. 2004. "Fast Hough transform with controllable robustness". In *Proc. of IEEE AIS'04 and CAD-2004*, V. 2, 303-309 (In Russian).

Kiryati, N and Bruckstein, A.M. 2000. "Heteroscedastic Hough Transform (HtHT): An Efficient Method for Robust Line Fitting in the 'Errors in the Variables' Problem". In *Computer Vision and Image Understanding*. V.78(1), 69-83.

Lawton, W.A 1988. "New polar Fourier transform for computer-aided tomography and spotlight synthetic aperture radar". In *IEEE Trans. Acoustics Speech Signal Process*. V. 36(6).

Nikolaev, D.P. and Nikolayev, P.P. 2004. "Linear color segmentation and its implementation". In *Comp. Vision and Image Understand.* V. 94, 115-139.

Nikolaev, D.P. and Nikolayev, P.P. 2007. "On spectral models and colour constancy clues". In *Proc. of 21st European Conference on Modelling and Simulation*. 318-323.

Nolle, L. 2007. "Application of computational intelligence to target tracking". In *Proc. of 21st European Conference on Modelling and Simulation*. 289-293.

Pratt W., 2001. *Digital image processing (3rd ed.)* Wiley.

Song, J.; Cai, M.; Lyu, M.R. and Cai, S. 2002. "A New Approach for Line Recognition in Large-size Images Using Hough Transform". In *Proc. of 16th International Conference on Pattern Recognition (ICPR'02)*. V1, 33-36.

AUTHOR BIOGRAPHIES

DMITRY P. NIKOLAEV was born in Moscow, Russia. He studied physics, obtained his Master degree in 2000 and Ph.D. degree in 2004 from Moscow State University. Since 2007 he has been a head of the sector 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 and his Web page can be found at http://www.ddt.ru/chaddt/Lace/pub.html.

SIMON M. KARPENKO was born in Moscow, Russia. He studied mathmatics, obtained his Master degree in 2003 from Moscow State University. Since 2007 he has been a research scientist at the Institute for Information Transmission Problems, RAS. His research activities are in the areas of computer vision and machine learning.

ILIA P. NIKOLAYEV was born in Moscow, Russia. He studied physics and obtained his Master degree in 1994 and Ph.D. degree in 1997 from Moscow State University. He is an associate professor at Moscow State University, his research activities are in the areas of non-linear optics.

PETR P. NIKOLAYEV was born in Moscow, Russia. He studied physics and obtained his Master degree in 1966 from Moscow State University. He obtained a Ph.D. degree in biology in 1975 from the Institute for Information Transmission Problems, RAS and a Dr. of Science degree in physics in 1993 from the 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.

Simulation Applications in Industry

SIMULATION IMPROVES MANUFACTURE AND MATERIAL HANDLING OF FORGED METAL COMPONENTS

Teresa Lang
Edward J. Williams
Onur M. Ülgen
Industrial & Manufacturing Systems Engineering Department
College of Engineering, Engineering Complex, University of Michigan - Dearborn
4901 Evergreen Road
Dearborn, MI 48128 U.S.A.
(tmlang | williame | ulgen)@umd.umich.edu

KEYWORDS

Manufacturing simulation, discrete-event process simulation.

ABSTRACT

As competitive pressures increase within the manufacturing sectors of economies worldwide, and especially within the automotive sub-sector, the importance of achieving operational efficiencies to reduce costs and thence to increase profits while keeping and attracting customers steadily increases. Simulation, time studies, and value stream mapping have long been key allies of the industrial engineer assigned to find and progress along the often difficult and challenging road leading to such efficiencies. The presentation here, and undertaken collaboratively between the university and the company involved, concentrates primarily on the use and achievements of discrete-event process simulation in improving the manufacture and material handling of forged metal components sold in the automotive and industrial manufacturing marketplace.

INTRODUCTION

Historically, the first major application area of discrete-event process simulation was the manufacturing sector of the economy (Miller and Pegden 2000). With the passage of time, simulation has become more closely allied with other industrial engineering techniques such as time and motion studies, value stream mapping, ergonomics studies, and "5S" examinations used concurrently to improve manufacturing operations (Groover 2007). Illustrative examples of simulation applications to manufacturing and industry appearing in the literature are: analysis of pig iron allocation to blast furnaces (Díaz et al. 2007), construction of a decision support system for shipbuilding (Otamendi 2005), and layout of mixed-model assembly lines for the production of diesel engines (Steringer and Prenninger 2003)

In the application documented here, simulation was applied to reduce manufacturing lead times and

inventory, increase productivity, and reduce floor space requirements. The client company was and is a provider of forged metal components to the automotive light vehicle, heavy lorry [truck], and industrial marketplace in North America. The company has six facilities in the Upper Midwest region of the United States which collectively employ over 800 workers. Of these six facilities, the one here studied in detail specializes in internally splined (having longitudinal gearlike ridges along their interior or exterior surfaces to transmit rotational motion along their axes (Parker 1994)) shafts for industrial markets. The facility also prepares steel for further processing by the other five facilities. Components supplied to the external marketplaces are generally forged metal components; i.e., compressively shaped by non-steady-state bulk deformation under high pressure and (sometimes) high temperature (El Wakil 1998). In this context, the components are "cold-forged" (forged at room temperature), which limits the amount of re-forming possible, but as compensation provides precise dimensional control and a surface finish of higher quality.

OVERVIEW OF PROCEDURES AT THE FORGING FACILITY

As mentioned, the facility examined in this study specializes in internally splined shafts for one dedicated customer in the industrial marketplace, and in steel preparation processes for two colleague plants within the same company. Therefore, this particular plant has exactly three distinct customers. The figure below shows a typical forging produced here:
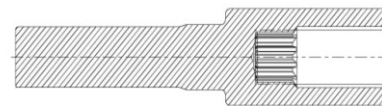
Figure 1. Typical Forging

The major production equipment used at this facility comprises:
1. Eight hydraulic presses (150-750 tons, single station, manually fed)
2. Eleven tank coating lines with five traveling tumblers

3. Two saws
4. One "wheelabrator™" (trademark name of equipment used for shot blasting)
5. Eight small and two large heat treatment areas, with five bell furnaces.

The overall layout of these operations is shown in the Appendix (Figure 4).

Having three dedicated customers, this facility produces parts in three distinct families, each with its own process routing. Parts of family #1 go first to shot blast (a cleaning process to remove surface scale and dust from the parts or billets) at the "Wheelabrator," a manually operated machine; then to lubrication at the coating line, and then to the outgoing dock for weighing and shipping. Families #2 and #3 have longer itineraries, summarized in the following tables:

Table 1. Process Routing for Production Family #2

Operation	Workcenters
Saw cutting	Saw 1 and Saw 2
Shot blasting	"Wheelabrator"
Annealing	Heat treat
Lubricating 1	Coating line
Weighing and shipping	Outgoing dock

Table 2. Process Routing for Production Family #3

Operation	Workcenters
Saw cutting	Saw 1 and Saw 2
Shot blast	"Wheelabrator"
Annealing	Heat treat
Lubricating 1	Coating line
Cold Hit 1/Inspect	390T, 490-2T, 500T
Stress relief	Heat treat
Lubricating 2	Coating line
Cold Hit 2/Inspect	150T, 490-1T, 490-2T
Final audit	Final audit
Weighing and shipping	Outgoing dock

At the saw cutting process, bar stock is received in 5-ton bundles 30 feet long. A bundle is loaded onto the saw using a crane; only then is the bundle broken open and fed into the saw. Although the saw routinely cuts every piece to an exact length (vital), it is more difficult, and equally vital, to control the weight of the billet (bar after cutting). The two saws share, and are run by, one operator.

Two varieties of heat treating are used. Spherodize annealing converts strands of carbon in the steel to spheroids before forging, rendering the steel more formable and hence capable of being forged at room temperature. Stress relieving, done after forging, relieves the stresses accumulated in the steel during forging, thereby permitting distortion-free carburizing of the internal splines. This carburizing is done at customers' sites. These two heat-treat operations share

one operator, who is responsible for loading the parts into "heat treat pots" (Figure 2 below) to be placed in the furnace and unloading the parts afterwards. Since the parts expand during heat treat, the unloading times are 50% longer and also have triple the standard deviation of the loading times.

Figure 2. Heat-treat Pot

After final heat treat, the parts are coated in a zinc-phosphate and soap lubricant; this requires that they be dumped into tumblers (Figure 3 below) which can be rotated and submerged in the lubricant, and then lifted and rotated again to drip excess solution. This work also requires operator intervention.

Figure 3. Tumbler

After lubrication, those parts destined for either of the two corporate downstream plants are ready for final inspection, weighing, and shipment thereto; the lubrication prepares them for further cold-forging there. Parts destined for the external customer are cold-forged locally subsequent to inspection, weighing, and shipment to the customer.

DATA COLLECTION AND INPUT ANALYSIS

As usual, data collection consumed a significant percentage (about 35%) of time invested in this process improvement study (Carson 2004); educators must gently explain to students that simulation studies are unlike Exercise 4 in the textbook, with "givens" such as "the machine cycle time is gamma distributed with parameters…." Much of the data collection work simultaneously supported both the value stream

mapping and the simulation analyses. Historical data on the arrival times and quantities of raw material, which occurred approximately daily at 9am by truck, was readily available. The quantities of raw material delivered were approximately normally distributed, as verified by the Anderson-Darling goodness-of-fit test available in the Minitab® statistical software package (Ryan, Joiner, and Cryer 2005) and the Input Analyzer of the Arena® simulation software. Machine cycles, such as the lubricant immersion time, the shot blast time, or the required length of heat-treat time, were well known, but operator intervention times, such as time to load or unload the heat-treat pots or the tumblers, had to be collected by traditional time-&-motion study stopwatch measurements (Mundel and Danner 1994). The stopwatches made the workers uneasy at first, raising the specter of the Hawthorne effect; data collection needed to be as quiet and unobtrusive as possible (Czech, Witkowski, and Williams 2006). Two significant aids in this data gathering were: (1) it occurred across *all* manually assisted operations – hence no *one* operator or group of operators felt threatened by special vigilance, and (2) labor-management relations at the company were and are historically favorable. Downtime frequency of occurrence, downtime duration, and scrap rate data were conveniently available from historical records, a commendable situation described vividly in (Weiss and Piłacińska 2005).

CONSTRUCTION, VERIFICATION, AND VALIDATION OF THE SIMULATION MODEL

Owing to ready availability within both academic and industrial contexts, and ample software power to both simulate and animate the production processes in question, the Arena® simulation modeling software (Kelton, Sadowski, and Sturrock 2007) was used. The animation was basic and, given the time limitations of this study, only two-dimensional, but these limitations were of little importance to the client management. Arena® provides direct access to concepts of process flow logic, queuing disciplines (e.g., FIFO), modeling of processes which may be automated, manual, or semi-automated, use of Resources (here, the various machines and their operators), definition of shift schedules, constant or variable transit times between various parts of the model, extensibility (in its Professional Edition) via user-defined modules (Bapat and Sturrock 2003), and an Input Analyzer (used as discussed in the previous section to verify distributions).

Verification and validation techniques used included a variety of methods such as tracking *one* entity through the model, initially removing all randomness from the model for easier desk-checking, structured walkthroughs among the team members, step-by-step examination of the animation, and confirming reasonableness of the preliminary results of the model with the client manager by use of Turing tests (Sargent 2004). For the "one-entity" tests, an entity of each

product type for each of the three customers was used in succession. Since the facility has maintained accurate and complete inventory data over a lengthy period of time, the inventory and work-in-process levels predicted by the model furnished an excellent "test bed" for validation. Comparison of localized performance data pertinent to each work center (e.g., machine utilization and length of queue preceding the machine) with model results was also helpful to the validation effort. Validation of the first model built – the "current operations model" was considered complete by both the analysts and the client when machine utilizations, operator utilizations, inventory levels, and throughput all correctly matched recent historical data to within 6%.

RESULTS AND OPERATIONAL CONCLUSIONS

The simulation model representing current operations was specified to be terminating, not steady-state, because this manufacturing process, unlike most, "empties itself" each night (here, at the last of three shifts) and resumes work the next day with the delivery of new raw material (Altiok and Melamed 2001). Therefore, warm-up time was always zero. Results and comparisons between the current and proposed systems were based on ten replications of the current-state model and on thirty replications of the proposed-state model (described next, and of higher intrinsic variability) each of length five working days (one typical work week). The number and duration of replications were chosen based on the helpful Arena® capability of predicting confidence interval widths for performance metrics on their standard deviations among replications run.

The initial model vividly exposed the inefficiencies in material handling already suspected of existing in the production system. Each time parts are dumped into or out of any container, they are at risk of dings and dents. The dumping that occurs in the coating line (into and out of the tumblers) is necessary – these tumblers are attached directly to the coating line, are made of stainless steel to withstand the caustic chemicals used in this operation, and have mechanisms permitting their rotation to "spin-dry" the parts as mentioned above. Therefore, the tumblers, costing about $60,000 each, represent a significant capital investment. On the other hand, the dumping into and out of containers – the heat-treat pots – seemed wasteful. Certainly the parts must be stacked in containers to be heat-treated, but the processes immediately upstream (shot blast and/or forging) and downstream (coating) from heat-treat presume the parts to be in some type of container already. Therefore, a second model was built in which these material handling operations were revised under the hypothesis that parts would be put in heat treat pots instead of other containers for all operations up to (but not including) the actual coating process. Under this new scenario, day-to-day operations would certainly need more heat-treat pots, and this second model was

used primarily to answer the question "How many more heat-treat pots would be needed to avoid excessive work-in-process inventory and delays?"

Point estimates and confidence intervals built at the 95% level, using the Student-t distribution (since population standard deviation was estimated from sample standard deviation) for the current system predicted the following:

- Mean number of heat-treat pots in use in the current system is 93 during any one work week.
- Maximum number of heat-treat pots in use in the current system at any time during any one work week is 191.
- In the proposed system (material-handling revision) the mean number of heat-treat pots in use is between 308 and 316 with 95% confidence.
- In the proposed system (material-handling revision) the maximum number of heat-treat pots in use is between 422 and 435 with 95% confidence.

Hence the simulation results were summarized for management as a recommendation to buy 225 heat-treat pots (there being currently 204 heat-treat pots on hand). The disadvantage: this recommendation entails a capital expenditure of $225,000 ($1,000 per pot). The advantages are:

1. One heat-treat dumping operator on each of the three shifts is no longer needed (annual savings $132,000).
2. Less material handling (dumping parts into and out of pots) entails less risk of quality problems (dings and dents).
3. The work to be eliminated is difficult, strenuous, and susceptible to significant ergonomic concerns.

Hence, from a financial viewpoint, the alternative investigated with this simulation study has a payback period just under 1¾ years, plus "soft" but significant benefits.

INDICATED FURTHER WORK

Further work to be investigated next via simulation involves balancing the schedule so that parts do not, as they do now, "flood" into either the heat treatment or the coating departments. The saw cuts one job at a time, and the order in which those jobs are run is discretionary. Saw cycle time is highly variable (from one to seven hours) based on the number of workpieces per box fed to a saw. Simulation may be able to prove that having all short jobs run on one saw and all long jobs run on the other saw will smooth the flow of parts downstream. If so, the gap between mean and maximum number of heat-treat pots in use can perhaps be narrowed with detriment to neither work-in-process inventory nor work-in-process time. Then the number of pots to be purchased will decrease and the payback period will likewise decrease, thereby making the

operational alternative suggested by the simulation study even more attractive.

OVERALL CONCLUSIONS AND IMPLICATIONS

Taking a longer view, the benefits of this study extend beyond the improvement of manufacturing and material handling in one facility of one moderate-sized company in the automotive sector. Publicity accorded to the study by the senior professor in charge of the simulation course (as is routinely done for many "senior projects" or "capstone projects") has drawn beneficial local attention to the ability of simulation (and by implication, other analytical methods [e.g., the value-stream mapping used here] within the discipline of industrial engineering) to help local companies increase their competitiveness. Such help is particularly pertinent to the beleaguered automotive and manufacturing industry, especially in Michigan, which is currently the 50th of the 50 United States economically (Morath 2007). Additionally, the success of this study has increased the willingness of local business and management leaders to welcome and provide project opportunities for advanced undergraduate students. This willingness stems partly from the short-term attraction of having useful industrial-engineering work done, and partly from the long-term attraction of making an investment in the experience level of students who will shortly be entering the labor market as industrial engineers (Black and Chick 1996). A student who, within the auspices of this simulation course, understands the "connection between the physical activities and the consequential financial flows" (Ståhl 2007) is well prepared to make both technically sound and financially valuable contributions at his or her place(s) of career employment.

ACKNOWLEDGMENTS

REFERENCES

Altiok, Tayfur, and Benjamin Melamed. 2001. *Simulation Modeling and Analysis with Arena*. Piscataway, New Jersey: Cyber Research, Incorporated, and Enterprise Technology Solutions, Incorporated.

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.

Black, John J., and Stephen E. Chick. 1996. Michigan Simulation User Group Panel Discussion: How Do We Educate Future Discrete Event Simulationists.

International Journal of Industrial Engineering – Applications and Practice 3(4):223-232.

Carson II, John S. 2004. Introduction to Modeling and 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, 9-16.

Czech, Matthew, Michael Witkowski, and Edward J. Williams. 2007. Simulation Improves Patient Flow and Productivity at a Dental Clinic. In *Proceedings of the 21st European Conference on Modelling and Simulation*, eds. Ivan Zelinka, Zuzana Oplatková, and Alessandra Orsoni, 25-29.

Díaz, Diego, Francisco J. Lago, M. Teresa Rodríguez, and Sergio Rodríguez. 2007. Knowledge Based Extendable Simulation of Pig Iron Allocation. In *Proceedings of the 21st European Conference on Modelling and Simulation*, eds. Ivan Zelinka, Zuzana Oplatková, and Alessandra Orsoni, 45-49.

El Wakil, Sherif D. 1998. *Processes and Design for Manufacturing*, 2nd edition. Boston, Massachusetts: PWS Publishing Company.

Groover, Mikell P. 2007. *Work Systems and the Methods, Measurement, and Management of Work*. Upper Saddle River, New Jersey: Pearson Education, Incorporated.

Kelton, W. David, Randall P. Sadowski, and David T. Sturrock. 2007. *Simulation with Arena*, 4th edition. New York, New York: The McGraw-Hill Companies, Incorporated.

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.

Morath, Eric. 2007. Auto Sales Outlook Gloomier. *The Detroit News* 134(100) [18 December 2007], pages 1A and 6A.

Mundel, Marvin E., and David L. Danner. 1994. *Motion and Time Study: Improving Productivity*, 7th edition. Englewood Cliffs, New Jersey: Prentice-Hall, Incorporated.

Otamendi, Javier. 2005. Simulation-Based Decision Support System for an Assembly Line. In *Proceedings of the 19th European Conference on Modelling and Simulation*, eds. Yuri Merkuryev, Richard Zobel, and Eugène Kerckhoffs, 335-340.

Ryan, Barbara, Brian Joiner, and Jonathon Cryer. 2005. *Minitab® Handbook*, 5th edition. Belmont, California: Thomson Learning, Incorporated.

Parker, Sybil P., editor. 1994. *McGraw-Hill Dictionary of Scientific and Technical Terms*, 5th edition. New York, New York: McGraw-Hill, Incorporated.

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.

Ståhl, Ingolf. 2007. Teaching Simulation to Business Students: Summary of 30 Years' Experience. In *Proceedings of the 2007 Winter Simulation Conference*, eds. S. G. Henderson, B. Biller, M.-H. Hsieh, J. Shortle, J. D. Tew, and R. R. Barton, 2327-2335. CD ISBN #1-4244-1306-0.

Steringer, Robert, and Johann Prenninger. 2003. Simulation of Large Standard Stillage Placement on a Diesel-Engine Assembly. In *Proceedings of the 15th European Simulation Symposium*, eds. Alexander Verbraeck and Vlatka Hlupic, 425-259.

Weiss, Zenobia, and Maria Piłacińska. 2005. Data Collection for Systems of Production Simulation. In *Proceedings of the 19th European Conference on Modelling and Simulation*, eds. Yuri Merkuryev, Richard Zobel, and Eugène Kerckhoffs, 364-369.

AUTHOR BIOGRAPHIES

TERESA LANG is a student of Industrial and Systems Engineering at the University of Michigan - Dearborn campus. She expects to be graduated with a Bachelors of Science after the winter 2008 semester. She currently holds a 3.12 overall grade-point average and a 3.48 in her engineering discipline (maximum = 4.0). She was drawn to industrial engineering due to her passion for creating efficient systems, satisfaction in creating organization out of chaos, and enjoyment of statistical analysis. She has been an employee in the forging industry for the past seven years, where she has worked as a Product and Process Engineer, Tooling Coordinator, Customer Service Engineer, and Program Manager. Currently she is working as the Quality and Engineering Coordinator / Lean Promotion Officer / TS Management Representative, responsible for development of new business, maintenance of the quality management system, and improvement of plant operations through elimination of waste and reduction of variability. She is a six-sigma black belt, a certified lead TS auditor, and certified lean champion. She specializes in cold forging and die design, statistical analysis, and program management.

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 PMC, 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 or seminars in Monterrey, México; İstanbul, Turkey; Genova, Italy; Rīga, Latvia; Göteborg, Sweden; and Jyväskylä, Finland. He has served as Program Chair of the 2004, 2005, and 2006 Summer Computer Simulation

Conferences, and also for the 2005 IIE Simulation Conference. His university web page is: http://www-personal.umd.umich.edu/~williame.

ONUR M. ÜLGEN is the president and founder of Production Modeling Corporation (PMC), a Dearborn, Michigan, based industrial engineering and software services company as well as a Professor of Industrial and Manufacturing Systems Engineering at the University of Michigan-Dearborn. He received his Ph.D. degree in Industrial Engineering from Texas Tech University in 1979. His present consulting and research interests include simulation and scheduling applications, applications of lean techniques in manufacturing and service industries, supply chain optimization, and product portfolio management. He has published or presented more that 100 papers in his consulting and research areas.

Under his leadership PMC has grown to be the largest independent productivity services company in North America in the use of industrial and operations engineering tools in an integrated fashion. PMC has successfully completed more than 3000 productivity improvement projects for different size companies including General Motors, Ford, DaimlerChrysler, Sara Lee, Johnson Controls, and Whirlpool. The scientific and professional societies of which he is a member include American Production and Inventory Control Society (APICS) and Institute of Industrial Engineers (IIE). He is also a founding member of the MSUG (Michigan Simulation User Group).

APPENDIX
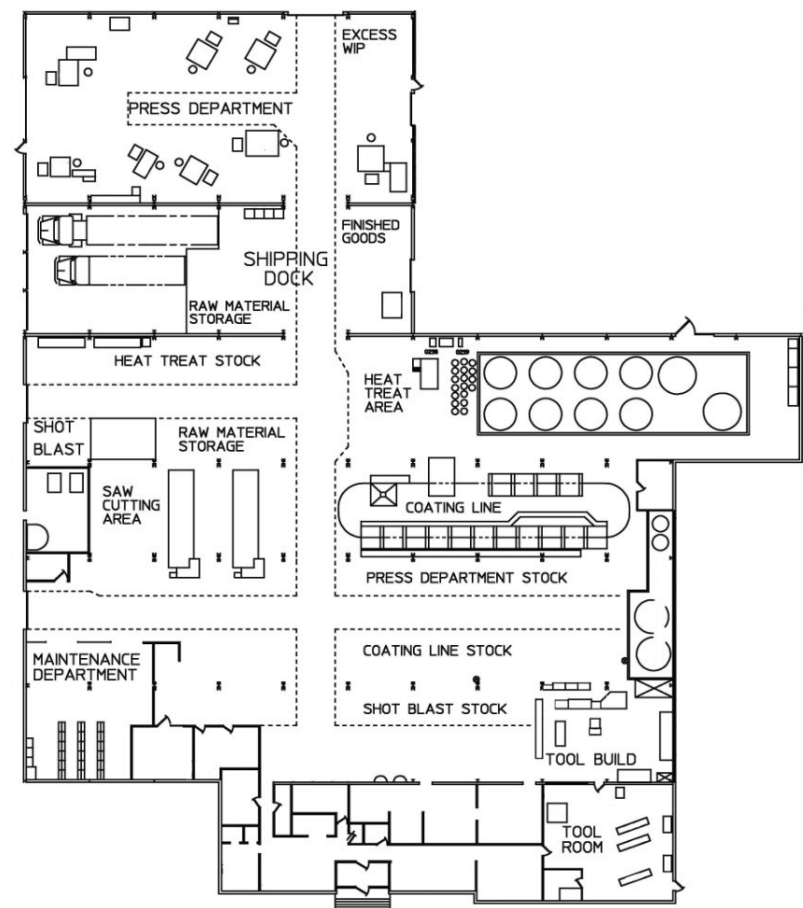
Figure 4. Current Plant Layout (Departments and Stock)

ENSURING FAULT-TOLERANCE IN GENERIC NETWORK LOCATION SERVICE

Laban Mwansa
Czech Technical University in Prague, Department of Computer Science and Engineering
Karlovo nam. 13, CZ12135 Praha 2, Czech Republic
mwansl1@fel.cvut.cz

Jan Janeček
Department of Computer Science and Engineering,
Czech Technical University in Prague
Karlovo nam. 13, CZ12135 Praha 2,
Czech Republic
janecek@cs.felk.cvut.cz

KEYWORDS

Network addressing, replication, name service, location service, DHT.

ABSTRACT

Generic Network Location Service provides a Location Service system based on DHT technology, which is storing device location records in nodes within a Chord DHT. Location records are consisting of network device identification keys as attributes, which are used to create replicas of additional location records through established hashing mechanisms.

Specifically, we first discuss the design of an alternative approach to address translations based on DHT usage then report how fault-tolerance can be ensured through replication strategies namely proactive and reactive. As a result, we show how the record is stored in more copies and each copy provides access to others, allowing us to create efficient update and recovery mechanisms. Storing collections of network device identifications has some advantages: straightforward implementation of all possible translations at the same time (provides efficient discovery mechanisms for location services which are unavailable in current systems), and fault-tolerance to DHT's node and network failures due to inherent replication of identification records.

INTRODUCTION

The current miniaturization of computer systems and recent advents in computer storage and memory technologies has brought about new dimensions in research. Notably, the continuing decline in memory prices means that huge amounts of storage is available with considerably high speeds of data transfer from primary to secondary storage. This ensures that data can be replicated synchronously and asynchronously with relatively low overheads concerning acceptable lag time in restoring replicated data. Synchronous replication in this case means that data is 100% synchronized at both the primary node and remote node whereas asynchronous replication will allow for the choice of an acceptable lag time for replicating and restoring data.

Each replication approach requires careful consideration of the infrastructure including architecture framework and intended application software. Some of the factors include the rate of recovery, recovery point objectives as well as the impact on performance of the system. This presents an opportunity for researchers to analyze efficient replication strategies for optimal system performance.

In this paper, we describe reactive and proactive replication strategies in ensuring high availability of data. The scope is limited to GNLS (Mwansa and Janecek 2007) in which the architectural framework is based on Chord DHT system (Stoica et al. 2001), essentially because of its wide acceptance and simplicity; other DHT technology might be used as well.

The contribution of this paper is an attempt to extend the GNLS system described in (Mwansa and Janecek 2007) by describing reactive replication strategies of ensuring fault-tolerance of data at various servers. PlanetSim simulator was used to obtain experimental data presented.

The following section presents some notes on DHT systems, specifically Chord followed by related work, description of the architectural framework and functions of the General Network Location Service (GNLS). Thereafter we briefly describe the importance of replication and mechanisms assuring fault-tolerance to failures. Lastly conclusion including future research perspective

Distributed Hash Table

DHT (Distributed Hash Table) systems have received considerable research interest due to their attractive properties. We can refer to the DHT's described in (Stoica et al. 2001) (Ratnasamy et al. 2001) (Balakrishnan et al. 2003). The hash table interface is an attractive foundation for a distributed lookup algorithm because it places few constraints on the structure of DHT keys and the data they refer.

DHT implements just one operation: lookup(key) yields the identity (e.g., IP address) of the node currently responsible for the given key. A simple distributed storage application might use this interface as follows: someone who wants to publish a data / file under a particular unique name would convert the name to a numeric key using an ordinary hash function, say SHA1 (Lewin 1998), then call lookup(key). The publisher would send the data / file to be stored at the resulting node. Someone wishing to read that file would obtain its name, convert it to a key, call lookup(key), and ask the resulting node for a copy of the data / file.

A complete storage system base on DHT would have to take care of replication, caching, authentication and other issues; these are outside the immediate scope of the lookup problem. To implement DHT's lookup algorithms, the following issue has to be addressed: mapping keys to nodes, preferably in a load balanced way. All DHT systems do this in essentially the same way.

Both node IDs and keys are represented by numerical identifications DHT IDs. The key is then assigned to the node with the closest node ID using a specific metric (e.g., the one that is the closest numeric successor in Chord, the one with the longest matching prefix in Kademlia, the one having the lowest distance in OpenDHT).

Chord

Chord (Stoica et al. 2001) assigns IDs from the same one dimensional ID space $< 0, N - 1 >$ to both names and nodes. The ID space wraps around to form a circle. Chord performs lookups in O(logN) time, using a finger tables of logN entries. A node's finger table contains the node ID of a node halfway around the ID space from it, a quarter-of-the-way, and so forth in powers of two (Figure 1).
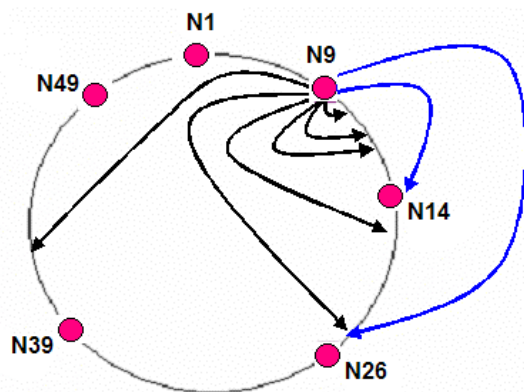
Figures 1: Chord finger table keys and node successors (N =26)

A node forwards a query for key k to the node in its finger table with the highest ID less than k. The power-of--two structure of the finger table ensures that the node can always forward the query at least half of the remaining ID space distance to k. As a result, Chord

lookups use O(logN) messages. Chord ensures correct lookups despite of node failures using the successor list: each node keeps track (of the IP addresses) of the next r nodes immediately following it in the ID space. This allows a query to make incremental progress in ID space even if many finger table entries turn out to point to crashed nodes. The only situation in which Chord cannot guarantee to find the current live successor to a key is if all r of a nodes immediate successors fail simultaneously, before the node has a chance to correct its successor list. Since node IDs are assigned randomly, the nodes in a successor list are likely to be unrelated, and thus suffer independent failures. In such a case, relatively small values of r (such as logN) make the probability of simultaneous failure vanishingly small.

Chord reconfigurations

A new node n finds its place in the Chord ring by asking any existing node to lookup n's ID. All that is required for the new node to participate correctly in lookups is to update its and its predecessor successor lists. Chord does this in a way that ensures correctness even if nodes with similar ID's join concurrently. The new node and the existing nodes, have to update their finger tables; this happens in the background because it is only required for performance, not correctness. The new node must also acquire whatever data is associated with the keys it is responsible for, the successor relationship ensures that all of these keys may be fetched from the new node's successor.

RELATED WORK

GNLS was inspired by previous work on location lookup services and name resolution technologies based on DHTs.

Applications of DHT technology for Internet name service support are numerous. As examples, we can present DDNS (Dynamic DNS) (Cox et al. 2002) based on Kademlia and Microsoft's PNRP (Peer Name Resolution Protocol) (Huitema and Miler 2002) based on Pastry and supporting IPv6 name service in Windows XP and Vista.

Another area of DHT applications is device registration and lookup in SIP (Session Initiation Protocol). Original SIP has been based on SIP servers, accessible by DNS. Current development in the frame of IETF is based on DHTs and it resulted in several propositions, an example of the SIP implementation based on OpenDHT was presented in (Singh and Schulzrinne 2006).

In (Bryan et al.2005), the authors describe a DHT overlay network based on Chord (Stoica et al. 2001) supporting SIP service. The authors discuss important issues of security and authentication, as well as adaptations of conventional P2P routing for the social networks typical in personal communications.

The authors of (Yusuke 2005) propose a name system that combines DHT and DNS, and describe how to eliminate bottleneck between the two name systems. Using the distributed nature of DHT, cost consuming processes such as protocol translation are distributed. A set of gateways that execute DNS name delegation

dynamically is used to bind between a client side DNS resolver and translators running on DHT nodes. In contrast, our work does not propose a new name translation system but seeks to complement the lookup effort of existing technologies.

(Risson et al. 2006) presents a global location service based on hierarchical DHTs. Authors propose a Location Information Plane (LIP) design based on hierarchical DHT that supports networks based on the Session Initiation Protocol (SIP).

Web based locations services have also received considerable research attention. (Quanhao et al. 2006) proposed decentralized web services discovery mechanism (DWSDM) based on a DHT to solve web originated location problems.

All these efforts are directed towards achieving efficient discovery mechanisms. (Fiedler et al. 2006) shows how peer-to-peer technologies can be integrated in load balancing and failover requirements with a centralized VoIP server concept.

GENERIC NETWORK LOCATION SERVICE (GNLS)

In this section we will briefly describe the Generic Network Location Service (GNLS) according to (Mwansa and Janecek 2007). The GNLS system consists of the servers, GNLS nodes, which provide a distributed service that allows network devices, GNLS clients, to insert, lookup, and remove location records consisting of several names, using these names as name keys.

Name keys

The GNLS system works with names that are representing network identifications / locations and treats them as byte sequences. These items are converted to DHT keys for the lookup, the most frequently used network identifications /locations are:
- Medium Access Control (MAC) address,
- Internet Protocol (IP) address,
- Domain Name (DNS),
- Electronic Numbering (ENUM),
- GSM IMEI number, and
- GPS position.

The list can be completed by including others, not so widely identification mechanisms.

Location records and Collections

The basic idea of the generic location system is storing location records, which are containing different identifications of the network device instead of purely unstructured data. The location records contain names uniquely identifying network devices mentioned: MAC addresses, IP addresses, GPS waypoint coordinates DNS names, ENUM numbers, SIP names, GSM IMEI numbers, etc. Other information can be added to location records, for example keys for authentication, which should be a vital component of any practical implementation of the location service (the issue of

corresponding security protocols is not further discussed in this material).

Suppose we have several disjoin sets S1, S2, ...Sn of names, identifying network devices in different naming systems. As location records we will denote records, consisting of names from different sets Si, e.g. $r = \{s_{1r}, s_{2r}, ..., s_{nr}\}$, where $s_{1r} \in S_1$, $s_{2r} \in S_2$, ..., $s_{nr} \in S_n$, which are identifying a single network device.

For simplicity (and in our simulation studies), we can assume that location record identifies the device in all naming systems involved. However, leaving any of the names undefined presents no problem, an example of such a realworld situation could be a personal computer connected by Ethernet to Internet, its location record will definitely involve the MAC and IP addressees (and apparently the DNS name), whereas a SIP name need not be defined, if the computer does not run SIP VoIP.

Assignment of names to devices is not static: a network device may get different IP address from DHCP after disconnection, it will definitely get different IP address when moved to different IP subnet (problem solved by Mobile IP), it will get modified GPS coordinates when moving around, a GSM phone changes its IMSI identity when we are replacing SIM. Well, there are many other situations when location information (represented by the location record) changes, the GNLS approach seems to be flexible enough to cover them all. Implementation of the dynamics presents no theoretical problem, modification of the location record can be implemented as a removal of the invalidated followed by an insertion of the modified version. However, the implementation should be optimized to increase effectiveness. More copies of the record (one identifiable by each of names involved) are stored in GNLS nodes, changing one name only in the record will result in replacement of one replica at most, others may be modified at their previous locations.

Devices having more names in any of the sets S_i (e.g. more IPv6 addressees, more DNS names, more SIP URIs) will be given more location records. Though such a solution presents no problem in theory, it could result in the explosion of number of location records in practice. (Number of location records increases with the power of numbers of names in different naming systems involved). To solve this issue we could use more complex structures, collections, in implementation of the location system.

Having disjoin sets S_1, S_2, ...S_n of names, identifying network devices in different naming systems, we will denote as collections sets of names identifying individual network devices, for example $c = \{s_{11c}, ..., s_{1m1\ c}, s_{21c}, ..., s_{nmn\ c}\}$, where $s_{1ic} \in S_1$, $s_{2ic} \in S_2$, ..., $s_{nic} \in S_n$. There is no need to limit number of names in a specific naming system (e.g. the network device with more IP addresses may have them included in a single collection) unless such a situation is, for the specific naming system, forbidden (for example a GSM phone is given a single IMEI, any physical network device will get a single GPS position, and so on). As a result, every network device can be given a single collection

involving all its names. Such a collection is valid for the specific device in a specific time instant, changing any of the names in the collection can be described as collection invalidation (removal of the collection from the location system) followed by insertion of the modified one. Well, since more replicas of the collection are stored in GNLS nodes, more elaborated set of functions will increase effectiveness.

As we mentioned in the previous paragraphs, in both approaches, based on location records and collections, we are storing several copies / replicas in DHT, each of the copies / replicas identified by the DHT key computed from one of the names and placed using this DHT key onto a corresponding DHT node (the successor of the DHT key in Chord). There is an interesting issue to estimate how many nodes will be used to store copies / replicas of the location record (or collection). Although the probability that location record involving n names will be placed in m different nodes (m <= n) can be evaluated analytically, we will present here only the results we have got from the simulation of the Chord DHT consisting from 10 to 10000 nodes (Figure 2). Probability of placement more copies / replicas onto a single node is acceptable even for tens of GNLS servers. Making use of this fact in the GNLS server's implementation provides acceptable fault-tolerance, even without data replication support at the DHT layer.
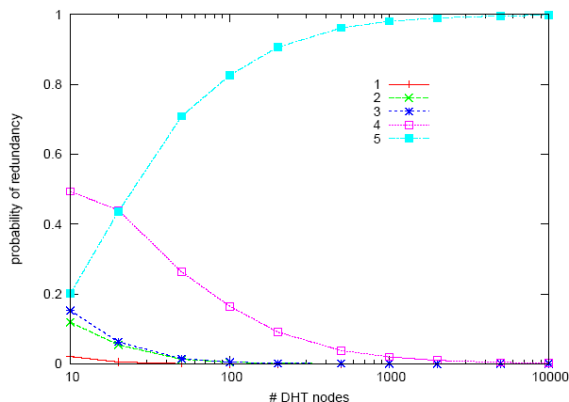
Figures 2: Probability of redundancy factor

The results indicate that, having sufficient number of GNLS nodes, the intrinsic redundancy is sufficient and probability of extreme concentration of replicas on a one or two nodes (well, excluding binary relations) can be vanishingly low. However, when required, location conflicts, defined as two or more replicas identified by different DHT keys placed at the same DHT node, can be solved by moving "overflowing" replica(s) to subsequent DHT node(s).

GNLS Architecture

The GNLS system consists of the servers, GNLS nodes, which provide a distributed service that allows network devices, GNLS clients, to insert, lookup, and remove location records. Whereas servers are forming relatively stable infrastructure and can run DHT mechanisms

effectively, clients are generally connected only for shorter time and may change their connection to GNLS servers. Restricting DHT mechanisms to the stable core of GNLS servers decreases overhead of DHT stabilization, which has to manage only node failures and infrequent core reconfigurations.

Though DHT technology promises effective lookup, enabling servers of the infrastructure and clients with caches would be necessary to get a service quality comparable to DNS. (Remember that GNLS provides basis for name translation unavailable in any current specific translation mechanism). Simple time constrained caches can be used, but more flexible distribution of change reports can be implemented as well at both levels, GNLS servers and clients.

Another technical problem is related to accessibility of GNLS. The simplest mechanism can be based, similarly to SIP, on= the standard DNS service, i.e. the client will need only a single DNS name to get access to one or more GNLS servers.

Generic Network Location Functions

Though GNLS could provide more complex services, the basis for them is the interface to location record storage. Location records are inserted into GNLS system by the insert method that has a form:

insert (location record) , where location record is a data structure containing information on types and values of involved name keys (MAC and IP addresses, DNS names, ENUM numbers, IMEI identifiers, etc.)(RFC 3761).

The method is invoked by remote GNLS clients. The GNLS node converts individual name keys of the location record into DHT keys. Then it stores the copies of the location record in GNLS nodes designated by the Chord lookup applied to individual DHT keys.

A counterpart of the insert method is the get method that has the form location record = **get**(name) , where name is a data structure representing type and value of the selected name key. The method is invoked remotely from GNLS clients. The GNLS node converts the name key into the DHT key and finds, using the Chord's lookup function, the node, which stores the required location record. It asks DHT for the record and receiving it, the GNLS node passes the record to the GNLS client and stores it into its cache for an eventual future use.

Finally, location record can be removed by the method **remove**(name) , which takes any of the name keys as the parameter name. The GNLS node, which is addressed by the remove request gets a corresponding copy of the location record and uses contained name keys to locate and remove all location record replicas from DHT.

The simple interface just described covers functionality of the GNLS system. However, real world implementation should include the modify method, for example in the form **modify**(location record) , which will remove, insert and locally modify individual

replicas of the location record on the basis of the previously stored record and the modification request.

To illustrate function of the GNLS system we will present an example describing insertion of location records for two devices in the Chord DHT which is using 6bit ID space (the real world Chord uses 160bit IDs). Location records of the devices includes MAC address, IP address, fully specified DNS name, phone number in ENUM format and GSM identification, and are presented in Figure 3.

Location record 1

Key	Value	Hash
MAC	00-D0-8F-59-7C-4B	3
IP	10.0.4.6	10
DNS	CS.FELK.CVUT.CZ	15
ENUM	7.4.1.0.6.4.9.7.0.2.4.4.e164.arpa	28
GSM BPS	284011234567890	50

Location record 2

Key	Value	Hash
MAC	00-BB-6E-59-7D-4B	30
IP	128.0.3.5	51
DNS	CS.PUMBA.MSU.ZA	44
ENUM	5.3.1.0.6.4.9.7.0.2.4.4.e164.arpa	30
GSM BPS	386711234567890	11

Figures 3: Location Records

Distribution of location records replicas onto GNLS nodes is presented in Figure 4.

KEY

▲ Location record device 1.
● Location record device 2.
■ Chord DHT node
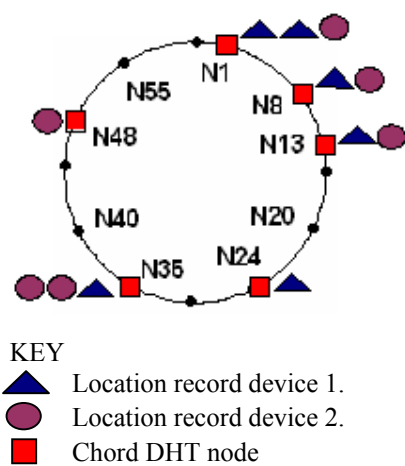
Figures 4 Distribution of location records

Nodes in Chord are distributed randomly over the DHT ID space, the consequences of this fact are uneven distances between nodes and therefore uneven number of records they store. The distributions we observed on the model of GNLS in PlanetSim are presented in the following graphs. Figure 5 depicts the probability density function and Figure 6 the cumulative probability, both for the number of collections of five names stored at GNLS nodes. Results are presented for sets of 100, 1000 and 10000 randomly distributed GNLS nodes and one million of stored collections. Results are normalized with respect to average number of collections at the nodes.
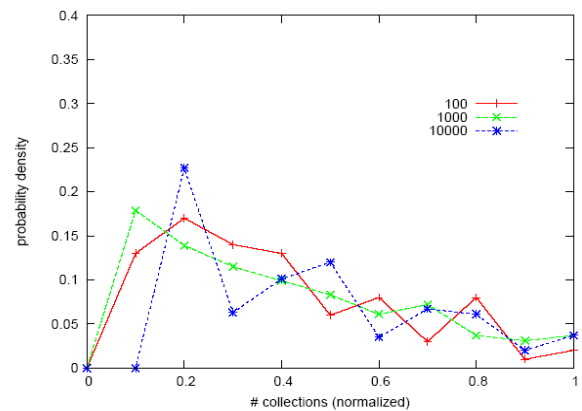
Figures 5 Distribution of collections – density
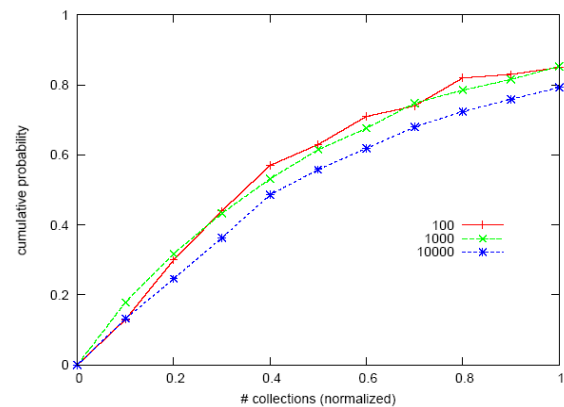
Figures 6 Distribution of collections - cumulative

FAULT-TOLERANCE

The GNLS system decreases sensitivity to failures by replication of individual location records. Every location record containing n location IDs is stored in n copies in GNLS.

Although intrinsic replication of location records and their distribution among more nodes itself decreases negative effects of concentration in classical systems (e.g. failure of a DNS server and its secondary replica can disconnect the service area), we need a mechanism able to recover from damages resulting from individual node failures.

Moreover, the DHT systems themselves include mechanisms, which assure correct DHT functioning after a node failure occurs.

In the following paragraph we propose our solution for replication maintenance problem in GNLS and possibilities of data reconstruction after a node failure. We use an illustrative example in figure 4 using two location records. The figure below shows the placement of the replicas as well as the Chord fingers tables.

Figures 7

Finger table at Node1

k	interval	succ.
1	[2,3]	8
2	[3,5]	8
3	[5,9]	13
4	[9,17]	24
5	[17,33]	35
6	[33,1]	1

Finger table at Node 8

k	interval	succ.
1	[9,10]	13
2	[10,12]	13
3	[12,16]	24
4	[16,24]	24
5	[24,40]	48
6	[40,8]	8

Finger table at Node 48

k	interval	succ.
1	[49,50]	1
2	[50,52]	1
3	[52,56]	1
4	[56,1]	1
5	[1,17]	24
6	[17,48]	1

Finger table at Node 35

k	interval	succ.
1	[36,37]	48
2	[37,39]	48
3	[39,43]	48
4	[43,51]	1
5	[51,4]	8
6	[4,35]	35

Finger table at Node 24

k	interval	succ.
1	[25,26]	35
2	[26,28]	35
3	[28,32]	35
4	[32,40]	48
5	[40,56]	1
6	[56,24]	24

Finger table at Node 13

k	interval	succ.
1	[14,15]	24
2	[15,17]	24
3	[17,21]	24
4	[21,29]	35
5	[29,45]	48
6	[45,13]	13

KEY

▲ Location record device 1.
● Location record device 2.
■ Chord DHT node

Figures 7: Distribution of location records

We illustrate distribution of location records onto GNLS nodes together with theirs finger tables (Figure 7).

Replication Strategies

Though DHT systems is equipped by additional mechanisms, which are able to avoid loss of stored data GNLS system provides additional possibilities to fault-tolerance of stored data from node failures. Proactive strategy entails mechanism able to assure high availability of location records in GNLS system is to replicate them at some node, which takes on an original node's function after the failure.

A drawback of the proactive strategy is that it almost doubles (for higher number of nodes) memory requirements. Reactive strategy employs mechanism that exploits multiplicity of location records in GNLS reacts to the detected node failure as follows: Together with starting stabilization mechanism (i.e. an update of node links and finger tables) lookup request is broadcast using the finger tables' tree. For a single node failure, the worst case analysis shows that in at most O (2.logN) steps all replicas located out of the failing node will be found. That means, the lookup request can return the result, and, since the failing node is identified, lost replicas can be reproduced at the failing node surrogate. The advantage of the reactive strategy to replication is that, it benefits from essence of location records: multiple copies distributed among more nodes, i.e. the mechanism needs no extra memory.

The risk of placing all copies to a single failing node is acceptably low for higher number of DHT nodes. Moreover, such a situation can be simply detected and the additional copy of the location record can be created elsewhere, e.g. at the node's successor. Any node may be used as a place for such an additional copy, linkage of this backup copy (found by lookup broadcast in the case of failure detected during regular lookup) to the true location record can be maintained without difficulties. The influence of multiple failures is lower

than in proactive methods assuming the number of fields in location records is greater then number of copies in proactive methods (including the original).

Replica reconstruction after node failure

Figure 8 illustrates the distribution of location records in the GNLS after node 1 has been removed. The removal of the node from the DHT ring network maybe as a result of a node failure or voluntary withdraws.

Once a node having a replica placement is detected as having failed, a replica reconstruction mechanism is triggered. If a node does not respond for sometime, a time out situation occurs and it would be concluded that the node is either dead or unavailable.

One technique to solve this problem of replica reconstruction after node failure, is to reconstruct additional copies of location records using hashing mechanism as a replacement copy. Figure 8 shows the contents of the Chord network after reconstruction. The location record replicas will be placed on new node id's after a network stabilization protocol is run.

Finger table at Node 48

k	interval	succ.
1	[49,50]	8
2	[50,52]	8
3	[52,56]	8
4	[56,1]	8
5	[1,17]	24
6	[17,48]	8

Finger table at Node 8

k	interval	succ.
1	[9,10]	20
2	[10,12]	20
3	[12,16]	20
4	[16,24]	35
5	[24,40]	48
6	[40,8]	20

Finger table at Node 35

k	interval	succ.
1	[36,37]	48
2	[37,39]	48
3	[39,43]	48
4	[43,51]	8
5	[51,4]	8
6	[4,35]	48

Finger table at Node 24

k	interval	succ.
1	[25,26]	35
2	[26,28]	35
3	[28,32]	35
4	[32,40]	48
5	[40,56]	8
6	[56,24]	35

Finger table at Node 13

k	interval	succ.
1	[14,15]	24
2	[15,17]	24
3	[17,21]	24
4	[21,29]	35
5	[29,45]	48
6	[45,13]	24

KEY

▲ Location record device 1.
● Location record device 2.
■ Chord DHT node

Figures 8: Distribution of location records and content of finger tables after node 1 has been removed

CONCLUSION

The paper presents the idea of storing location records, defined as collections of different location identifications (e.g. MAC address, IP address, DNS name, E.164 number, GSM BTS identification) of a single device, in a DHT overlay system. Replication strategies are discussed ensuring the availability of data, to clients

Storing location records in places "addressable" (using the DHT lookup) by individual location record fields provides a simple way to implementation of translation functions similar to well-known network services (e.g. ARP, DNS, ENUM). Storing records consisting of several location identifications in a DHT space benefits from a $O(logN)$ complexity of the lookup.

Fault-tolerance to failures can be based on stored replication records without creating additional

secondary copies (up to some very infrequent cases mentioned). The GNLS proposed in the paper is not supposed to be a substitution of the existing translation techniques (e.g. ARP, DNS, ENUM), but it is considered as an overlay that uses data available in existing systems and provides some translations currently unavailable. Future research challenges include investigating the cost and latencies of messages exchanges involved in replica reconstruction after node failure.

REFERENCES

Bryan, D.A., Lowekamp B.B., Jennings "C.: SOSIMPLE: A Serverless, Standards based, P2P SIP Communication System. In: Advanced Architectures and Algorithms for Internet Delivery and Applications*", AAAIDEA* 2005, 42-49

Cohen E., Shenker S. "Replication strategies in unstructured Peer-to-Peer Networks." In: *Proceedings of the ACM Sigcomm02 Conference*, August 2002.

Cox R., Muthitacharoen A., Morris R."Serving DNS Using a Peer-to-Peer Lookup Service", *Revised Papers from the First International Workshop on PeertoPeer Systems,* LNCS 2429, Springer 2002, 155-165.

Yusuke Doi: "DNS Meets DHT: Treating Massive ID Resolution Using DNS Over DHT", *The 2005 Symposium on Applications and the Internet (SAINT'05)*, 2005.

Fiedler J., Kupka T., Magedanz T, Kleis M. "Reliable VoIP Services Using a PeertoPeer Intranet", *Eighth IEEE International Symposium on Multimedia (ISM'06)*, Dec. 2006, 121-130.

Balakrishnan H., Kaashoek M.F., Karger D., Morris R., Stoica I. "Looking up Data in P2P systems", *Technical and social components of peer-to-peer computing*, 2003, 43-48.

Huitema Ch., Miller J. L. "Peer-to-peer name resolution protocol (pnrp) and multilevel cache for use therewith", *EP1248441 Patent, Microsoft*, 2002.

RFC 3761, "The E.164 to Uniform Resource Identifiers (URI) Dynamic Delegation Discovery System (DDDS) Application (ENUM)".

Lewin, D. "Consistent hashing and random trees: Algorithms for caching in distributed networks", *MSc. thesis, MIT*, 1998.

Stoica I., Morris R., Karger D., Kaashoek M.F., Balakrishnan H. "Chord: A Scalable PeertoPeer Lookup Service for Internet Applications", *Proceedings of the 2001 ACM SIGCOMM Conference*, 2001.

Quanhao Lin, Ruonan Rao, Minglu Li. "DWSDM: A Web Services Discovery Mechanism Based on a Distributed Hash Table", *Fifth International Conference on Grid and Cooperative Computing Workshops*, Oct. 2006, 176-180.

Mwansa L., Janeček J. "P2P: Generic Network Location Service for Web Applications", *9th Annual Conference on World Wide Web Applications [CDROM]*, Cape Town Cape Peninsula University of Technology, 2007, 1-16. ISBN 9780620398374.

Ratnasamy S., Francis P., Handley M., Karp R., Shenker S. " A scalable content addressable network", *Proc. ACM SIGCOMM*, San Diego, CA, August 2001, 161-172.

Risson J., Qazi S., Moors T., Harwood A. "A Dependable Global Location Service using Rendezvous on Hierarchic Distributed Hash Tables", *International Conference on Networking*, April 2006, 4.

Singh K., Schulzrinne H. "Using an External DHT as a SIP Location Service", *Columbia University Technical Report CUCS00706*, New York, NY, Feb 2006.

AUTHOR BIOGRAPHIES

LABAN MWANSA was born in Zambia, received his Masters degree from Moscow State University of Statistics and Economics, He has worked both in Zambia and South Africa as a Senior Lecturer in Information Technology. He is currently a PhD candidate in Computer Science at the Czech Technical University in Prague. He is a member of the South Africa Computer Society and IEEE. His e-mail address is mwansl1@fel.cvut.cz.

DOC. ING. JAN JANEČEK, CSC. Associate Professor, Academic Board Member, Head of the distributed systems research group, Supervisor, Teacher Czech Technical University in Prague. His e-mail address is janecek@cs.felk.cvut.cz

SIMULATION AT A MARITIME CONTAINER TERMINAL: MODELS AND COMPUTATIONAL FRAMEWORKS

Pasquale Legato
Dipartimento di Elettronica, Informatica e Sistemistica
University of Calabria
Via P. Bucci, Cubo 41C, 87036, Rende (CS), Italy
E-mail: legato@deis.unical.it

Daniel Gullì and Roberto Trunfio
Centro di Supercalcolo per l'Ingegneria Computazionale
CESIC – NEC Italia S.r.l.
Via P. Bucci, Cubo 22B, 87036, Rende (CS), Italy
E-mail: {daniel.gulli, roberto.trunfio}@eu.nec.com

Riccardo Simino
High Performance Computing Division
NEC Italy S.r.l.
Via L. Da Vinci, 97, 20090, Trezzano sul Naviglio (MI), Italy
E-mail: riccardo.simino@it.neceur.com

KEYWORDS

Maritime container terminals, holistic approach, discrete-event simulation, parallel computing.

ABSTRACT

NEC-Italy high-performance computing division recognises that computer simulation aided organisation and management is the true challenge for a new generation of advanced decision tools for supporting operations in modern logistic platforms. Collecting academic expertises, research skills, computing machinery and dynamic realities with a significant growth into the field of container terminal logistics is a new R&D project at the CESIC-NEC center at the University of Calabria. The starting idea of this project is that discrete-event simulation is the best modelling approach to manage the complexity of logistic processes at container terminals. For these time-based systems, operating in a stochastic environment, becomes crucial to highlight both congestion and starvation phenomena embedded into logistic processes, in order to achieve reasonable targets of a good management of resources and, therefore, stay on the market. Here we present some queuing network based representations that are at the basis of an integrated simulation model under development. Since large and complex models are affected by a high burden on execution, we also remark the benefits of parallel and/or distributed computational frameworks. Numerical results on parallel analysis of simulation output data are given.

INTRODUCTION

The increasing international division of labour in the course of liberalisation, the resulting trade movements and the consequent need for faster, just in time freight transportation and cargo handling in real time lead transportation companies to study and develop more efficient and effective transport systems able to follow customers on their demand. Computing system companies may have a crucial role in this business, by providing sophisticated decision support systems on powerful, advanced computational resources and platforms.

The common, basic idea in freight transportation is that of resorting to the aggregation of multicommodity flows to be transported, by large carriers, for long distance routes across a certain number of intermediate logistic platforms (terminals), where freight consolidation may occur together with store and forward logistic functions and other source/sink based functions. Standardised containerization appears as the most notable and consolidated practice for freight transportation, especially through long distance maritime routes. Containerization is a system of intermodal freight transport using standardized large bins known as containers that can be loaded and sealed intact using multiple modes of transportation: ship, rail, road and air. Containers are an ISO standardized metal box of 8-ft wide by 8-ft high; the most common lengths are 20-ft and 40-ft. The container capacity is often expressed in twenty-foot equivalent unit (TEU). Containers are made out of steel and can be stacked on top of each other.

Nowadays, approximately 90 percent of the non-bulk cargo worldwide moves by containers stacked on transport. On ships they are typically stacked up to seven units high and there are ships that can carry over 9,000 TEUs. The world container fleet amounts to about 23.2 million TEUs and the container throughput reached 440 million TEUs in 2006 (UNCTAD 2007).

Maritime container terminals are the most important crossroads for transshipment and intermodal container transfers. The maritime container shipment follows the spokes-hub distribution paradigm: containers are shipped from a port (spoke) to another one through a small number of maritime transshipment terminals (hubs). Both spokes and hubs could be connected with inland container terminals by road and rail (the so called intermodal transfer). In opposition with the high number of container ports in the world, there are a few

of transshipment terminals (e.g., the Gioia Tauro terminal). Transshipment container terminals are large facilities for intermodal transport, able to handle millions of containers per year and berth large container vessels. These hubs are linked with spokes by means of small vessels called feeders, through minor (or short sea trade) routes. There are a few major liner trade routes that link the hubs (or deep sea trade routes). There are also oceanic container vessels, known as mother vessels, which sail on these intercontinental routes.

A maritime container terminal is a large and complex logistic platform organised around a set of logistic processes. The logistic activities at a container terminal often belong to more logistic processes, which require a whole vision of the system to be properly organised. This fact is critical for a good management of the system and the choice of the system modelling approach: an efficient and effective management of logistic activities in a container terminal can decrease the operating costs and service times and increase the quality of services in order to achieve a better market position. In this paper we report on the development of a discrete-event simulation platform for the whole management of a maritime container terminal, as it is ongoing at the CESIC-NEC research laboratory in southern Italy. The availability of a simulation environment where decision models could be developed and tuned according to a final customisation phase is considered as a novel, challenging market option. Simulation modelling and analysis should become an ordinary planning and control tool, e.g., to evaluate the performance of the container terminal when changes occur in the system configuration (what-if analysis) or to use simulation-based optimization techniques to optimize the whole system (Law and Kelton 2000).

As matter of fact, the whole model of a large and complex system, e.g. a container terminal, is affected by a high burden on execution. This statement is true especially for the most commercial simulators that are generally based on the *process interaction worldview* (Derrick et al. 1989). As recently suggested (Taylor et al. 2002), it is preferable to integrate heavy simulation modelling into a computational framework oriented to high-performance computing (HPC).

In the following, we classify and describe the logistic problem at hand. Afterward, we depict the queuing network model used to represent the core logistic processes at a container terminal. At the end, we discuss why HPC computational frameworks are a key added value into the study of real systems through discrete-event simulation

PROBLEM CLASSIFICATION

Vis and De Koster (2003) produced an interesting overview paper in the area of container logistics that gives a classification of the logistic processes in modern container terminals: *i)* arrival of the ship, *ii)*

discharging and loading of the ship, *iii)* transport of containers from ship to stack and vice versa, *iv)* stacking of containers, and *v)* inter-terminal transport and other modes of transportation. With respect to this classification, it is possible to identify a set of features that are common to many maritime terminals. The main difference between the greater part of container terminals located in Europe and North America and those in the Asia-Pacific region regards the logistic processes *iii)* and *iv)*. The latter relies on the "Indirect Transfer System" (ITS), in which process *iii)* and *iv)* are closely connected: a fleet of shuttle vehicles transports the containers from a vessel to beside the stack area (and vice versa) while dedicated cranes stack containers in compact regions. European and North-American container terminals are based upon the "Direct Transfer System" (DTS) in which process *iii)* and *iv)* are performed by the same actors: in this case a fleet of shuttle vehicles (called *straddle carriers*) moves the containers from a vessel to the storage area (and vice versa) for container stacking (retrieval) operations into (from) the slots assigned in the storage area (Cordeau et al. 2007). In our study we refer to a DTS maritime container terminal.

A SIMULATION PLATFORM FOR A MARINE CONTAINER TERMINAL

To fix ideas, we give a brief description of a real case that we are familiar with: the Gioia Tauro terminal.

This terminal, located in southern Italy, since it opened in 1995 has become in just a few years the largest transhipment port on the Mediterranean Sea. Its management has been early characterised by significant efforts towards an increasing computer aided organisation and control of the various logistic processes. Recently a manager friendly simulator has been designed for studying the congestion phenomenon at the port-input channel, the port admission policy for the newly arrived vessels in the roadstead and the allocation policies of berthing slots (Canonaco et al. 2007). Now, the problem of achieving an integrated management based on the usage of simulators and other operations research tools becomes more and more important.

The terminal is a large facility composed by: a harbour entrance of 250m wide and 18m water depth followed by a large roadstead for incoming vessels; 2 pilot boats; a 3,100m quay length with along a channel of multiple water depth ranging from 13.5 to 15.5 m (the quay is discontinuous, thus it is usually decomposed into two sub-quays); 18 quay rail-mounted gantry cranes (RMGs); a fleet of 75 straddle carriers (SCs), handling vehicles used to transfer the containers between the quay cranes and the yard; and a yard surface of 1.1 million square meters that can store nearly 59,000 TEUs. The yard has 32 sectors parallel to the quay and organized in two lines. An average quay has 32 lines, each containing 16 slots. A slot can host two one-TEU containers stacked one on top of the
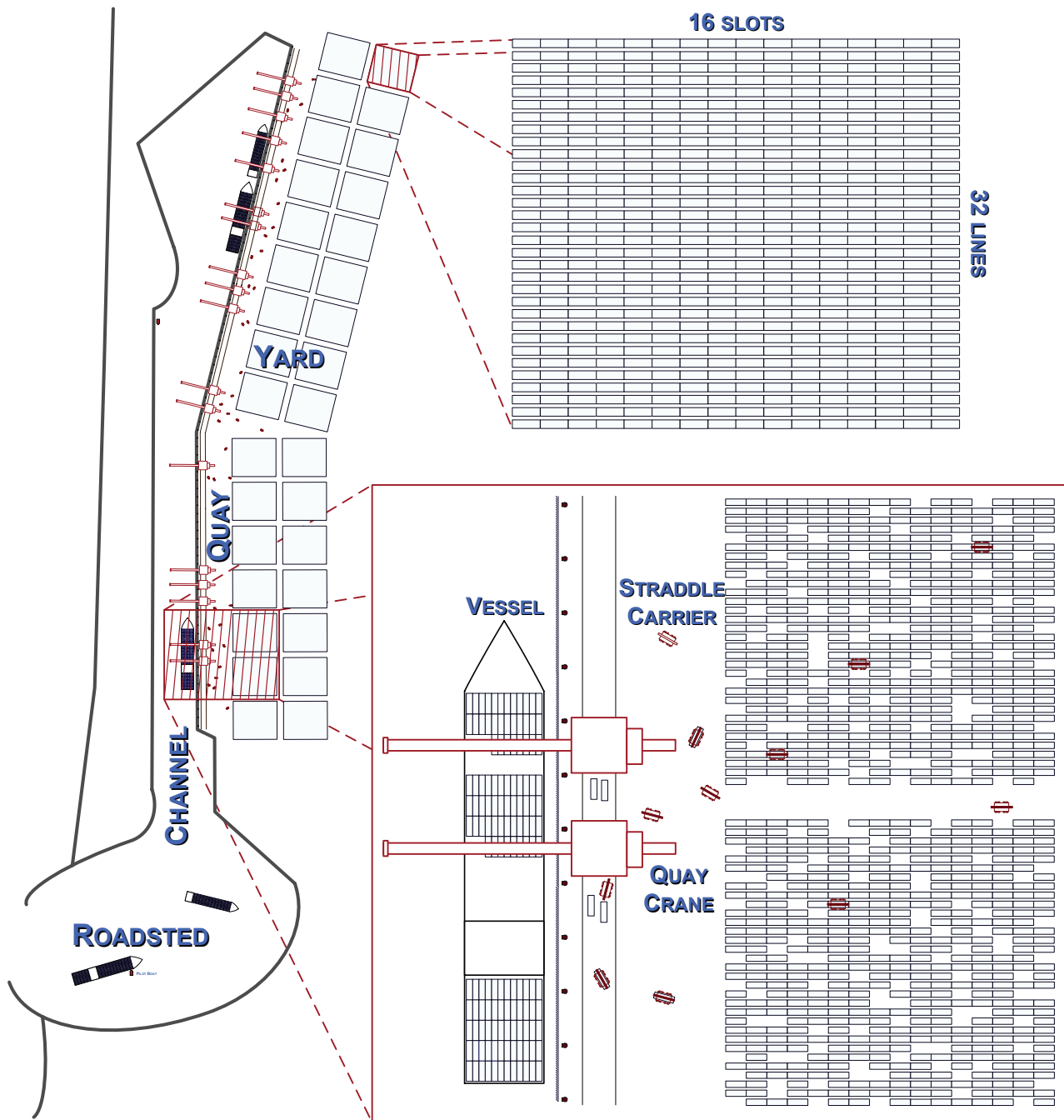
Figure 1: The Gioa Tauro Maritime Terminal Layout. Special Zoom on a Yard Sector and on the Loading/Unloading and Containers Transport Processes.

other. The layout of the container terminal is depicted in Figure 1.

Description of the Logistic Processes

The logistic processes are described in the following.

Planners of the terminal company construct a "berth schedule" on a weekly basis, which shows the berthing time and position for each incoming vessel according to the ETA (Expected Time of Arrival) and DTD (Due Time of Departure) of the vessel and the preferred sub-quay. This is the so called Berth Planning Problem (Park and Kim 2003; Cordeau et al. 2005). The aim is to find the optimal berth position for each vessel, i.e., the berth position that minimizes container handling cost from the vessel to location in the marshalling yard where outbound containers for the corresponding vessel are stacked. In Figure 2 there is an example of a berth schedule. We assume that the berth schedule is an exogenous input to the part of the simulation model discussed here. Actually, vessel entrance at port and berth schedule may be affected by the specific features and physical characteristics of the real port of interest, but, here we focus on common logistics connected with

other basic operations to be performed on berthed (standard) vessels by (standard) rail mounted gantry cranes. These operations are at the kernel of the simulator platform under development at CESIC-NEC Italy.

Figure 2: An Example of Berth Schedule

The arrival of the ship process proposed in (Legato and Mazza 2001) is the starting point for a new generalised model and is described in the following. Once the vessel has arrived outside the port (the arrival time is an uncertain time close to the vessel ETA), its mooring along the 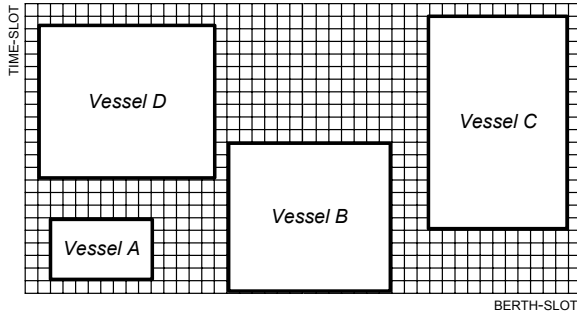assigned berth position depends on the following requirements: *i) formal conditions* (e.g. contractual agreements between the vessel's shipping line and the port of call for the use of port facilities), which means a priority policy for the port entrance queue; *ii) operational settings* (i.e. pilot mariner and pilot boat availability, berth space assignment and the least number of free quay cranes). If requirements are met, the ship is manoeuvred down the navigation channel and into its berth slots by one or two pilot boats; otherwise it must wait in the roadstead. In our model we have a finite vessel population of 97 ships of which 12 are mother vessels.

Once a vessel is at berth, container discharge/loading can be initiated only if mechanical (and human) resources are allocated; if not, the ship waits in its berth position until resource assignment. Discharge/loading operations are performed by RMGs placed along the berth: one or multiple cranes move containers between the ship and the quay area. The maximum number of quay cranes that is possible to assign to each vessel is restricted by *i)* the total number of cranes in the quay and *ii)* the maximum number of allowable cranes to each vessel due to physical (i.e. the length of the vessel) and logical constraints (i.e. interference between crane booms). Considering the span of the cranes (30 m) and the horizontal space necessary to stack and transfer away the incoming/outgoing containers of a vessel, the maximum number of cranes assignable to the longest vessel is 5 (this number is proportionally decreased for shortest vessel). The problem of assigning RMGs to the vessels for each time-slot is called "Quay Crane Deployment Problem" (QCDP) (Park and Kim 2003).

When multiple cranes are assigned to the same ship, crane interference has to be avoided and a complex scheduling problem arises to manage the relationships (precedence and mutual exclusion) existing among the holds of the same ship. This is the "Quay Crane Scheduling Problem" (QCSP) (Canonaco et al. 2008).

We manage both problems together in a two-phase approach: the first phase concerns the QCDP, while the second one involves the QCSP. Assuming that the berth schedule is deterministic for the model, in the first phase we use CPLEX to solve a mathematical model that is able to assign for every vessel at each one-hour time-slot a sub-set of RMGs (Legato et al. 2008a). Afterwards, in the second phase we use a metaheuristics to schedule the ship holds to the assigned RMGs (Legato et al. 2008b). Thus, the simulation model initialization is schematized in Figure 3.
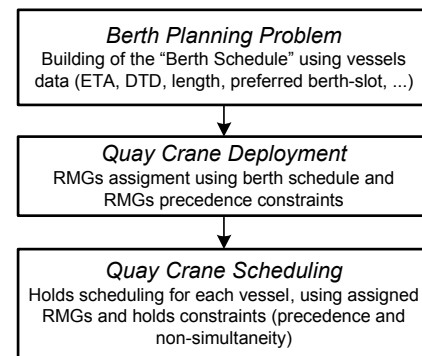
Figure 3: Simulation Model Initialization Process

The performance of the discharge/loading process highly depends on the availability of this type of cranes and their turnover speed. RMG performances are summarized in Table 1. Again, this problem has been deeply studied within the real context of Gioia Tauro Terminal (Canonaco et al. 2008), but without being integrated with the problem of simulating yard organisation, SC travel back and forth from the yard and analysing the effects of crane shifting along the berth, under the condition that multiple vessels are at berth.

Table 1: RMG Crane Average Performance

Performance	(m/min)
Container hoisting speed rated with load/light-load	50/120
Trolley travel speed	210
Gantry travel speed	45

In a DTS container terminal the transport of containers from ship to yard (and vice versa) and the container stacking processes are performed by a fleet of straddle carriers (SCs). SCs take in charge containers and cycle between the berth area and the assigned storage positions within the yard. Straddle carriers are capable with a laden container of relatively low speeds (up to 20–26 km/h). The yard is a passive resource: it consists of a matrix of 2 lines and 16

columns, in which each matrix element is a three-dimensional matrix that stands for a yard sector. SCs are able to select a slot within the yard structure to load/unload or stack containers. They are also able to compute the distance from the assigned RMG to the yard slot and back.

Each RMG has four SCs assigned. We have not developed any optimization code to dynamically assign SCs to RMGs (and we only use 72 of the 75 available SCs).

In the present case, no considerable inter-modal TEU transportation will be considered.

Modelling approach

The model has been developed by using a discrete-event simulation architecture based on finite state automata. The approach used to model the whole container terminal is based on previous promising works (Pidd and Castro 1998; Legato and Trunfio 2007): it represents the interaction between different logistic processes through special model objects called resource managers. Resource managers are gifted of a high-level view of the whole system and are able to operate on the system resources. They use exogenous data (i.e. berth schedule, RMG deployment and scheduling) to dynamically assign the container terminal resources to the jobs.

A Queuing Network Model Description

The logistic processes described above have been depicted using queuing networks. The arrival of the ship process is described in Figure 4. The model is based on the hierarchical representation proposed by Legato and Mazza (2001). Incoming vessels wait in a priority queue that represents the roadstead; outgoing vessels wait in another priority queue that stands for served vessels waiting in the assigned berth slot. A hidden model object called "Berth Manager" assigns pilot boats to incoming and outgoing vessels according to previously defined conditions (i.e., active and passive resource availability).
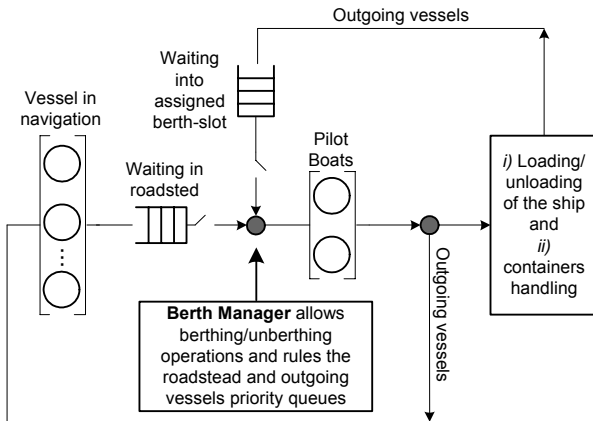
Figure 4: Arrival of the Ship Process

A vessel entering the rightmost (black) box in the figure means that it has been berthed and another queuing network model takes charge of it.
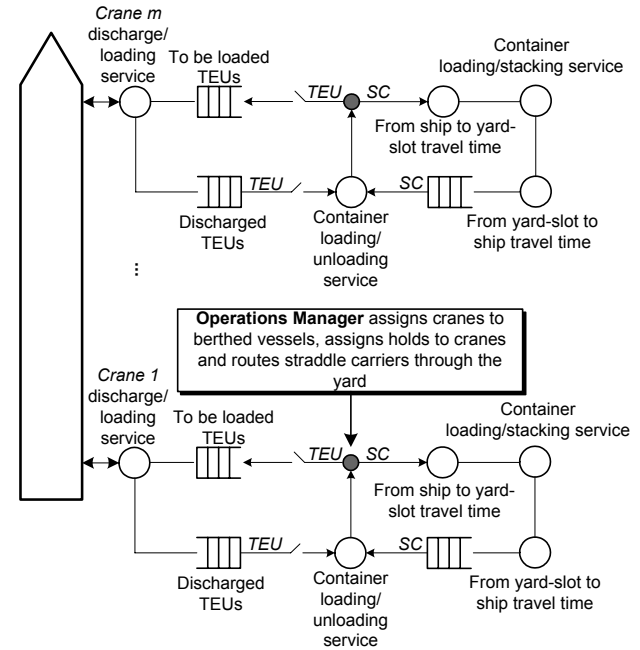
Figure 5: Discharging/Loading of the Ship Process

Figure 5 shows the queue network model for both the *i)* loading/unloading of the ship process and *ii)* container transport from the ship to the assigned yard-slot (and vice versa) and the consequent storage process. The simulation model uses a list of berthed vessels. Each vessel is assigned to a berth slot (a discrete part of the quay assigned in the berth schedule). For each vessel in list, a model object called "Operations Manager" assigns cranes to the vessels and starts the operations on the vessels. When discharging and/or loading operations have been performed, the Operations Manager pushes the served vessel out of the model and updates the system performance measures (e.g., overall completion times). In Figure 5, both the discharging and loading process of the ship are depicted. The Operations Manager tasks are: *i)* for each vessel in the berth, to assign the vessel holds to be served to each RMG ; and *ii)* to route SCs within the network in order to execute the optimal handling operations.

COMPUTATIONAL FRAMEWORKS

As part of the industry, our belief is that large and complex simulation models of logistic systems are not only an interesting academic research field of interest, but that they can be applied to real systems to provide effective scenario analysis and decision support to operational management in a reasonable time. Here is the key point: detailed models of large and complex systems are affected by a high burden on execution,

but, on the other hand, we do not believe as a good idea, when modelling logistic systems, that of using simplified system description and interdependencies among subsystems to get a lower computational burden. Modern simulation methods and technologies, such as simulation-based optimization are applied more and more in the organisation and management of logistic systems. Frequently these methods require evaluating thousands of alternative system configurations with the goal of searching for better system policies and configurations. Hence, only computational frameworks based on high-performance computing will be able to be candidate as effective tools for the tactical and operative managers.

Integration of Queuing-Network Models in a Parallel Computational Framework

In this context, we aim to integrate the model previously described into a HPC computational framework devoted to the statistical analysis of the output data from simulation runs.

Actually, we have developed a simple test case based on a single server, queue-model with the aim of experimenting some different methods for confidence interval estimation upon average performance measures that can be produced from simulation output data. Using a PC, wall-clock time experienced by us to get a satisfactory simulation output analysis for a single configuration of the queuing-network model that we proposed is of the order of dozens of minutes.

The methods to be compared have been selected between the following approaches (Katzaros and Lazos 2001): the well known *independent replications* (IR) and *batch means* (BM) and a mixed recent approach known as *replicated batch means* (RBM) (Andradóttir and Argon 2001).

The IR approach is a pure parallel methodology: the term independent replications refer to simulation runs of the same fixed length; each run starts from the same initial state, but uses a different seed in the pseudo random number generator. Despite the fact that each run starts from the same system state, there is no correlation between each replication: in fact, considering that for each run there is the presence of an initial transient that must be removed, using different seeds we guarantee that the remaining part of the run is really independent. Moreover, samples from the different replications are identically distributed. This approach is perfect whenever a parallel machine, with a set of m CPUs large enough ($m=20,...,30$) to submit one run per CPU, is available. In this case, to obtain m independent replications, we need the same number of CPUs and we commit m-times the effort to execute the initial transient.

The BM approach is a merely sequential method that cannot be used for variance estimation purposes in transient simulation. It is a really effective variance-estimation methodology for steady state simulation, in fact it uses only one long run, purged by the initial transient and then divided into equal batches. All the batches registered on the same run are used for variance estimation purposes. Using a sequential machine, the BM is better than the IR methodology, because it allows collecting m independent replications with the effort of estimating the initial transient only once. The drawback of this method lies in the correlation between samples used for variance estimation, Nevertheless, when simulating complex systems (e.g., the queuing network that we proposed) too much computational effort is needed using the BM method, while the IR methodology with the use of parallel machine seems to better suit our needs and overcome the BM.

The RBM variance-estimation methodology combines good characteristics of IR and BM: as a matter of fact, the number of initial transients that are estimated is reduced by simulating a system on m CPUs using the BM methodology. As for the IR, each CPU uses a different seed for the pseudo random number generation and simulates the system from the same initial state. Recently, it has been empirically proved that the RBM methodology is better than BM in terms of overall coverage of confidence intervals (Andradóttir and Argon 2001; Alexopoulos et al. 2006).

We believe that the RBM is the perfect trade-off between the IR and BM methodology and that it will be used as a basis for simulation output analysis in high-performance computing frameworks.

Here, we test the best methodology for simulation output analysis that we used for our complex queuing-network model, via a basic and repeatable test case consisting of a single queue model that may be considered as extracted from one of the queuing network models presented in the previous section. By fixing the arrival rate, $\lambda = 9$, from a Poisson distribution and the service rate, $\mu = 10$, from an exponential distribution we may use the well known analytical (exact) result on the mean waiting time $(1/(\mu - \lambda))$ to evaluate the coverage and the quality of the confidence intervals produced by different computational choices. In Tables 2, 3 and 4, we present numerical results on mean, variance and interval estimate of the waiting time, as obtained after having deleted 5.000 observations to avoid the bias effect of the initial transient behaviour. In particular, results in Table 4 are referred to a shorter batch length – i.e. 1.000 observations instead of 5.000 – because we were exploiting the effect of a quick stop of the run length by each processor. As one may see, numerical results are still quite goods whenever one decides to restrict the level of parallelism from 30 parallel processors, each performing 1 batch, to 5 parallel processors, each performing 6 batches. But we may not recommend a sort of saving on the computational power of each processor, by reducing the batch length: related interval estimates become unsatisfactory.

Table 2: IR with one batch for 30 parallel runs

Test	Mean / Interval	Variance
1	0.9348 [0.8315, 1.0382]	0.1110
2	0.9449 [0.8394, 1.0503]	0.1155
3	0.9387 [0.8400, 1.0374]	0.1012
4	0.8743 [0.8113, 0.9374]	0.0413
5	0.8592 [0.7984, 0.9200]	0.0384
6	0.9747 [0.8985, 1.0509]	0.0603
7	0.9490 [0.8597, 1.0384]	0.0829
8	0.8640 [0.8056, 0.9224]	0.0355
9	0.9299 [0.8247, 1.0351]	0.1150
10	0.8930 [0.7987, 0.9874]	0.0924

Table 3: RBM with 6 Batches for 5 parallel Runs

Test	Mean / Interval	Variance
1	0.8488 [0.7864, 0.9112]	0.0404
2	0.9624 [0.8792, 1.0457]	0.0720
3	0.8503 [0.7891, 0.9114]	0.0389
4	1.0339 [0.9189, 1.1490]	0.1376
5	0.8556 [0.7772, 0.9339]	0.0638
6	0.9160 0.8289, 1.0030]	0.0788
7	0.9085 [0.8187, 0.9983]	0.0838
8	0.8592 [0.7924 0.9260]	0.0463
9	0.8869 [0.8136, 0.9602]	0.0558
10	0.9263 [0.8700, 0.9826]	0.0329

Table 4: RBM with 6 Batches for 5 Runs, under Smaller Batch Length

Test	Mean / Interval	Variance
1	0.8965	0.1641

	[0.7708, 1.0221]	
2	1.1334 [0.8089, 1.4579]	1.0940
3	1.0181 [0.8697, 1.1665]	0.2289
4	0.9365 [0.7924, 1.0807]	0.2160
5	0.8664 [0.7310, 1.0019]	0.1906
6	0.7576 [0.6637, 0.8516]	0.0918
7	0.9304 [0.7750, 1.0857]	0.2508
8	0.7781 [0.6821, 0.8741]	0.0957
9	0.9381 [0.7850, 1.0912]	0.2436
10	1.0041 [0.8021, 1.2061]	0.4239

Another feature of the computational framework we are using is the master/slave approach proposed in (Legato et al. 2006). To ensure fault tolerant computing, we resort to the common practice of checkpoints, a policy to manage two different types of crashes: a crash of the *master node*, and a crash of a *slave node*. The idea is described in the following. In our master/slave approach, the master keeps track of the pool of independent replications dispatched to the slaves. Thus, whenever a slave crashes, the master can reassign the same configuration to another slave. On the contrary, whenever the master node crashes there is no way to recover the instantaneous state of the computational procedure. To react at the occurrence of a master crash, we have implemented a periodic checkpoint. In particular, at a sequence of suitable time points, we take a copy of *i)* the model object state (e.g., resource state, queued jobs), *ii)* the current seed value and *iii)* the list of scheduled events; therefore, the list of the ''latest'' parallel replications that have been dispatched to the slaves is saved in a file system. In such a way the simulation can be restarted from the latest point of each replication. This consolidated technique is based on using the so-called inter-communicators. The framework uses machine built-in error handlers as well as other ones defined by us: both allow us to detect a slave failure due to a non-success return code from a communication operation (i.e. a send or receiver) on one of the inter-communicators.

The computational framework is based on a *NEC TX-7 CC-NUMA* parallel machine with 32 *Intel 64-bit Itanium-II* CPUs with 64Gb of RAM: the machine is located at the CESIC research centre (CESIC 2006). To overcome the problem of the pseudo random number generator to be used, we use the *Mersenne Twister* algorithm (Matsumoto and Nishimura 1998) as suggested by Katzaros and Lazos (op cit).

Through this framework, we will conduct practical investigation using "what if" scenario analysis or using simulation-based optimization to evaluate the impact of the possibility of purchasing new resources (RMGs and SCs) in the container terminal. Hence, considering that in a real organisational framework, this kind of application is expected to run on PCs, one may recognize that an HPC-based computational framework appears to be the unique cost effective tool which can provide proper solutions to decision problems in a reasonable amount of time for the terminal manager.

CONCLUSION

Current practice with the usage of simulation for the optimal management of logistic processes reveals that decisions are pursued by decomposing and separately modelling the whole process in a set of sub-process. In this paper, we report on the CESIC-NEC project for developing an overall simulation model based on previous experiences at the largest container terminal on the Mediterranean Sea. The simulation model has been depicted using queuing-networks presented here and model development has been done using finite state automata. The model uses mathematical models or heuristics approaches to evaluate such feasible solution for the berth planning, quay crane deployment and quay crane scheduling problems. Currently, we are involved in a simulation-based optimization model to optimize the core terminal operations. Finally, we are attempting to address what the implications of high-performance computing in simulation-based computational frameworks are for this industry. In particular, we stated that the RBM variance-estimation methodology for simulation output analysis is the best, using modern low cost multi-core machines for simulation output analysis.

REFERENCES

Alexopoulos, C.; S. Andradóttir; N.T. Argon; and D. Goldsman. 2006. "Replicated Batch Means Variance Estimators in the Presence of an Initial Transient". *ACM Transactions on Modeling and Computer Simulation* 16, No. 4, 317-328.

Andradóttir, S.; N.T. Argon. 2001. "Variance Estimation Using Replicated Batch Means". In *Proceedings of the 2001 Winter Simulation Conference*. (Arlington, VA, USA). Vol 1, 338-343.

Canonaco, P.; P. Legato; R.M. Mazza; and R. Musmanno. 2008. "A Queuing Network Model for Management of Berth Crane Operations". *Computers & Operations Research*. Vol 35, 2432-2446.

Canonaco, P.; P. Legato; R.M. Mazza. 2007. **"**An Integrated Simulation Model For Channel Contention And Berth Management At A Maritime Container Terminal". In *Proceedings of the 21th European Conference on Modelling and Simulation*. (Prague, Czech Republic). 353-362.

CESIC. 2006. Centro di Supercalcolo per l'Ingegneria Computazionale. http://www.cesic.unical.it.

Cordeau, J.-F.; M. Gaudioso; G. Laporte; and L. Moccia. 2007. "The Service Allocation Problem at the Gioia Tauro Maritime Terminal". *European Journal of Operational Research* 176, 1167-1184.

Cordeau, J.F.; G. Laporte; P. Legato; and L. Moccia. 2005. "Models and Tabu Search Heuristics for the Berth Allocation Problem". *Transportation Science* 39, No. 4, 526-538.

Derrick, E.J.; O. Balci; and R.E. Nance. 1989. "A Comparison of Selected Conceptual Frameworks for Simulation Modeling". In *Proceedings of the 1989 Winter Simulation Conference*. (Washington, DC, USA). 711-718.

Katsaros, P. and C. Lazos. 2001. "Steady-state Simulation of Queuing Processes in Parallel Time Streams: Problems and Potentialities". In *Proceedings of the 5th Hellenic European Research Conference on Computer Mathematics and its Applications Conference* (Athens, Greece). 370-376.

Laganá, D.; P. Legato; O. Pisacane; and F. Vocaturo. 2006. "Solving Simulation Optimization Problems on Grid Computing Systems". *Parallel Computing* 32, 9, 688-700.

Law, A.M. and W.D. Kelton. 2000. *Simulation Modelling and Analysis*, 3d ed. New York: McGraw-Hill.

Legato, P. and R.M. Mazza. 2001. "Berth Planning and Resources Optimisation at a Container Terminal via Discrete Event Simulation". *European Journal of Operational Research* 133, 537-547.

Legato, P. and R. Trunfio. 2007. "A Simulation Modelling Paradigm for the Optimal Management of Logistics in Container Terminals". In *Proceedings of the 21th European Conference on Modelling and Simulation*. (Prague, Czech Republic). 479-488.

Legato, P.; D. Gullì; and R. Trunfio. 2008a. "The Quay Crane Deployment Problem at a Maritime Container Terminal". Submitted to the *22th European Conference on Modelling and Simulation*. (Nicosia, Cyprus).

Legato, P.; R.M. Mazza and R. Trunfio. 2008b. "Simulation-based Optimization for the Quay Crane Scheduling Problem". Submitted to the *2008 Winter Simulation Conference*. (Miami, FL, USA).

Matsumoto, M. and T. Nishimura. 1998. "Mersenne Twister: A 623-dimensionally equidistributed uniform pseudorandom number generator". *ACM Transactions on Modeling and Computer Simulation* 8, 3-30.

Park, Y.-M. and K.H. Kim. 2003. "A Quay Crane Scheduling Method Considering Interference of Yard Cranes in Container Terminals". *OR Spectrum* 25, 1-23.

Pidd, M. and R.B. Castro. 1998. "Hierarchical Modular Modelling in Discrete Simulation". In *Proceedings of the 1998 Winter Simulation Conference*. (Washington, DC, USA). 383-389.

Taylor, S.J.E.; A. Bruzzone; R. Fujimoto; B.-P. Gan; S. Straßburger; and R.J. Paul. 2002. "Distributed Simulation and Industry: Potentials and Pitfalls". In *Proceedings of the 2002 Winter Simulation Conference*. (San Diego, CA, USA). 688- 694.

Vis, I.F.A. and R. De Koster. 2003. "Transshipment of Containers at a Container Terminal: an Overview". *European Journal of Operational Research* 147, No. 1, 1-16.

UNCTAD. 2007. United Nations Conference on Trade and Development. Review of Maritime Transport 2007. (United Nations, New York and Geneva).

AUTHOR BIOGRAPHIES

PASQUALE LEGATO is an Assistant Professor of Operations Research at the Faculty of Engineering (University of Calabria), where he teaches courses on simulation for system performance evaluation. He has published on queuing network models for job shop and logistic systems, as well as on integer programming models. He has been involved in several national and international applied research projects and is serving as reviewer for some international journals. His current research activities are on the development and analysis of queuing network models for logistic systems, discrete-event simulation and the integration of simulation output analysis techniques with combinatorial optimization algorithms for real life applications in Transportation and Logistics.
His home-page is http://www.deis.unical.it/legato.

DANIEL GULLÌ received a Laurea degree (*cum laude*) in Automotive Engineering from the University of Calabria, Rende, Italy, in 2005. He is devoted to research in numerical simulation at NEC Italy Center for High-Performance Computing and Computational Engineering (CESIC). His current research interests include discrete-event simulation models for logistic systems and parallel simulation.

ROBERTO TRUNFIO received a Laurea degree (*cum laude*) in Management Engineering from the University of Calabria, Rende, Italy, in 2005, and is currently pursuing the Ph.D. degree in Operations Research from the same university. Moreover, he is a logistics engineer at NEC Italy Center for High-Performance Computing and Computational Engineering (CESIC). His current research interests include modelling languages for discrete-event simulation, simulation-based optimisation of logistics systems, and parallel simulation.

RICCARDO SIMINO Graduated in Electronics Engineering at the Politecnico of Milano in 1984 presenting a research on 3D vision systems for industrial robots based on Artificial Intelligence Technologies. He achieved a master in Marketing at the University of Bocconi, Milan, Italy. After having worked for several years as employee for Digital Equipment Corporation and as consultant for International Government Agencies, from April 2001 he is director of the High Performance Computing Division of NEC Italy and from 2003 General Manager of CESIC.

A DERIVATIVE CONTROL MECHANISM FOR SUPPLY CHAIN PERFORMANCE IMPROVEMENT

Olatunde Temitope Baruwa
Miquel Àngel Piera
Department of Telecommunications and System Engineering (LogiSim)
Universitat Autònoma de Barcelona, Barcelona, Spain.
Email: OlatundeTemitope.Baruwa@campus.uab.cat, MiquelAngel.Piera@uab.es

KEYWORDS

Derivative Control, Simulation, Coloured Petri Net, Supply Chain.

ABSTRACT

Organizations increasingly find that they must rely on effective supply chains to successfully compete in the global market and networked economy. Over the years, simulation models have proved to be a useful decision support tool for examining the performance of different system configurations and/or alternative operating procedures for complex systems to improve operational and organizational efficiency. This paper presents a new approach that transforms an optimization problem to a search problem by evaluating the interrelationship of decision making when either one or more actors in the supply chain take aggressive or conservative positions. This is aimed at improving and optimizing performance, both individually and for the entire chain in order to reduce costs, lower inventories, and increase manufacturing throughput.

INTRODUCTION

The complexity of supply chain systems requires the design of new decision support systems (DSS) that can evaluate and analyze the effects of an action applied to any of the sub-systems. Such a DSS requires a new modelling approach with a deeper understanding of the nature of the complexity itself. The effect of an action is propagated through the chain in different ways according to the interrelationship of independent components. Thus, a DSS must be able to accommodate perturbations that might occur and propagate at any point within the system.

Modern-day technology has become a vital tool in making important business decisions with a growing number of commercial packages, Advanced Planning and Scheduling applications and Enterprise Resource Planning (ERP) solutions employed to optimize supply chains. From the research community, some new developments such as connectionist approaches developed by the artificial intelligence community have provided different methodologies for optimizing practice based knowledge, just by codifying the expert experience in form of case histories [3]. Nevertheless, most of these approaches are unable to give

explanations about how optimal solutions are reached. Over the years, simulation models have proved to be a useful decision support tool for examining the performance of different system configurations and/or alternative operating procedures for complex systems to improve operational and organizational efficiency. The model must represent in the most simplified way, the dynamic interests of the real system which should allow understanding, describing, modifying and possibly controlling the behaviour of the system.

The model developed in this paper is a classical supply chain business model based on the beer game ideology developed at the Sloan School of Management in early 1960's [17] where the goal of the participants is to minimize costs either individually or globally. It is modelled as a discrete-event system with four stages having one participant at each stage. Customer satisfaction through the most efficient use of resources while minimizing costs serves as the primary objective of the supply chain. The business model considers order processing, fulfilment and delivery of orders, decisions on the quantity to produce or order and the effect of these decisions on the supply chain in order to maximize customer service level.

Several simulation modelling approaches to tackle supply chain decision problems, optimizing and improving the supply chain performance have been proposed. David Simchi-Levi et al(Beer Game Authors) developed a Web/Java based model that seek to clarify the advantages of taking an integrated approach to managing the supply chain; the value of sharing information across the supply chain components by demonstrating the bullwhip effect. [3] developed a timed-hierarchical Coloured Petri Net for the simulation and performance of the beer game supply chain model to study the bullwhip effect. [10] extended the simulation model developed by [3] using different ordering policies to demonstrate the bullwhip effect and analyse the impact of standard demand forecasting methods on the performance of the system.

A four parameter discrete model developed by [14] and the use of genetic algorithm techniques was proposed in [15] to simulate the effects of two ordering policies under a particular customer demand to optimize the performance of the chain. This approach restricts some parameters that might be of great interest to

improve the performance of the supply chain. In [1], [6] and [7], a Just-in-Time Petri Net model, Artificial agent based model and Arena software based supply chain model respectively were proposed to complement and understand the complexity of managing decision making in the beer distribution supply chain game.

In general, the developed models demonstrate and validate the bullwhip effect observed in supply chains and they are able to provide good solutions under standard conditions. However, none of these approaches can explain or justify why the optimal solution is obtained which would be very useful to deal with optimal or quasi-optimal solutions when the system is operating under different context conditions. When the dynamics of the problem changes, it is almost impossible to obtain a near optimal solution since the agents that drive the solution are unknown. Moreso, these models are not useful to understand the cause-effect relationship when either one or more of the actors in the supply chain takes an aggressive or conservative position. This is critical to the performance of any supply chain decision support system.

In order to solve this problem, the model should help to understand the consequences of a decision at a certain level. To deal with this, the model should give a proper solution to the following questions:
- Which are the consequences of firing, delaying or disabling a certain activity? And how are such actions propagated in the supply chain?
- Does the model support new dynamics specification?
- Does the model permit automatic evaluation of different policies using the same model?
- Does the model support the combination of simulation for global optimization and decision making in complex operations?
- Does the model provide feasible, quasi-optimal and optimal solutions in a timely fashion?

This paper presents a new approach that transforms an optimization problem to a search problem by evaluating the interrelationship of decision making when either one or more actors in the supply chain takes aggressive or conservative positions and fulfilling the above mentioned characteristics.

The paper is structured as follows: The next section describes the supply chain model and the major activities considered in the model. The third section discusses modelling formalisms while the fourth section gives the description of the formalism employed and the developed model. The fifth section focuses on the proposed approach and the results obtained are presented in the sixth section. Finally, the last section gives a summary of the paper and ideas for future research.

THE BUSINESS (SUPPLY CHAIN) MODEL

It is worth mentioning that there is a clear distinction between Business models and Production models. A business model is not a description of a complex social system itself with all its actors, relations and processes. Rather, it describes the logic of a 'business system' for creating value that lies behind the actual processes [11]. [9] defines business model as the managerial equivalent of the scientific method – you start with a hypothesis, which you then test in action and revise when necessary. Nevertheless, a business model should not be misconstrued as a strategy. A strategy is what makes you unique in the market place. However, performance of real production systems is very sensitive to various kinds of factors such as unexpected delays, queuing, breakdowns, operation time and inadequate labour on a day to day basis. Thus, a business model should also incorporate these low abstraction level aspects (i.e. production model)

Effective inventory management strategies and their efficient implementation are recognized as important elements in a business model. The targets are reduction of the level and value of inventory maintained, high speed in transforming them to orders delivered to customers and flexibility in dealing with changing customer demand[5]. We consider a decentralized four stage supply chain consisting of a Manufacturer, Distributor, Supplier and Retailer based on the Beer Distribution Game developed to demonstrate a number of key principles of supply chain management. We assume that there is a single participant at each stage. Figure 1 highlights the major activities carried out by each participant in the supply chain. In the decentralized supply chain, each participant possesses good local information (about his inventory, remaining orders, receiving amounts from his upstream supplier and amounts delivered to the downstream participant), but not in possession of global information.

A supply chain must be able to handle a significant amount of events, both expected and unexpected. The unexpected events, also called *exceptions or perturbations*, typically arise because there is usually a gap between supply chain planning and execution [8]. As described in figure 1, the main events to be considered are: order filling, fulfilment of orders, delivery and the decision on the quantity to order or produce. The decision made on the quantity to order or produce plays a significant role in evaluating the overall effective performance of the supply chain. When decisions on what to order comes to play, there exists two possibilities; the possibility of having excess stock with low demand or the possibility of excess demand with low stock. The presented approach tends to strike a balance between the two so that the expected demand is closer to the actual demand in order to avoid out of stock and unnecessary back orders.
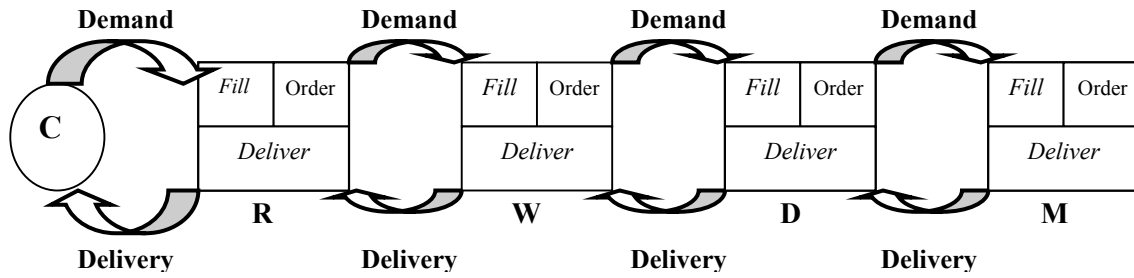
Figure 1: The Supply Chain Model
Key: C-Consumer; R-Retailer; W-Wholesaler; D-Distributor; M-Manufacturer

The model considers a "*Derivative*" control mechanism for managing these events and responding to them dynamically when perturbations occur. These events can be implemented if the modeller perceives that each action such as placing an order, delivering a product, or taking a decision regarding inventory levels is a discrete event with dynamic properties [10]. Since the number of reachable states in a supply chain is typically very large it is necessary to describe clearly all activities associated with events.

The triggering of an event can give rise to several consequences. For example, if an order is placed on a participant in a particular period and there is no sufficient inventory to fill the order, this results to a back log of order, a possible out of stock in the next period and a possible aggressive order placement on the upstream participant. Each time an order is placed on a participant, it is first checked whether the amount of products in inventory is sufficient to fill it. In case the amount is sufficient, the order placed will be delivered to the downstream component. Otherwise, this generates back order that will be fulfilled along with a new order placed in the next period. The supply chain model allows the delivery of incomplete orders. When an order is received, it is checked against the order placed. If an incomplete order is delivered, this might trigger an unwanted demand in the next period by the downstream participant depending on the pattern of orders it receives.

MODELLING FORMALISM

A great deal of formalisms for modelling discrete event systems exists in the literature. Amongst them are: Flow Charts, Block Diagrams, Automata, Temporal Logic, Calculus of Events, Artificial Intelligence, Discrete Event System Specification(DEVS) and Classical Petri Nets(PN). Business models require a formalism that is dynamic, descriptive, depicts causality, easily accessible interpretation and able to integrate every aspect of the system specification. Most of these formalisms do not support the specification of low level production aspects which are necessary to implement the proposed decision making approach. Though, classical PN provides an appealing graphical representation which facilitates easy understanding of the system behaviour and the synchronisation of

concurrent processes, it does not support information flow for decision making which is essential for any DSS.

The evolution of Coloured Petri Nets (CPN) has been driven by the desire to develop a modelling language – at the same time theoretically well-founded and versatile enough to be used in practice for systems of the size and complexity that can be found in typical industrial projects [4]. CPN supports specification, information flow and the manipulation of data values through a functional programming language. The main advantages of CPN models are given in [4]. The main CPN components that fulfil the cause-effect event relationship analysis include: places, transitions, input arc expressions, colour sets and state vector. See [12, 15] for the description of these terms and tutorial on CPN.

THE CPN MODEL

The proposed approach is implemented on CPN Tools software developed and maintained by the CPN Group, University of Aarhus, Denmark [16]. CPN allows the representation of a system in a composite structure with few places and transitions. To analyze and understand the cause-effect relationship of the system in an efficient way the model elements have been represented at the same level rather than the hierarchical structures developed by other authors to simply serve simulation purposes.

The supply chain is modelled as a timed net with the global clock of the CPN tool used as the weekly timer (time 1 represents week 1, time 2 represents week 2 etc). Each time the global clock increments, this implies that the cycle of events for a particular week has been completed. Within each week, all the possible events in the supply chain must be completed before the clock changes. Figure 2 shows the CPN supply chain model and tables 1, 2, 3 and 4 summarize the meaning of the place nodes, transitions (events), colours and functions. It should be noted that the same model can be used when events behave as a random variable.

The system is described by 9 transitions classified in a group of 4 major activities – order filling (T1, T2), delivery (T5, T6, T7, T8), decision on quantity to order
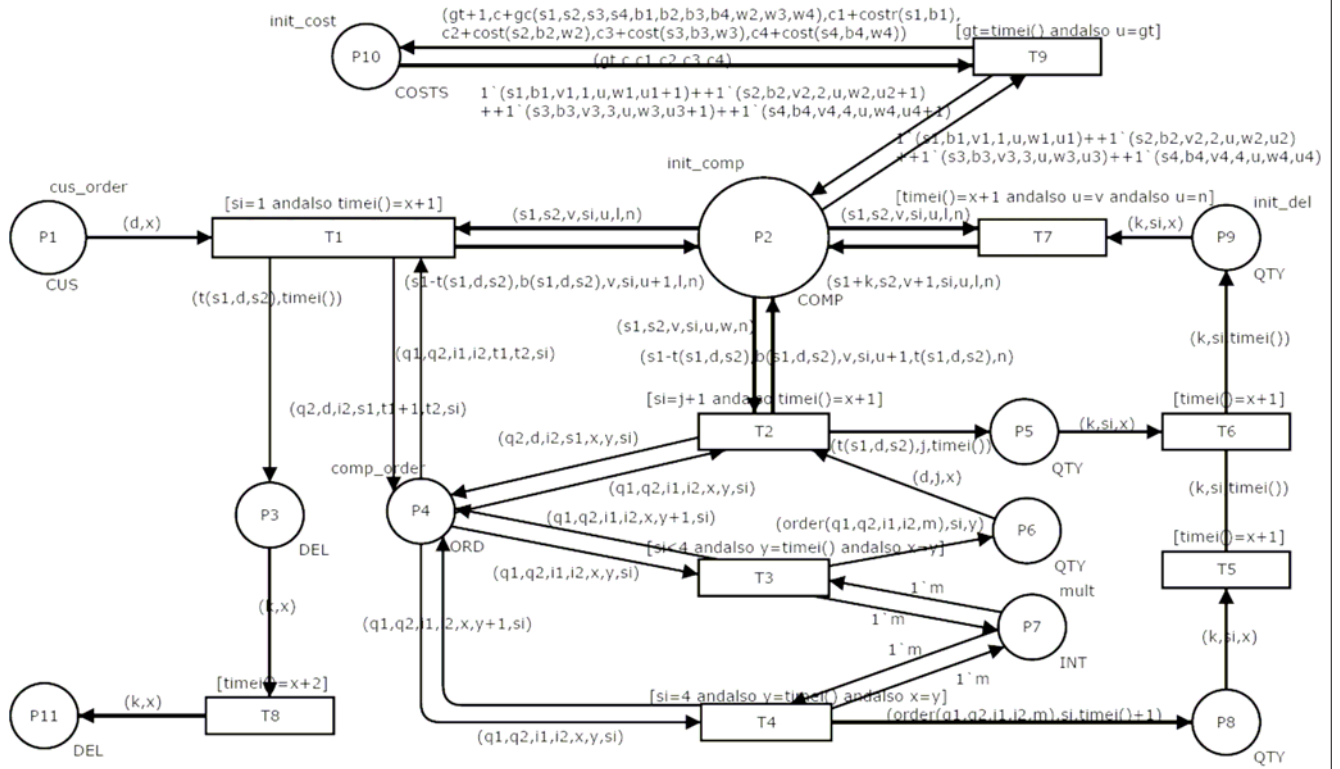
Figure 2: The CPN Model of the Supply Chain

Table 1: Event Description

Transition	Description
T1	Filling of customer orders by the retailer and updating P4 for the computation of new orders.
T2	Filling of weekly retailer, wholesaler and distributor orders by the wholesaler, distributor and manufacturer respectively and updating P4 for the computation of new orders.
T3	Computation of weekly orders by retailer, wholesaler and distributor.
T4	Computation of quantity to produce by manufacturer.
T5,T6	One week delay for quantity produced and filled orders.
T7	Delivery of orders after the two-week delay to be added to inventory.
T8	Delivery of customer orders after the two-week delay.
T9	Computation of weekly global cost and individual component cost.

Table 2: Place Nodes Information

Place	Colour	Description
P1	(d, x)	Weekly customer orders
P2	(s1, s2, v, si, u, l, n)	Component Information(inventory, back order, control for delivery, component identifier, order filling control, delivered orders, cost control)
P3	(k, x)	Quantity to deliver to customer
P4	(q1, q2 ,i1 ,i2 ,x, y)	History of previous and current quantity demanded and inventory level.
P5,P8,P9	(k, si, x)	Quantity to deliver
P6	(d, j, x)	Quantity to order by retailer, wholesaler and distributor.
P7	m	Multiplier
P10	(gt, c, c1, c2, c3, c4)	Cost (global, retailer, wholesaler, distributor, manufacturer)
P11	(k, x)	Customer delivered orders

Table 3: Function Description

Function/Value	Description
cus_order	Weekly customer orders
comp_order	Initial marking for order computation 1`(4,4,4,4,0,0,1)++1`(4,4,4,4,0,0,2)++1`(4,4,4,4,0,0,3)++1`(4,4,4,4,0,0,4)
init_del	Initial marking for delivery in the first three weeks 1`(4,1,(~1))++1`(4,2,(~1))++1`(4,3,(~1))++1`(4,4,(~1))++1`(4,1,0)++1`(4,2,0)++ 1`(4,3,0)++1`(4,4,0)++1`(4,1,1)++1`(4,2,1)++1`(4,3,1)++1`(4,4,1)
mult	Marking for multiplier (1`(1)++1`(2)++1`(3)++1`(4))
init_comp	Initial marking for component information 1`(0,0,0,1,0,0,0)++1`(0,0,0,2,0,0,0)++1`(0,0,0,3,0,0,0)++1`(0,0,0,4,0,0,0)
init_cost	Initial cost (1`(1,36,0,12,12,12))
t(s1,b,s2)	Computation of quantity to be delivered
b(s1,b,s2)	Back order computation
order(q1,q2,i1,i2,m)	Computation of quantity to order/produce by each component
gc(s1,s2,s3,s4,b1,b2,b3,b4)	Global cost computation
cost(s1-4,b1-4,w1-w4)	Component cost computation
timei()	Global clock

or produce (T3, T4) and cost computation (T9). The function "order ( )" in the model shows the "derivative" control mechanism implemented in the supply chain for decision making. This will be fully discussed in the next section. An initial steady state is assumed in the supply chain. During the first three weeks, the customer and the participants order constant demands per week, the supply chain members hold the same quantity in inventory as the customer's demand in the first week and the same quantity is delivered to the supply chain members in the subsequent two weeks. The participants can order as much as they want in the beginning of the fourth week. The transition guard expressions in the model ensure the synchronization, sequence and priority of events with the use of the colours embedded in each place. Place P2 represents the participants – retailer, wholesaler, distributor and manufacturer with each having an identifier. This can be modelled with more participants at each stage as CPN allows the inclusion of more parameters with the use of colours.

PROPOSED APPROACH – DERIVATIVE CONTROL MECHANISM

Simulation gives different solutions according to each scenario. With this, a manager can only understand how the system works; he has no information on how to improve the reality. The derivative control mechanism introduced in the model explores all possible existing solutions and a decision maker is able to understand the effect of each decision taken at any point in time on his own system and the entire chain as a whole. The approach also incorporates the flexibility in dealing with changing customer demands.

To enhance the formal analysis of the model, the control mechanism is supported with the application of the coverability tree (CT). It is generally believed that adding more parameters to the system in one single

model makes the size of the state space grows exponentially. However, the use of the coverability tree coupled with heuristics tackles the state space explosion [13] by running a check on every path of the tree to identify dead-end nodes, stop the possible expansion of old nodes (same markings) and the successive increments of markings on each node for all reachable states. Table 5 shows the logic table for the derivative control mechanism (function "order ( )" in the model).

The decision mechanism is driven by changes in demand and inventory levels whose derivative is taken over two (2) periods (n and n-1). This is weekly ordering or production decision in an effort to satisfy downstream demands. A multiplier "m" with values ranging from 1 to 4 is introduced in order to evaluate the conservative and aggressive positions of the actors involved in the chain. Let's consider decisions 1, 4, and 8 as an example:

Decision 1: Order or produce the current demand when there is no change in demand and inventory level.
Decision 4: Order or produce the multiplier of the current demand if there is an increase in changes in quantity demanded and no inventory level change.
Decision 8: A decision to order or produce the divisor of the current demand implies a decrease in quantity demanded and an incremental change in inventory level.

As a result, it allows checking a variety of state space solutions, opening the whole tree, understanding the cause-effect relationship of each action and how the actions are propagated through upstream and downstream. Since the model developed is not just for simulation but a decision support system that would allow the stakeholder to choose from a variety of solutions the model presents. In essence, the approach transforms the optimization problem to a search problem in which a decision maker can select a near optimal solution that best fits the current situation in a

Table 4: Derivative Control Mechanism

	∇D_n	∇I_n	Decision
1	0	0	D_n
2	0	1	$1/m * D_n$
3	0	-1	$m * D_n$
4	1	0	$m * D_{n-i}$
5	1	1	D_{n-i}
6	1	-1	$m * D_n$
7	-1	0	D_{n-i}
8	-1	1	$1/m * D_{n-i}$
9	-1	-1	D_n

Key: -1 – Decrease   0 – Constant   1 - Increase

$\nabla D_n = (D_n - D_{n-i}) / \nabla n$ $\nabla I_n = (I_n - I_{n-i}) / \nabla n$

D_n / I_n – Demand/Inventory at week n D_{n-i} / I_{n-i} – Demand/Inventory at week (n-1)

m – Multiplier (1, 2, 3, 4); i = 1 $\forall n = 1, 2...N$ (N is the total number of weeks)

timely fashion. The next section justifies the approach employed with the results obtained.

RESULTS

The model was driven by a step change in customer demand as shown in Figure 3 for a period of 15 weeks. Given the change in demand, the system can be ruled according to different costs. Figure 4 shows the global costs obtained for the entire chain during this period. The state space generated a good number of solutions at the end of this period in which 190 were collected for analysis. The system cost range from $524 to $1052. The inventory holding cost attracts a charge back of $1 weekly per case while a penalty of $2 for back orders.

Most of the policies will lead the system to high production cost and huge penalties. However, with state space analysis, it is possible to obtain a policy that supports customer demand at a very minimal cost ($524 – marked point on the graph). The policy that drives the system to the optimal cost during the first ten weeks is shown in Figure 5. The figure gives a clearer view of how customer demands are propagated from downstream to upstream. Participants take positions based on downstream demand and periodical inventory levels. Also, a steady state was maintained by the participants during the first six weeks. After the fourth week, the position of the retailer changed due to upward change in customer demand in the fourth week. The retailer's reaction to this change prompted the wholesaler to re-adjust his position on the chain. This affected the next two upstream participants in the subsequent weeks.

CONCLUSION

A derivative control mechanism for supply chain performance optimization has been proposed. The approach implemented supports the integration of simulation for global optimization and decision making. The efficiency of the approach was shown in the results section.
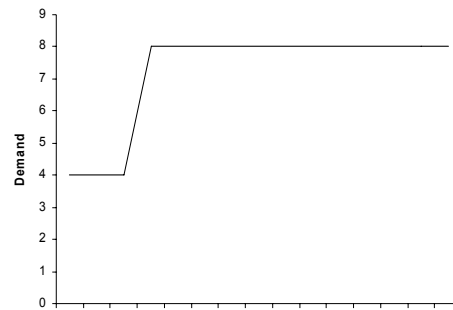
Figure 3: Customer Demand
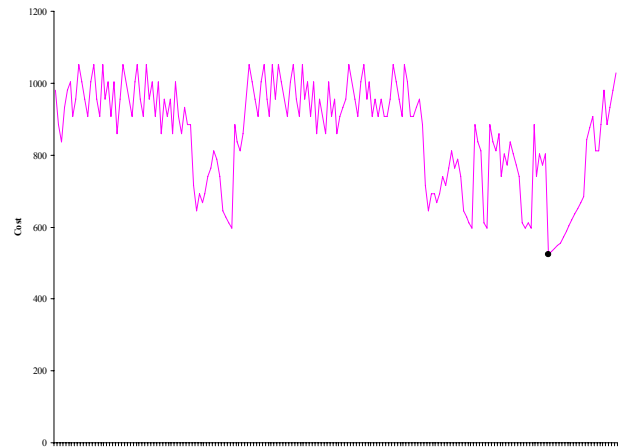
Figure 4: Global Cost

In order to reduce computational time and the size of the coverability tree, future research work should focus on the introduction of heuristics to prune the search space. Also, the effects of taken the derivative over a longer period can be considered with respect to product lead time.
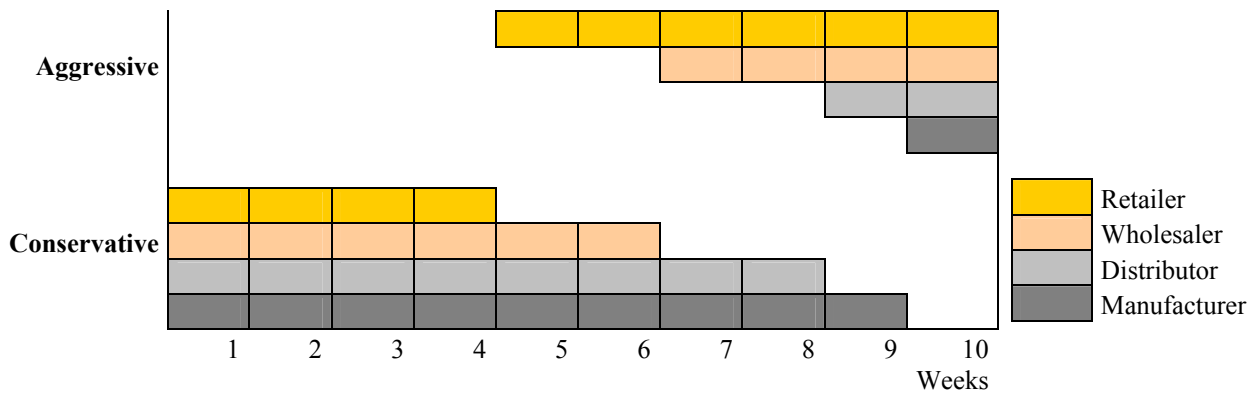
Figure 5: Positions of Participants for the Optimal Policy

REFERENCES

[1] Desrochers A. and Pia Fanti M. 2005, "A Supply Chain Model Using Complex-valued Petri Nets" *Proceedings of the IEEE International Conference on Robotics and Automation.*

[2] Dias W.P.S 2007, "Philosophical Grounding and Computational Formalization for Practice Based Engineering Knowledge" *Knowledge Based Systems 20.*

[3] Dragana M. et al 2004, "Bullwhip Effect And Supply Chain Modelling And Analysis Using CPN Tools*", Proc. of the 5th Workshop and Tutorial on Practical Use of Coloured Petri Nets and the CPN Tools. Aarhus, Denmark 219-234.*

[4] K. Jensen 1997, "Coloured Petri Nets. Basic Concepts, Analysis Methods and Practical Use*" Volume 1, Basic Concepts. Monographs in Theoretical Computer Science, Springer-Verlag, 2nd corrected printing 1997. ISBN: 3-540-60943-1.*

[5] Ka-Fu C. and Chi-Fai C. 2006, "A Manufacturing Supply Chain Business Model and Its Implementation in a Plastics Manufacturing Company" *Materials Science Forum Vols. 532-533 Pg 1096-1099.*

[6] Kimbrough S.O. et al 2001, "Computers Play the Game: Can Artificial Agents Manage Supply Chains? *34th Hawaii International Conference on System Sciences.*

[7] Kumar S. et al 2007, "Demonstrating Supply Chain Parameter Optimization Through Beer Game Simulation, Information" *Knowledge, Systems Management, Volume 6, Number 4 Pg 291 – 322.*

[8] Liu R. et al 2007, "A Formal Modelling Approach for Supply Chain Event Management" *Decision Support Systems, Volume 43, Issue 3, Pg 761 – 778.*

[9] Magretta J. 2001, "Why Business Models Matter" *Harvard Business Review Article, May PN R0205F*

[10] Papanagnou C. I. and Halikias G. D. 2006, "Analysing Different Ordering Policies in a Series Supply Chain by Using Coloured Petri Nets" *20th European Conference on Modelling and Simulation, ECMS.*

[11] Petrovic, O. et al 2001, "Developing Business Models for eBusiness" *Proceedings of the International Conference on Electronic Commerce,, Vienna, Austria, October 31 -- November 4.*

[12] Piera M.A et al 2004, "Optimization of Logistic and Manufacturing Systems through Simulation: A Colored Petri Net-Based Methodology". *Simulation Journal, V. 80 (1). 2004.*

[13] Sònia B. et al 2005, "Specification of Metaheuristics in Colored Petri Net Models"International Mediterranean Modelling Multiconference, EMSS.

[14] Sterman J.D 1989, "Modelling Managerial Behaviour: Misperceptions of Feedback in A Dynamic Decision Making Experiments" *Management Science 35 Pg. 321-379.*

[15] Strozzi F. et al 2006, "Beer Game Order Policy Optimization under Changing System Demand" *Science Direct (Elsevier BN) Pg. 2153-2163.*

[16] http://wiki.daimi.au.dk/cpntools/cpntools.wiki

[17] http://beergame.mit.edu/

AUTHOR BIOGRAPHIES

OLATUNDE T. BARUWA holds a bachelor's degree in Electronic and Computer Engineering (Lagos State University, Nigeria, 2004) and a master's degree in Management (Universitat Pompeu Fabra (UPF), Spain, 2007). He is currently a PhD student in the Department of Telecommunication and Systems Engineering at the Universitat Autònoma de Barcelona, Spain where he is developing his dissertation and a member of LogiSim, a Modelling and Simulation Institution sponsored and founded by the local government of Catalunya. Prior to studying at UPF, he worked at the Lagos Business School, Nigeria for 2 years. His research interests are in the areas of modelling, simulation and optimization of logistic systems, industrial processes and supply chain management. His e-mail address is olatundetemitope.baruwa@uab.cat.

MIQUEL ÀNGEL PIERA I EROLES (DR.) received his MSc (Control Engineering) from the University of Manchester Institute of Technology in 1990 and his PhD degree from the Autonomous University of Barcelona (Spain) in 1994. He participates in industrial research projects in the logistics and manufacturing field and at present he is Co-director of LogiSim, a Modelling and Simulation Institution sponsored and founded by the local government of Catalunya. Professor Piera has been the coordinator of the Spanish Simulation group of the International Federation of Automatic Control. Recently, he has published a modelling and simulation book that is being used for teaching in many Spanish Universities.

FRAMEWORK FOR INTEROPERABLE OPERATIONS IN PORT FACILITIES

Agostino G. Bruzzone
McLeod Institute of Simulation Science Genoa- DIPTEM
University of Genoa
Via Opera Pia 15, 16145 Genova, Italy
Email *agostino@itim.unige.it*

Simonluca Poggi
Liophant Simulation
Via Molinero 1, 17100 Savona, Italy
Email *simonluca.poggi@liophant.org*

Enrico Bocca
MAST srl
Torre Distripark, Nuovo Porto di Voltri
16158 Genova, Italy
Email *enrico.bocca@mastsrl.eu*

ABSTRACT

Among the most advanced techniques, interoperable simulation is actually emerging in those application fields where different entities/components dynamically interacts. Currently such a technology is becoming more and more effective in new areas, reproducing the experiences already collected in military sectors. For instance, logistics infrastructures and ports see many entities interacting dynamically, their synergy strongly affects the overall efficiency of the port terminals as well as the safety of the operators.

DIPTEM is involved in a new research, funded by COSMOLAB Consortium, for creating a distributed federation able to reproduce port operations with special emphasis on the dock cranes; in fact DIPTEM collected over the last five years multiple experiences in this sector and in bordering areas: SITRANET and INNOVARE researchs for Intermodal Terminal, RESET for River Logistic.

CYBERSAR FEDERATION FOR PORTAINER SIMULATION

The aim of the research is to simulate the behavior of intermodal equipment (cranes, stackers, trucks etc.) through the development of federates characterized by the capabilitiy to operate stand-alone or to be integrated in the HLA federation for cooperating. In such way, different trainees are able to work together in order to simulate real cases and complete missions. Consider a crane operator moving a container from a ship on a truck, driven by another operator, a third one is expected to move the container on a free area of the terminal using a reach-stacker simulator. Monitoring and supervising the results of the mission and operating both in LAN both in WAN the operators can act as previously described; in general sense not only driver simulators have to be integrated, the DIPTEM applied DIS in their past researches in order to create training exercise for managers/planners (i.e. dock manager, yard planner, control room, etc.). In the same way it is possible to integrate in the Federation also these entities directing the operators, providing support for testing policies and procedures.

The interaction among the terminal vehicles as prime cause of accidents and injuries in port operations has to be stressed; at the same time the handling policies and procedures are the key factors for improving productivity and safety. Based on such a consideration it is clear that past experiences in extended application in military sector combined with new technologies enabling low cost simulation, allow to export these

experiences in new areas, such as port training or industrial facilities.

The proposed case is related to the development of a federation focusing on the Portainers or dock gantry cranes: big equipment devoted to load/unload container ships.

The proposed integration among simulators is based on HLA (High Level Architecture) while the simulation engine is developed by the DIPTEM.

The proposed engine guarantees to have simulators fully scalable combining different federates in order to distributed/construct complex combinations (i.e. distributing heavy computational workload among federates); the most complex cranes are equipped of static mock-ups (i.e. port truck drivers) and full motion 6Degree of Freedom simulators. The design of the entire architecture needs special attention, actually the configuration is based on:

- one full scope simulator
- 6DOF motion platform
- advanced visualization solution integrated with eye tracking system (crane operator)
- interactive pilot station
- workstation
- large screen (truck drivers)

The system is installed in a shelter for easy mobility among different sites, this configuration includes:

- full motion station
- 3 basic stations
- instructor station (re-configurable for driving a vehicle)
- 1 panoramic observer station

Therefore the system is designed to be combined with other shelters and to be reconfigured based on the necessities.

Another innovative aspect is related to the integration of ship motions with movement at docks as well as detailed elastic models of cables; these parts represent special federates operating on High Performance Computers remotely (WAN) in case of critical events/conditions simulation, or to be substitute by meta-model running on the local workstations (LAN) for regular training.

Also, DIPTEM is currently experiencing the integration of traditional training with simulation: based on blended solutions combining different kind of training equipment it is possible to operate in cooperative/competitive mode.
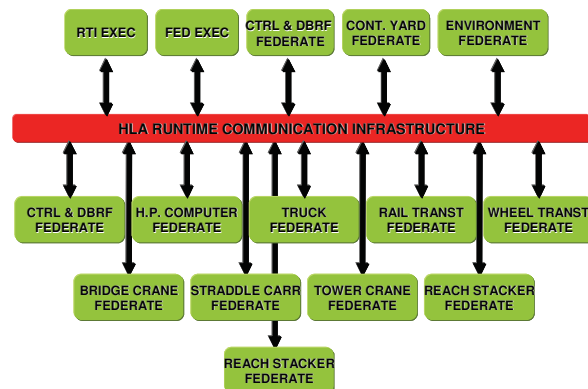

Figure 1: Mock-up motion platform

Figure 2: HLA Federation

VIRTUAL WORLD

The researchers have experienced new opportunities for applying Real Time Distributed Simulation, HLA-based, with special attention to Vehicle Driving and crane operation, taking into consideration the development of an ad hoc virtual world reproducing the entire Cagliari Port; the research obviously take into account several issues such as operative procedure definition, operators training & education, improvements on handling safety, improvements on operative efficiency.

Based on these inputs, the researchers developed the entire world including the surrounding areas, roads,

container yard, different type yard cranes, ship cranes, ships etc.

These world is integrated in the new generation of simulators oriented to the dissemination of simulation techniques as training tool in the logistics users community.
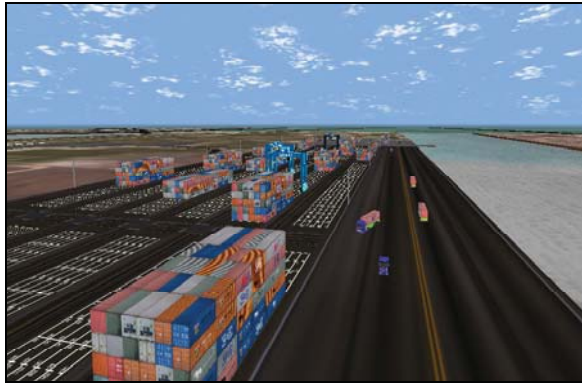

Figure 3: Virtual Yard with Virtual Cranes and Virtual Trucks

Figure 4: Portainers: Virtual Cranes operating on Ships

Figure 5: Virtual Ships with Four Portainers

Figure 6: Virtual Transtainer operating on the Yard

The researchers guaranteed that the virtual world was real-time, and emphasized the importance of creating a hierarchical structure of the objects in order to be integrated in the simulators; these are integrated over a LAN/WAN (Local Area Network/Wide Area Network) by using HLA (IEEE Standard and regulation for US DoD Simulators), this on the basis of HLA characteristics which make possible to realize an interactive real time simulation across a network.

Figure 7: The Whole Virtual World with Channels, Terminal and Roads

Based on such a choice a wide range of configurations and operative applications is possible, from operating stand alone on a single PC to creating a federation of cranes interacting in a networks.

The solution proposed is designed to allow dynamic reconfiguration of each PC in order to reallocate the equipment and to create different scenarios; obviously this is in some way limited by the availability of proper hardware device on a PC Workstation for driving/operating a specific vehicle.

Therefore by using simple game interface it is possible to configure a large set of position that can quickly changed from cranes to truck for creating complex cooperative scenarios; in this case it is also possible to define competitive operation where different team/squad are working concurrently in order to get better production.

The developed HLA configuration was designed in order to include several federates:

- Portainers federation allows the operator to practice a portainer in different scenarios. The operator can virtually load and unload container from a ship, in a virtual dock where different portainers work simultaneously. The portainer not in use by the operator is automatically under control of the simulator itself

- Control & Debriefing federation : the trainer can set such a federate every new training session. Trainer can change environmental conditions choosing different number of containers, of operative straddle carriers, number and type of ships in port and number of trains. This federation automatically controls all the trucks and carriers not directly used by operators and provides automatic traffic on the road around the intermodal node.

- Debriefing federate evaluates operator performance.

- High Performance Computer (HPC) Interface

- Yard Traffic Simulation controls traffic in the intermodal area

- Environmental Simulation: based on such a federate it is possible setting simulation starting time, weather condition, wind and sea condition included

- Truck, reach stacker, straddle carrier, wheel transtainer, rail transtainer, bridge crane, heavy crane federations improve the quality of the training session allowing the operator to drive different cranes in multipurpose operation.

In this case the users defined as critical the necessity to integrate also High Performance Computing facilities in the federation in order to guarantee real-time simulation of very complex phenomena (i.e. ship motion at dock and cable stress/strain).

Due to this necessity a specific federate was developed able to operate through simplified meta-models when HPC facilities are not available and to interact with the Supercomputing grid when possible.
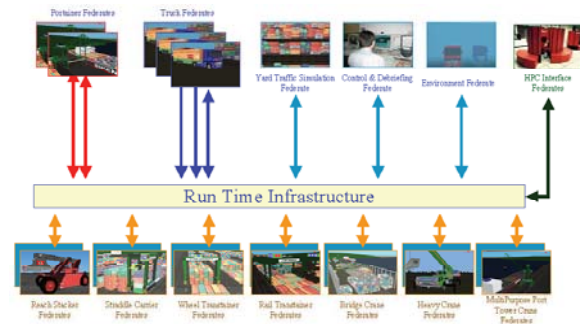

Figure 8: Run Time Infrastructure

OPERATIONAL MODES

At the beginning of each new training session, Cybersar simulator automatically generates the mission: the user can defines for every single operator which container has to be moved and the new position/destination to be reached.

The mission can be "multiple destinations", in other words the trainee could be asked for moving the container from a yard allocation or a truck trailer and back again. Also, different levels of complexity can be selected, extra-move included, based on the necessities; to this end the simulator provides the user with "interference" in term of means and persons.

By using C++ language intelligent algorithms have been implemented in Cybersar simulator models and devoted to track other federates during communication drawback. Due to this fact dynamics of physics are introduced in each federate model. In fact, containers and trailers properties are shared among different users, on the contrary trucks, cranes and people properties are functions of each platform user.

The high level of complexity of such procedures can be understood if we consider that during the simulation several kind of vehicles are involved; due to this the containers handling must include a continuous reassignment of the different attributes.

The adopted solution for the federation architecture will allow that different vehicles will be able to interact; this approach will support not only training, but also policy definition, procedure design and infrastructure reengineering. These operation are very important in a intermodal systems where the overall efficiency depend upon synergy and harmony among equipment, people and planning. The policy redesign was already experienced by the researchers in port ship handling by using virtual simulation; in logistics

intermodal operation it is also possible to get great benefits from this analysis; in effect the proposed synthetic environment allows to proceed in the redesign of the handling devices themselves; for instance it will be possible to change the virtual cockpit of a crane and to identify the benefits in term of overall logistics performances and safety levels through an experimental campaign on the simulators. During development phase it will be critical to complete proper VV&A (Verification, Validation and Accreditation); in effect it is necessary to check logical consistency of conceptual models and proper identification of interoperability factors among federates as well as detailed validation of each single federate and overall federation. The validation of the federation in term of proper implementation and correct tuning of the factors and parameters represent another critical step. Considering these aspects the researchers proceed in this process by applying standard IEEE 1516.3 High Level Architecture Federation Development and Execution Process as well as fundamentals of 5000.61 directive related to VV&A (Verification, Validation and Accreditation); obviously it will be necessary to tailor properly the VV&A on this specific case; this approach will guarantee to complete effectively the verification and validation of the overall system; the researchers use extensively a network of experts in simulation applied to port logistics and DIPTEM laboratories in order to guarantee the success of this phase; vice-versa the availability of real logistics operators will provide full validation of the redesign capabilities of the system in relation to virtual function test of scenarios involving both HIL (Hardware in the Loop, for instance automation systems and sensors on the crane spreader) and MIL (Man in the Loop, for instance policies for exchanging container between different cranes).

Figure 9: Virtual Yard Operations

CONCLUSIONS

The presented study represents a step forward in the field of simulation, allowing such a technique to be promoted in new areas and providing the final users with a low cost interactive distributed HLA-based environment.
This makes possible to extend the application of simulation as training support system in new sectors, effectively facing interaction, cooperation, competition in a wide range of scenarios, taking into consideration a significant cost saving as well.
The testing experience allows to validate the system by extensive training campaign.
Currently the research is in its preliminary phase, objective definition and architecture design, however, thanks to previous experiences (INNOVARE, RESET, SITRANET) the integration test is expected for next October.

REFERENCES
[1]. Bluemel E. (1997) "Managing and Controlling Growing Harbour Terminals", SCS Europe BVBA, Ghent, Belgium
[2]. Bontempi, Gambardella, Rizzoli (1997) "Simulation and Optimization for Management of Intermodal Terminals", Proc. of ESM97, Istanbul
[3]. Brandolini M., Viazzo S. (2004) "Simulation to Support Design and Training of Operative Procedures in Intermodal Terminals", Proceedings of HMS2004, Rio de Janeiro, Brazil, 25-30
[4]. Bruzzone A.G. (1995) "Fuzzy Logic and Genetic Algorithms Applied to the Logistical and Organisational Aspects of Container Road Transports", Proc. of ESM95, Praha, June 5-7
[5]. Bruzzone A.G., Kerckhoffs (1996) "Simulation in Industry", Genoa, Italy, October, Vol. I & II, ISBN 1-56555-099-4
[6]. Bruzzone A.G., M.E., Cotta G., Cerruto M.(1997) " Simulation & Virtual Reality To Support The Design

Of Safety Procedures In Harbour Environments ", Proceedings of ITEC97, Lausanne (CH), April 22-25

[7]. Bruzzone A.G., Giribone P. (1998) "Decision-Support Systems and Simulation for Logistics: Moving Forward for a Distributed, Real-Time, Interactive Simulation Environment", Proceedings of the Annual Simulation Symposium IEEE, Boston, 4-9 April

[8]. Bruzzone Agostino (1999) "Port Terminal Simulators as Main Supports for Design, Training & Management", Proc. of Port Logistics99, Alexandria, Egypt, Feb 14-16

[9]. Bruzzone A.G., Merkuryev Y.A., Mosca R. (1999) "Harbour Maritime & Industrial Logistics Modelling & Simulation", SCS Europe, Genoa, ISBN 1-56555-175-3

[10]. Bruzzone A.G, Mosca R., Revetria R., Rapallo S. (2000) "Risk Analysis in Harbour Environments Using Simulation", International Journal of Safety Science, Vol 35, ISSN 0925-7535

[11]. Bruzzone A.G., Mosca R., Revetria R. (2002) "Cooperation in Maritime Training Process using Virtual Reality Based and HLA Compliant Simulation", Proceedings of XVIII International Port Conference, Alexandria Egypt, January 27-29

[12]. Bruzzone A.G. (2002) "Supply Chain Management", Simulation, Volume 78, No.5, May, 2002 pp 283-337 ISSN 0037-5497

[13]. Bruzzone A.G., Itmi M. (2003) "Summer Computer Simulation Conference 2003", SCS International, San Diego, ISBN 1-56555-268-7 (887 pp)

[14]. De Ruit, Schuylebburg, Ottjes (1995) "Simulation of shipping traffic flow in the Maasvakte port area of Rotterdam", Proc. ESM95, Prague

[15]. Fleming D.K. (1997) "World Container Port Ranking", Maritime Policy and Management, Vol. 24, No. 2, pp. 175-181

[16]. Frankler E.G. (1987) "Port Planning and Development", John Viley and Sons, New York

[17]. Hayuth Y., Pollatschek M.A., Roll Y. (1994) "Building a Port Simulator", SIMULATION, vol. 63, no. 3, pp. 179-189

[18]. Koh P.H., Goh J.L.K., Ng H.S., Ng H.C. (1994) "Using Simulation to Preview Plans of a Container Port Operation", Proceedings of Winter Simulation Conference, Lake Buena Vista, Florida, December

[19]. Merkuryev Y., Bruzzone A.G., Merkuryeva G., Novitsky L., Williams E. (2003) "Harbour Maritime and Multimodal Logistics Modelling & Simulation 2003", DIP Press, Riga, ISBN 9984-32-547-4 (400pp)

[20]. Merkuriev Y., Bruzzone A.G., Novitsky L (1998) "Modelling and Simulation within a Maritime Environment", SCS Europe, Ghent, Belgium, ISBN 1-56555-132-X

[21]. Mosca R., P.Giribone & A.G.Bruzzone (1994) "Simulation & Automatic Parking in a Training System for Terminal Container Yard Management", Proceedings of ITEC94, The Hague, April 26-28

[22]. Nevins M.R., Macal C.M., Joines J. (1998) "A Discrete-Event Simulation Model for Seaport Operations", SIMULATION, vol. 70, no. 4, pp. 213-223, April

[23]. Ottjes J.A., Hengst S., Tuturima W.H. (1994) "A Simulation Model of a Sailing Container Terminal Service in the Port of Rotterdam", Proc. ESM94, Barcelona

[24]. Rizzoli A.E., Gambardella L.M., Bontempi G. (1997) "Simulation of an Intermodal Container Terminal to assist Management in the Decision-Making Process", Proc. of MODSIM9, International Congress on Modeling and Simulation, Hobart, Tasmania

[25]. Teo Y.M. (1993) "Simulation and Graphics Animation in Port Design", Proceedings of ESM93, Lyon, France

[26]. Thiers G., Janssens G. (1998) "A Port Simulation model as a Permanent Decision Instrument", SIMULATION, Vol. 71, no. 2, pp. 117-125, August

[27]. Villefranche L., Pecuchet J.P, Serin F. (1994) "Service Processes for Container Terminal Simulation", Proc. ESM94, Barcelona

Modelling and Simulation Methodologies

A NON-MODULAR CELLULAR DEVS MODEL OF THE DEGRADATION OF A CULTIVATED SOIL SURFACE BY RAINFALL

Gilles Valette[1,2], Stéphanie Prévost[1], Laurent Lucas[1]
[1]CReSTIC/SIC/MADS, EA3804
University of Reims Champagne-Ardenne, France
Email: laurent.lucas@univ-reims.fr

Joël Léonard[2]
[2]INRA UR1158 Agronomie
Laon-Reims-Mons, France
Email: leonard@inra.laon.fr

ABSTRACT

We aim to model and simulate the evolution of the surface structure of a cultivated soil surface during rainfall. The surface degradation is mainly the consequence of the creation and the transport of soil fragments, which are caused by the circulation of water, rainfall and runoff in particular. Our first intent was to use Cellular Automata (CA), but these processes cannot easily be modelled in a pure CA model because they are both discrete and continuous, local and global. We explain in this paper how non modular cellular DEVS can efficiently model this natural system and we present in detail the coupled model of the simulator and the atomic model of the terrain, and we give a sketch of the way we model the processes involved.

INTRODUCTION

The objective of our project, called SoDA (Soil Degradation Assessment), is to develop and validate a dynamic simulation of the evolution of the surface structure of a cultivated soil surface during a rainfall at the meter scale, keeping in mind a constant care for visualization. Major aspects of this evolution are the formation of soil crusts and the development of cracks (Valette et al., 2006a) which strongly influence water infiltration or seedling emergence. Therefore, such a model, able to predict soil structure under different initial soil conditions and climatic scenarios, would be a useful tool to select for example adequate tillage and sowing practices.

In the literature, most models predicting soil erosion generally do not aim to simulate the evolution of the soil surface and its structure, but rather only to predict soil loss, see for example (Lane and Nearing, 1989; Morgan et al., 1998; Favis-Mortlock et al., 2000). They often operate at rather large scale, and share the same basic principles and range of processes described, as well as the formalism used to describe these processes. For example, they almost have in common the principle of flow of water or material according to the greatest gradient, or the transport capacity concept. Although our simulator does not significantly differ from most models on that point, it presents several originalities arising mainly from the (small) scale considered and the fact we focus on the evolution of soil surface relief and structure rather than on soil loss. Our model considers explicitly a 3D

space, it allows the fragmentation of the soil into particles of different sizes which can be mobilized, projected by splash, transported or deposited by runoff, which permits a tracking of the granulometry of sediments. This is an important issue as it allows to describe a spatial (vertical and horizontal) and temporal evolution in the physical state of the soil, and interactions between this evolution and the processes at work. Finally, most processes descriptions have been revisited to include new important aspects or the most recent knowledge, and the description remains open so that our simulator could be a useful tool to test hypothesis and new modelling ideas.

A preliminary version of our work was presented in (Valette et al., 2006b). This model was based on Extended Cellular Automata (Avolio et al., 2003) but we were confronted with two main difficulties: (i) rainfall is better represented as a discrete events process than as a discrete time process; (ii) we needed the possibility of modelling a process in a global way, in order to keep the simulator as open as possible to new ideas. Thus, we changed our modelling approach by using Discrete EVents system Specification (DEVS), and in particular the concept of non modular cellular DEVS proposed in (Shiginah, 2006). The aim of the present paper is to present in detail this new model of soil degradation by rainfall.

BACKGROUND

Informal description

We study the evolution of a portion of land of metric size which is submitted to a rainfall, real or simulated. This evolution is governed by the three types of water transfer: rain, infiltration, runoff. We don't take into account the effect of evaporation because it is negligible during rainfall. The three processes transport water but rainfall and runoff have also an effect on the soil: rainfall can break the soil into fragments and project them, runoff can mobilize, transport or deposit fragments (Fig. 1).

These three processes are very comparable in their effects on the soil, but they are very different by their nature: when they occur, infiltration and runoff are continuous phenomena, whereas rain is a succession of arrivals of raindrops, each one being localized precisely in time and in space. That is why we chose to model the first ones as discrete time processes, and the third one as a discrete events process (Fig. 2). Another difference between these

	Rain	Infiltration	Runoff
Water	Transport	Transport	Transport
Soil fragments	Detachment	Detachment	Detachment
	Mobilisation	Mobilisation	Mobilisation
	Transport	Transport	Transport

Figure 1: Description of the processes governing the evolution of the soil.
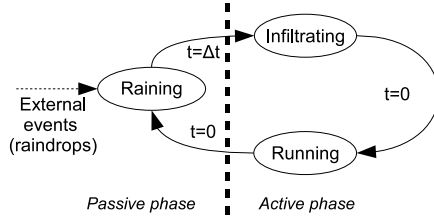

Figure 2: The event scheduling graph of the processes.

processes is their localization: infiltration and runoff can be modeled by using local information, i.e., the state of the immediate neighbours of a cell, whereas the transport of fragments by splash (i.e., the impact of a raindrop) may concern distant cells. Finally, the time step required to simulate these different processes may differ by one or two orders of magnitude.

Cellular Spaces and DEVS
In order to model a system in a spatialized way, cellular spaces are of great interest. When the behaviour of the system depends on local interactions, like diffusion, Cellular Automata (CA) can be used, and many examples can be found in the literature, in various domains. Nevertheless, natural phenomena are often complex behavioral dynamic systems and cannot easily be modelled with a standard CA model. For this reason, many works were dedicated to improve simulation based on CA. In order to simulate different phenomena like lava and debris flows, landslides or contaminated soils bioremediation, Di Gregorio et al. (Di Gregorio and Serra, 1999; Di Gregorio et al., 1999) proposed the principle of Macroscopic CA, with four main novelties: (i) for each cell, a nearly unlimited number of states is permitted, each state being composed of substates, with possibly continuous values; (ii) the transition function is split in several parts, each one corresponds to an elementary process of the macroscopic phenomenon; (iii) each elementary process may have its own neighbourhood and internal transformations are allowed; (iv) substates of type "outflow" are used in order to account for quantities moving from a cell toward another one in the neighbouring. This model assumes that the whole phenomenon can be described by sequentially calculating internal transformations and local interactions. Avolio et al. (Avolio et al., 2003) developed this empirical approach and used Extended CA in order to model surface flows, taking into account "external influences" which cannot be described in terms of local rules.

We used Extended CA to describe a preliminary version of our simulator (Valette et al., 2006b). However two difficulties raised: (i) Extended CA does not offer a way to rigorously specify these external influences which are simply considered as special or additional functions; (ii) these influences must be treated as discret time processes, thus, they consider the state of the cells at the end of the previous iteration in order to compute the new state of the cells. In the case of the rain, it is obvious that the arrival of a raindrop is better represented as a discret event: a raindrop modifies the state of some cells, and this state must be taken into consideration for the next event. Asynchronous CA (Schönfisch and de Roos, 1999) can model such behaviour, but runoff and infiltration should then be also considered as discrete events, although they are better modeled as discret time (i.e., synchronous) CA.

For these reasons, we chose to use the Discrete EVent system Specification (DEVS) formalism. This formalism was introduced by Zeigler (Zeigler, 1976), in order to allow specifying simulation programs independently of a particular language. In (Zeigler, 1984) it was extended to enable constructing discrete event simulation models in a hierarchical and modular manner. At the lowest level, an "atomic" DEVS describes the autonomous behaviour of a discrete-event system with three transition functions: an internal transition function δ_{int} which determines the transitions between sequential states, an external transition function δ_{ext} which describes how the system reacts to external input (events), and an output function λ which is in charge of the generation of output (events). An advantage of DEVS is this separation between internal and external transition functions, which permits to specify in an independant way the behaviour of the system in case of the presence, or absence, of an external influence — that is exactly what we needed. At the higher level, a coupled DEVS describes a system as a network of "coupled" components. These components are either atomic DEVS or, thanks to the property of closure under coupling, coupled DEVS. To reproduce the interactions between components, a component communicates via ports to send outputs to other components and to receive inputs from components. The parallel version of DEVS, Parallel DEVS (P-DEVS), was introduced in (Chow and Zeigler, 1994). This model preserves hierarchical and modular construction properties of DEVS and permits to introduce Cellular DEVS (Zeigler et al., 2000) and Cell-DEVS (Wainer and Giambiasi, 2001b) which integrated the theories and algorithms of CA in DEVS.

In conventional cellular DEVS approaches, the cell space is implemented as a coupled DEVS model that contains a number of cells that are arranged in an array, each cell being considered as an atomic DEVS model which communicates with its neighbours through the system of ports. In case of a great number of cells (e.g., in our case, a typical cell space contains 200x200x10 cells), the inter-cell communication generated during simula-

tion represents a huge volume of messages and can be very time consuming. To overcome this drawback, in (Wainer and Giambiasi, 2001a) the coordinator hierarchy of the simulator is flattened. In (Muzy and Nutaro, 2005) unnecessary coordinator objects are eliminated and scheduling algorithms deal with active cells only, like in (Hu and Zeigler, 2004). These approaches consider the implementation level and keep treating each cell as an atomic model. The work presented in (Shiginah, 2006) takes advantage of these enhancements and applies similar methods to the model development level, giving as a result a non-modular cellular space representation, equivalent to a P-DEVS atomic model, which simplifies the modelling process.

Following Shiginah's approach, we consider our soil degradation model as a coupled P-DEVS model, in which the terrain is a cellular space, considered as an atomic P-DEVS model. This allows simpler specification and faster simulation. Another advantage is that DEVS naturally permits to consider raindrops as external discrete events, whereas infiltration and runoff can be yet considered as internal synchronous transitions. Moreover, the split of the transition function in elementary processes, which is an extension proposed in Macroscopic CA, is explicitly allowed in DEVS by using different phases of the model, each one corresponding to one process and to one function. Each function may have its own rules, including the neighbourhood. Finally, as we consider the cellular space in a non modular way, we allow a global knowledge of the cell space to compute the next state of a cell.

DETAIL OF OUR MODEL

Before detailing the coupled and the atomic models of our simulator, we present in the next section the structure of the terrain.

The structural model of terrain

The terrain is discretized into a regular 3D grid. The user can choose the dimension of the cells, which are geometrically identical rectangular parallelepipeds. Typically, we use cubic cells of 2 mm side, for two reasons. First, 2 mm is the resolution of the digital elevation maps we obtain by laser rugosimetry, and it is natural to keep this original information unchanged for the width and the length of the cells. Second, the biggest soil fragments we consider have a mean diameter comprised between 1 mm and 2 mm, thus a depth of 2 mm is necessary and sufficient to insure that the cells can contain these fragments.

As shown in Fig. 3, our structural model of terrain consists of one 3D cellular grid and five outlets. These outlets are adimensional cells which keep information of the quantities of water and fragments coming from the boundary cells of the terrain. There is one outlet for each cardinal direction and another for the bottom. There is no need for a top outlet because neither the water nor the fragments go to the atmosphere.
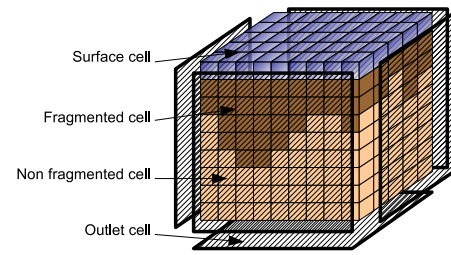

Figure 3: The structural model of terrain.

The cellular grid contains 3 types of cells:

- non-fragmented cells are cells which do not contain any soil fragments; they contain infiltrated water, matter which constitutes the original soil, considered as continuous, and the cumulated kinetic energy propagated vertically from the surface to the subsoil by the impact of raindrops;

- fragmented cells are non-fragmented cells which have been impacted by a raindrop or have received fragments from another cell, by splash or runoff; they have the same information than the non-fragmented cells, plus the volume of particles corresponding to each of 7 classes of particles, ranging from 0-20 μm to 1-2 mm;

- surface cells compose the top layer and contain the same information than the fragmented cells, plus 2D information: height of the water surface, height of the terrain at the center of the cell, and the volume of particles present in the flow above the cell for each of the size classes.

The coupled P-DEVS model

We use the concept of experimental frame in order to separate the experiments and the model itself. The experimental frame strategy (Zeigler, 2000) treats the model as a black box that is tested and it reflects the objectives of the experimenter who performs experiments on a real system or, through simulation, on a model (Vangheluwe, 2000). The same experimental frame can be used to test different models, or a model can be tested with different experimental frames. We use both of these possibilities, because we want our simulator to be able of producing realistic results but, in the same time, to be a way of testing new ideas about the processes involved. The experimental frame is decomposed in Acceptor, Transducer and Generator (Fig. 4).

The Acceptor produces an unique output which indicates if the simulation continues or not. To make its decision, it can compare the outputs generated by the model with expected values controls, if any.

The Transducer receives as input the state of the model and applies transformations for meaningful interpretation by the user or the Acceptor. One of the tasks of the Transducer, in our case, is to produce images or animations from the state of the cellular space. It is an important
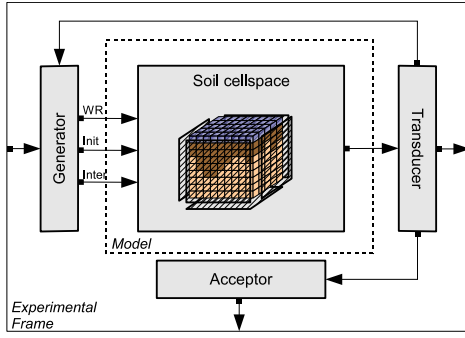
Figure 4: The coupled P-DEVS model of the soil degradation by rainfall.

issue because direct visual observation is a method of validation. We thus want to provide realistic or informative visual representation of the numerical results obtained during a simulation, in order to allow visual comparisons between the model and the real system for example. For that we use mesh rendering and volumic visualization with OpenGL.

The Generator describes the inputs or stimuli applied to the system or model during an experiment. In our case, we separate these inputs in 3 categories, which correspond to three ports (Fig. 4): initialization, interaction and water/rain generation. The initialization permits to define the conditions of the virtual experiment, including time step, values of parameters used in the equations of the transition functions, characteristics of the soil (topography, initial water content, size of the particles,...) It allows the user to change the model itself, by choosing the behaviour of the transition functions. For example, the user can inhibit infiltration to get an impermeable soil, or choose between different formalisms to calculate the splash distance. The interaction allows the user to send inputs to the model during the simulation, e.g., adding fragments on a cell of the terrain, changing the flow of a water source, which extends the possibilities of the numerical experiments. Finally, water/rain generation is responsible for adding water to the terrain or to let raindrops fall on it. The output is in this case composed by two indications: the coordinates (x,y) of the cell which will receive the water, and the quantity of water, i.e., the diameter d of a spherical raindrop or the height of water h to add on a cell. The Generator is able to reproduce a constant flow coming from one boundary, or a single drop falling in the center of the terrain. Of course, it can reproduce a simulated rain with respect to a given raindrop size distribution, limited or not to a portion of the terrain. Finally, it can simulate a real rain, using a hyetogram, and assuming a gamma distribution of raindrop sizes, which has been shown to be a good model of the raindrop arrival process (Uijlenhoet and Stricker, 1999). The main advantage of the discrete representation of raindrops is to permit an explicit discrimination between the effects of rain intensity and raindrop size.

Thus, the coupled P-DEVS model SD of the soil

degradation by rainfall is defined by the following structure, illustrated in Fig. 4:

$$SD = \langle X,Y,D,\{M_d\},\{I_d\},\{Z_{id}\}\rangle$$

- X is a set of the input values, which are indications sent by the external world, e.g., the user interface, to the Generator.

- Y is a set of the output values, coming from the Transducer (e.g., images, plots) or the Acceptor (yes/no).

- $D = \{g,t,a,s\}$ is a set of components references.

- for each d in D, M_d is a component, i.e., a P-DEVS model: M_g is the Generator, M_t is the Transducer, M_a is the Acceptor and M_s is the Soil.

- for each d in $D \cup \{SD\}$, I_d contains the influencees of d (d cannot be in I_d): the influences of the model are represented by the arrows in Fig. 4.

- for each i in I_d, Z_{id} is a function, the i-to-d output-to-input translation.

Atomic model of the terrain

In (Shiginah, 2006), there is a proof that a non-modular cellular space keeps the general P-DEVS structure and hence equivalency, the former model having more details and parameters which can be implied in the internal behavior of any P-DEVS atomic model. Thus, we use the specification of a P-DEVS model to describe the atomic non-modular cellular model T of the terrain:

$$T = \langle X,S,Y,\delta_{int},\delta_{ext},\delta_{con},\lambda,ta\rangle$$

- X is the set of the input values coming from the Generator, i.e., for the port "WR" the values $\langle(x,y),d,h\rangle$, determining the location (x,y) on the surface of the terrain and the diameter d of the impacting raindrop, or the height h of the incoming water.

- Y is the set of the output values, i.e., the state of the terrain, transmitted to the Transducer.

- S is the set of general states of the atomic model, i.e., $\{\{\text{"}active\text{"},\text{"}passive\text{"}\} \times S^*\}$, S^* being the set of values s of the variables contained in the cells. The model is "$passive$" when it is waiting for external events, and becomes "$active$" when it has to manage infiltration and runoff (Fig. 2).

- $\delta_{int} : S \to S$ is the internal transition function, i.e., the succession of the infiltration $\mathcal{I}$ and the runoff $\mathcal{R}$ functions:
  $\delta_{int}(\text{"}passive\text{"},s) = (\text{"}active\text{"},\mathcal{I}(s))$
  $\delta_{int}(\text{"}active\text{"},s) = (\text{"}passive\text{"},\mathcal{R}(s))$

- $\delta_{ext} : Q \times X^b \to S$ is the external transition function, where X^b is a set of bags over elements in X, i.e., incoming water or raindrop and their location, and $Q = \{(s,e) \mid s \in S,\ 0 < e < ta(s)\}$, e being the elapsed time since last state transition. This function is fired in response of the arrival of a raindrop, thus it is the splash function $\mathcal{S}_p$:
  $\delta_{ext}(\text{``passive''}, s, e, x^b) = (\text{``passive''}, \mathcal{S}_p(s, x^b))$.

- $\delta_{con} : S \times X^b \to S$ is the confluent transition function and is fired when a raindrop arrives at Δt (the time step) and has a standard behaviour:
  $\delta_{con} = \delta_{int} \circ \delta_{ext}$,
  meaning in our case that we execute the splash function $\mathcal{S}_p$ before infiltration $\mathcal{I}$:
  $\delta_{con}(\text{``passive''}, s, e, x^b) = (\text{``active''}, \mathcal{I} \circ \mathcal{S}_p(s, x^b))$.

- $\lambda : S \to Y^b$ is the output function, called before the internal transition function, and it simply transmits the state of the terrain to the Transducer when the phase is "*passive*":
  $\lambda(\text{``passive''}, s) = s$
  $\lambda(\text{``active''}, s) = \emptyset$

- $ta : S \to \mathbb{R}^+$ is the time advance function:
  $ta(\text{``passive''}, s) = \Delta t$
  $ta(\text{``active''}, s) = 0$

The next section briefly presents the principles of the transition functions which reproduce the elementary processes.

PRINCIPLES OF TRANSITION FUNCTIONS

Splash

The splash function $\mathcal{S}_p$ has five consequences on the terrain: (i) it adds water on some cells; (ii) it possibly detaches fragments from these cells (fragmentation of the soil); (iii) it mobilizes already present fragments; (iv) it projects fragments; (v) it transfers raindrop kinetic energy to the impacted surface cells and also to the corresponding subsoil cells. Water is redistributed according to the projected area of the raindrop over the target cells and its neighbours, taking into account the non spheroidal shape of raindrops for diameters above 2 mm (Beard and Chuang, 1987). Each drop detaches a quantity of soil which is proportionnal to its kinetic energy above a threshold (Sharma et al., 1991) and which decreases exponentially with the ratio of flow depth to raindrop size (Kinnell, 2005). We use results from aggregate stability tests to determine how to distribute the detached mass amongst the different particle size classes (Legout et al., 2004). All the newly created fragmented particles are mobilized and thus projected. The splash distance depends on the size of the fragments (Legout et al., 2005) and on the local slope (Furbish et al., 2007). The projection direction from the center of impact is randomly chosen from a probability density function which depends on local slope (Furbish et al., 2007). Finally, the transfer of kinetic energy to the downward cells is based on

a decreasing function of the depth to raindrop radius ratio. The density of each cell then evolves as a function of both the cumulated kinetic energy and the granulometry.

Infiltration

The infiltration function $\mathcal{I}$ removes water from the surface and adds water in the void space of the cells. Three different models of infiltration were implemented. We used first a CA-based version of the Richard's equation (both in full 3D and 1D vertical for each column of cells) based on a report algorithm between adjacent cells. The main drawback of this method (in addition to the fact it is very time consuming) was that it necessitated to extend the cellular domain well deeper than the wetting front to operate. It was not always easy to determine the correct extension before the simulation, and simulation time grows rapidly as we multiply the number of cells. We also tested the Green-Ampt approximation (Green and Ampt, 1911), with an "1D vertical for each column of cells" implementation. The results obtained with both models were compared with success, in a simple situation, to results from a numerical resolution of the Richards' equation. However the classical Green and Ampt model was not adapted to the description of flow through a crust whose properties, in addition, change through time. We thus implemented a third infiltration model, much more specific to flow through a crust, which allows a simple, quick and accurate description of infiltration (excepted at the very beginning of the simulation). In this implementation, we assume a steady state flux through the crust and the corresponding subsoil during an iteration, which allows to estimate from the hydrodynamic properties of the subsoil and the thickness and hydraulic conductivity of the crust the pressure at the soil crust interface, and thus the flux through the crust. The saturated hydraulic conductivity of the crust is estimated by the harmonic mean of the saturated hydraulic conductivity of the crusted cells, the conductivity of the cells depending on their density (which evolves with the granulometry and the cumulated kinetic energy). This method necessitates a definition of the crusted cells: we consider a cell as crusted when its saturated hydraulic conductivity is below a fraction of the saturated hydraulic conductivity of the initial soil. It is interesting to notice that this infiltration model is not based on local interaction, thus that it cannot be modeled with CA, which is not an issue with the current P-DEVS model.

Runoff

The runoff function $\mathcal{R}$ transports water and soil fragments after their mobilization. The transport of the soil fragments is determined by the transport of water. Thus, this function has to determine first how much water is transferred and where it is transferred, and then decides from this information how many particles, from which classes, are transported and where they are transported. In that purpose, we take into account the flow depth and the altitude of the soil surface, the sum of these two

quantities being used to define the total hydraulic head, neglecting the kinetic component which is always very small in our study context. We transfer water from one source cell to a unique target cell only, in the direction of the highest head gradient. We treat all the cells, from the lowest head cell to the highest head cell, as potential receivers of water emitted by their neighbours. When a cell emits water, it is marked and is no more eligible to transmit water for this iteration of the process. The flow from a source cell to a target cell is calculated using the Darcy-Weisbach equation (Chow et al., 1988). One main issue in this report algorithm is that the quantity of water transferred to a cell must not be higher than the difference in hydraulic head between the source cells and the target cell, because the target cell might become a source cell at the next iteration, and so on, causing instability. For this reason we calculate a maximum quantity of water that a cell can receive from each neighbour by means of an equilibrating algorithm which is detailed in (Valette et al., 2008). In order to validate this model, we have compared the results of a simulation on a simple impermeable slope to results obtained using a numerical solution of the classical Saint Venant's equations (Zhang and Cundy, 1989) on the same terrain, and both results are in close agreement.

Water can transport sediment in several ways. The total sediment load of flowing water can be subdivided into bedload and suspension load, which is important because these modes of transport are not equally sensitive to the local topography. In a cell which emits water, we first calculate the local shear stress, which depends on hydraulic conditions. Then, to decide how particles from a class will be transported, we use the Rouse number, defined as the ratio of particle settling velocity to the shear velocity (i.e., rate of fall versus strength of turbulence acting to suspend particles). The critical shear stress (which corresponds to the beginning of the motion of the particles) of particles from that class is calculated from the Shields' curve and adjusted to take into account the effect of slope (gravity and effect of the particle diameter to flow depth ratio (Lamb et al., 2007)). Sediment transport is then estimated from the excess shear stress using an empirical formula (Julien, 1998). It is worth noticing that we take also into account the effect of lateral erosion on a dry cell in the neighbouring of a cell emitting water.

CONCLUSION

We develop a simulator of the evolution of soil surface structure of cultivated soils under rainfall, at the meter scale. This is an important issue as it has both theoretical and practical interest. In this paper, we have detailed the model of our simulator, based on the P-DEVS formalism which permits to give a concise, not restrictive yet rigorous specification of its structure and its behaviour. Infiltration and runoff are modeled as discrete time processes, whereas raindrops arrivals are considered as external discret events. The models of the different processes in-

volved were validated by comparison with numerical solutions of continuous systems or with results of real experiments available in the literature. We plan to complete the results with our own experiment of rain simulation on a portion of soil, in order to calibrate the parameters of the simulator and to validate its global behaviour. Finally, we want to develop an algorithm to predict the nature of the crust close to the visual characterization made by an observer in a field.

REFERENCES

Avolio, M. V., Crisci, G. M., D'Ambrosio, D., Di Gregorio, S., Iovine, G., Rongo, R., and Spataro, W. (2003). An Extended Notion of Cellular Automata for Surface Flows Modelling. *WSEAS Transactions on Computers*, 2:1080–1085.

Beard, K. V. and Chuang, C. (1987). A new model for the equilibrium shape of raindrops. In *Journal of the Atmospheric Sciences*, volume 44, pages 1509–1524.

Chow, A. C. H. and Zeigler, B. P. (1994). Parallel DEVS: a parallel, hierarchical, modular, modeling formalism. In *WSC'94: Proceedings of the 26th conference on Winter simulation*, pages 716–722, San Diego, CA, USA. Society for Computer Simulation International.

Chow, V., Maidment, D., and Mays, L. (1988). *Applied Hydrology*. Water Resources and Environmental Engineering. McGraw-Hill, New-York.

Di Gregorio, S. and Serra, R. (1999). An Empirical Method for Modelling and Simulating Some Complex Macroscopic Phenomena by Cellular Automata. *Future Generation Computer Systems*, 16(2–3):259–271.

Di Gregorio, S., Serra, R., and Villani, M. (1999). Applying cellular automata to complex environmental problems. *Theoretical Computer Science*, 217:131–156.

Favis-Mortlock, D., Boardman, J., Parsons, A., and Lascelles, B. (2000). Emergence and Erosion: a Model for Rill Initiation and Development. *Hydrological Processes*, 14(11-12):2173–2205.

Furbish, D. J., Hamner, K. K., Schmeeckle, M., Borosund, M. N., and Mudd, S. M. (2007). Rain splash of dry sand revealed by high-speed imaging and sticky paper splash targets. *Journal of Geophysical Research-Earth Surface*, 112.

Green, W. H. and Ampt, G. A. (1911). Studies on Soil Physics Part I: The Flow of Air and Water Soils. *Journal of Agricultural Science*, 4(1):1–24.

Hu, X. and Zeigler, B. P. (2004). A high performance simulation engine for large-scale cellular DEVS models. In *High Performance Computing Symposium (HPC'04)*, pages 3–8. Advanced Simulation Technologies Conference.

Julien, P. Y. (1998). *Erosion and Sedimentation*. Cambridge University Press.

Kinnell, P. I. A. (2005). Raindrop-impact-induced erosion processes and prediction: a review. *Hydrological Processes*, 19(14):2815–2844.

Lamb, M. P., Dietrich, W. E., and Venditti, J. G. (2007). Is the critical shields stress for incipient sediment motion slope dependent? *Journal of Geophysical Research - Earth Surface*.

Lane, L. and Nearing, M. (1989). USDA - Water Erosion Prediction Project: Hillslope Model. Technical Report 2, USDA-ARS National Soil Erosion Research Laboratory, NSERL, West Lafayette, Indiana, USA.

Legout, C., Leguédois, S., and Le Bissonnais, Y. (2004). Aggregate Breakdown Dynamics Under Rainfall Compared With Aggregate Stability Measurements. *European Journal of Soil Science*, 56(2):225–238.

Legout, C., Leguédois, S., Le Bissonnais, Y., and Malam Issa, O. (2005). Splash Distance and Size Distributions for Various Soils. *Geoderma*, 124(3–4):279–292.

Morgan, R., Quinton, J., Smith, R., Govers, G., Poesen, J., Auerswald, K., Chisci, G., Torri, D., and Styczen, M. (1998). The European Soil Erosion Model (EUROSEM): a Dynamic Approach for Predicting Sediment Transport from Fields and Small Catchments. *Earth Surface Processes and Landforms*, 23(6):527–544.

Muzy, A. and Nutaro, J. J. (2005). Algorithms for efficient implementations of the DEVS & DSDEVS abstract simulators. In *1st Open International Conference on Modeling & Simulation (OICMS)*, pages 273–279. ACM Press.

Schönfisch, B. and de Roos, A. (1999). Synchronous and asynchronous updating in cellular automata. *BioSystems*, 51:123–143.

Sharma, P. P., Gupta, S. C., and Rawls, W. J. (1991). Soil detachment by single raindrops of varying kinetic energy. *Soil Science Society of America Journal*, 55:301–307.

Shiginah, F. A. S. B. (2006). *Multi-Layer Cellular DEVS Formalism for Faster Model Development and Simulator Efficiency*. PhD thesis, Electrical and Computer Engineering Dept., University of Arizona.

Uijlenhoet, R. and Stricker, J. N. M. (1999). A consistent rainfall parameterization based on the exponential raindrop size distribution. *Journal of Hydrology*, 218(3-4):101–127.

Valette, G., Prevost, S., and Lucas, L. (2006a). Modeling and visualization of cracks in a desiccating soil. In *11th International Fall Workshop Vision, Modeling, and Visualization*, pages 177–184, Aachen, Germany. Aka Verlag, Berlin.

Valette, G., Prévost, S., Lucas, L., and Léonard, J. (2006b). SoDA Project: a Simulation of Soil Surface Degradation by Rainfall. *Computers & Graphics*, 30(4):494–506.

Valette, G., Prévost, S., Lucas, L., and Léonard, J. (2008). A Discrete Modelling of Soil Fragments Transport by Runoff. In *14th Discrete Geometry for Computer Imagery (DGCI 2008)*, Lyon, France. Springer Verlag.

Vangheluwe, H. (2000). *Multi-Formalism Modelling and Simulation*. PhD thesis, Ghent University.

Wainer, G. and Giambiasi, N. (2001a). Application of the Cell-DEVS Paradigm for Cell Spaces Modelling and Simulation. *Simulation*, 76(1):22–39.

Wainer, G. and Giambiasi, N. (2001b). Timed cell-DEVS: modeling and simulation of cell spaces. *Discrete event modeling and simulation technologies*, pages 187–214.

Zeigler, B. P. (1976). *Theory of Modelling and Simulation*. Wiley-Interscience, New York.

Zeigler, B. P. (1984). *Multifacetted Modelling and Discrete Event Simulation*. Academic Press, London.

Zeigler, B. P. (2000). Verification and validation of DEVS models: Applying the theory of modeling and simulation to the needs of simulation based acquisition. In *Proceedings of Summer Computer Simulation Conference*.

Zeigler, B. P., Kim, T. G., and Praehofer, H. (2000). *Theory of Modelling and Simulation*. Academic Press, San Diego, CA, USA.

Zhang, W. and Cundy, T. W. (1989). Modeling of two-dimensional overland flow. *Water Resources Research*, 25:2019–2035.

AUTHOR BIOGRAPHIES

LAURENT LUCAS received the PhD degree in Computer Science in 1995 from the University of Reims, France. He is a professor in the Department of Computer Science at the University Institute of Technology of Reims. His research interests include scientific visualization, geometric modelling and cooperation between image processing and image synthesis. His email is `laurent.lucas@univ-reims.fr`.

JOEL LEONARD received the PhD degree in Hydrology in 2000 from the University of Paris 6. He is a searcher of INRA (National Institute of Agricultural Research). His research interests include infiltration and runoff, relationship between water circulation and physical state of the soil, determinism of soil erosion and its relationship with soil structure. His email is `leonard@inra-laon.fr`.

STEPHANIE PREVOST received the PhD degree in Computer Science in 2001 from the University of Reims, France. She is an assistant professor in the Department of Computer Science at the University Institute of Technology of Reims. Her research interests include scientific visualization, geometric modelling and simulation of natural phenomena. Her email is `stephanie.prevost@univ-reims.fr`.

GILLES VALETTE is a PhD student in the University of Reims Champagne-Ardenne, France. He is in the third year of thesis in the CReSTIC-SIC, a laboratory of research in Computer Science, and in the Laon-Reims-Mons Agronomy Research Unit of INRA. His work concerns the simulation and the visualization of the evolution of the structure of a cultivated soil under rainfall. His email is `gilles.valette@leri.univ-reims.fr`.

ON STABILITY AND MODEL ORDER REDUCTION OF PERTURBED NONLINEAR NEURAL NETWORKS

Marissa Condon[1] and **Georgi G. Grahovski**[1,2]

[1] School of Electronic Engineering, Dublin City University, Glasnevin, Dublin 9, IRELAND,
[2] Institute for Nuclear Research and Nuclear Energy, Bulgarian Academy of Sciences,
72 Tsarigradsko chaussée, 1784 Sofia, BULGARIA
Email: `marissa.condon@dcu.ie` (M. Condon)   `grah@eeng.dcu.ie` (G. Grahovski)

KEYWORDS

model reduction, balanced truncation, neural networks, perturbed nonlinear systems

ABSTRACT

In this paper, the qualitative theory of large-scale dynamical systems is surveyed. In particular, the focus is the Hopfield Neural networks both with and without perturbations. Properties relating to asymptotic and exponential stability and instability are detailed. A model reduction technique based on balanced truncation is applied to the neural networks. Its effect on the stability properties of the networks is then examined. A numerical test illustrates some important points.

1. INTRODUCTION

Nonlinear neural networks such as the Hopfield neural network [Hopfield (1982)], [Hopfield et al. (1986)], [Denker (1986)] are powerful computational systems for a wide variety of applications [Borisyuk et al. (2005)], [Edelstein-Keshet (1986)]. This lies in the fact that they are extremely robust to malfunctions. Their dynamic behaviour exhibits stable states and this is advantageous. For example, a time evolution of an array of neuron-like elements towards equilibrium points can be viewed as the evolution of an imperfect pattern towards a correct stored pattern. As stated in [Hinton et al. (Eds)], this is similar to the storage of information in an associative memory [Kohonen (1984)].

In most of the applications involving neural networks, the model equations form a large-scale system (see e.g. ([Pomerlau et al. (1988)], [Iwata et al. (1990)], [Griffin et al. (1991)] and [Wawrzynek et al. (1993)]) and consequently, this leads to costly and inefficient computations. Therefore, model reduction is of paramount importance. However, its effect on the stability properties of the system is crucial in determining its usefulness. The reduced model must mirror the properties of the original system if it is to be of practical utility. Furthermore, the behaviour of the reduced system subject to perturbations must also match that of the original system.

Nonlinear model reduction has increasingly become a focus of research as in general, linear models are inadequate to describe real-world processes. While nu-merous approaches for linear model reduction have been proposed [Antoulas (2003)], there is a dearth of effective nonlinear model reduction techniques. Balanced truncation, as pioneered by Moore [Moore (1981)], is a very effective linear model reduction technique and consequently, it has been extended by several authors for nonlinear systems. For example, Scherpen introduced the notion of controllability and observability functions to generalise the controllability and observability gramians which characterise linear systems [Scherpen (1993)]. However, their calculation is computationally expensive and their use is hence restricted [Scherpen (1993); Gray et al. (1998)]. To counteract this, empirical gramians have been proposed by several authors [Lall et al. (2002)], [Hahn et al. (2002a)] and [Condon et al. (2004)]. It is the technique in [Condon et al. (2004)] that is adopted in this work. In particular, the paper will examine the effect of the application of empirical balanced truncation on the stability properties of both unperturbed and perturbed neural networks.

The present paper looks at the robustness of neural networks to perturbations and examines if the related properties are preserved with empirical balanced truncation.

The paper is organized as follows: In section 2, the Hopfield neural network model is described briefly. In Section 3, the stability properties of neural networks are sketched. Then, in Section 4, the robustness properties of perturbed neural networks are reviewed, while the model reduction technique is reviewed in Section 5. The preservation of the robustness qualities in a reduced model is addressed in Section 6. These results are illustrated with a numerical example in Section 7.

2. NONLINEAR NEURAL NETWORK

Consider the following system of non-linear ODE's (known as Hopfield models [Hopfield (1982)], [Hopfield et al. (1986)) in the form:

$$\dot{u}_i = -b_i u_i + \sum_{j=1}^{n} A_{ij} G_j(u_j) + U_i(t), \qquad (1)$$

where $i = 1, \ldots, n$, A_{ij} and b_i are constants, $U_i(t)$ are functions of the time variable t. Model order reduction is applied to the model equations (1), and the paper studies the qualitative behaviour of the solutions of the reduced
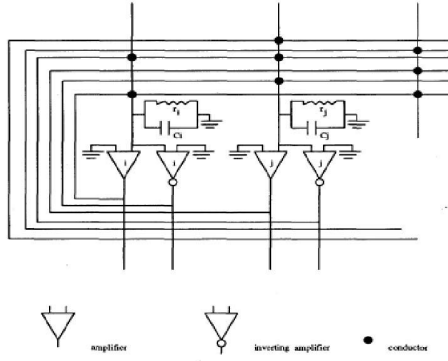
Figure 1: Hopfield neural network model

perturbed model (see eqn.(17) below) near equilibrium points (the positions where $\dot{u}_i = 0$, $i = 1, \ldots, n$). By setting the external inputs $U_i(t)$, $i = 1, \ldots, n$, equal to zero, $u^* = [u_1, \ldots, u_n]^T \in \mathbb{R}^n$ is defined to be an equilibrium for eqn. (1). The locations of such equilibria in $\mathbb{R}^n$ are determined by the interconnection pattern of the neural network (i.e., by the parameters A_{ij}; $i, j = 1, \ldots, n$) as well as by the parameters b_i, and the nature of the nonlinearities $G_i(\cdot)$, $i = 1, \ldots, n$.

It is frequently assumed also, that

$$G_i(u_i) = a_i \arctan(\lambda_i u_i), \tag{2}$$

for some $a_i > 0$ and $\lambda_i > 0$.

In neural network applications, usually the input currents $I_i(t)$ are held constant over a time interval of interest, so it is customary to define an equilibrium as a point $u^* \in \mathbb{R}^n$ having the following property:

$$-b_i u_i^* + \sum_{j=1}^n A_{ij} G_j(u_j^*) + c_i = 0, \tag{3}$$

where c_i, $i = 1, \ldots, n$ are time-independent constants, not necessarily equal to zero. Note that in this case, the locations of such equilibria in $\mathbb{R}^n$ are determined by the interconnection pattern of the neural network (i.e., by the parameters A_{ij}; $i, j = 1, \ldots, N$), by the parameters b_i, the nature of the nonlinearities $G_i(\cdot)$, and the constant inputs c_i, $i = 1, \ldots, N$.

In this study of the stability properties of such models, it will be assumed also that a given equilibrium point u^* is an isolated equilibrium point for eqn. 1, i.e., there exists an $r > 0$ such that in the neighbourhood $B(u^*, r) = \{(u(t) - u^*) \in \mathbb{R}^n : |u(t) - u^*| < r\}$ no equilibrium for eqn. 1, other than $u = u^*$, exists. It is shown in [Mitchel et al. (1989)], [Li et al. (1986)] that this assumption is a reasonable one for the case of the systems considered herein. When analyzing the stability properties of a given equilibrium point, it is possible to assume, without loss of generality, that this equilibrium is located at the origin of $\mathbb{R}^n$. By appropriate coordinate transformation, one can always map the equilibrium point u^* into the origin.

It is often convenient to view a system in the form of (1) as an interconnection of N free subsystems (or isolated subsystems) described by equations of the form:

$$\dot{p}_i = -b_i p_i + A_{ii} G_i(p_i^*) + U_i(t), \tag{4}$$

where $i = 1, \ldots, n$. From this viewpoint, the terms

$$g_i(x_1, \ldots, x_n) = \sum_{\substack{j=1 \\ j \neq i}}^n A_{ij} G_j(x_j^*), \tag{5}$$

make up the interconnecting structure of the system in (3).

3. STABILITY OF NONLINEAR NEURAL NETWORKS

3.1 Stability of Free Subsystems

Prior to describing the stability properties of the entire neural network, one will consider a brief sketch of the stability results concerning the individual free subsystems, described by eqn. (4).

Below, there is a sketch of the basic results on the various stability properties of the equilibrium $p_i = 0$ of the the model equations (4) for the case of vanishing external inputs $U_i(t) = 0$. The precise $\epsilon - \delta$ definitions of such concepts can be found, e.g., in [Miller et al. (1982)].

For any p_{i0}, sufficiently close to $p_i = 0$, if the solutions $\phi_i(t, t_0, p_{i0})$ of (4) with zero external inputs, remain close enough to the equilibrium $p_i = 0$, then the equilibrium $p_i = 0$ of (4) will be *stable* for $U_i(t) = 0$, $i = 1, \ldots, n$.

If $p_i = 0$ is not stable, then it is said to be *unstable*.

If $p_i = 0$ is stable and if, in addition, $|\phi_i(t, t_0, p_{i0})|$ tends to zero as $t \to \infty$ whenever $p_i \in \mathcal{D}_i$, where $\mathcal{D}_i$ is a subset of $\mathbb{R}$ containing the origin $p_i = 0$, then $p_i = 0$ is said to be *asymptotically stable* and $\mathcal{D}_i$ is called the *domain of attraction* for $p_i = 0$. If $\mathcal{D}_i \equiv \mathbb{R}$, then $p_i = 0$ is said to be *globally asymptotically stable*.

If $p_i = 0$ is asymptotically stable and if, in addition, $|\phi_i(t, t_0, p_{i0})|$ tends to zero exponentially, then $p_i = 0$ is said to be *exponentially stable*.

The *direct method of Lyapunov* enables the stability properties of the origin $p_i = 0$ for (4) to be determined without the actual determination of the solutions $\phi_i(t, t_0, p_{i0})$. In this method, continuously differentiable scalar-valued functions, $v_i(p_i)$, called Lyapunov functions, are employed. If a positive definite function $v_i(p_i)$ can be found such that the rate of change of $v_i(p_i)$ with respect to time along the solutions of (4), denoted $Dv_i(p_i)$, is negative semi-definite, then the equilibrium p_i for (4) will be stable. If a positive definite $v_i(p_i)$ can be found such that $Dv_i(p_i)$ is negative definite, then the equilibrium $p_i = 0$ will be asymptotically stable. If, in addition, both $v_i(p_i)$ and $Dv_i(p_i)$ are quadratic forms, then $p_i = 0$ will be exponentially stable. A more precise and complete treatment of Lyapunov's direct method is given in [Miller et al. (1982)].

3.2 Stability of Neural Networks

The equilibrium $u = 0$ of the neural network (1) is exponentially stable if: 1) for the system (1) all the external inputs are zero: $U_i(t) \equiv 0$, $i = 1, \ldots, n$; 2) the interconnections satisfy the estimate: $u_i A_{ij} G_j(u_j) \leq |u_i||a_{ij}||u_j|$, for all $|u_i| < r_i$, $|u_j| < r_j$ and $i, j = 1, \ldots, n$, where a_{ij} are real constants;

3) there exists a vector $\alpha \in \mathbb{R}_+^N$ (i.e. $\alpha^T = (\alpha_1, \ldots, \alpha_N)$ and $\alpha_i > 0$, $i = 1, \ldots, n$) such that the test matrix $\mathbf{S} = [s_{ij}]$

$$s_{ij} = \begin{cases} \alpha_i(-b_i + a_{ii}) & i = j \\ \frac{1}{2}(\alpha_i a_{ij} + \alpha_j a_{ji}) & i \neq j \end{cases} \quad (6)$$

is negative definite.

The Lyapunov function of the neural network (1) $v(\mathbf{u}) = \frac{1}{2} \sum_{i=1}^n \alpha_i u_i^2$ is considered as a weighted sum of the Lyapunov functions of the free subsystems with $U_i(t) \equiv 0$). The weighting factor $\alpha > 0$ is chosen so as to emphasize the qualitative properties of the individual subsystems.

Finally, it should be noted that the parameters A_{ij} need not form a symmetric matrix for the definitions of stability given previously to hold.

To conclude, the books by [Miller et al. (1982)], [Mitchel et al. (1977)] give a more detailed survey on stability analysis.

4. NONLINEAR NETWORKS WITH PERTURBATIONS

Consider the perturbed version of the Hopfield model (1)

$$\dot{u}_i = -\tilde{b}_i u_i + \sum_{j=1}^n \tilde{A}_{ij} \tilde{G}_j(u_j) + U_i(t), \quad (7)$$

where $\tilde{b}_i = b_i + \Delta b_i$, $\tilde{A}_{ij} = A_{ij} + \Delta \tilde{A}_{ij}$ (note, that $\Delta \tilde{A}_{ij}$ does not need to be a symmetric) and $\tilde{G}_j(u_j) = G_j(u_j) + \Delta \tilde{G}_j(u_j)$ ($\Delta \tilde{G}_j(u_j) \in C^2[\mathbb{R}, (-1, 1)])$, $1 \leq i \leq n$.

The system (1) is said to be *robust*, if for every asymptotically stable equilibrium u_e of (1), and for every $\varepsilon > 0$, there is a $\delta > 0$, such that for any perturbed system (7), as long as

$$\max\{|\Delta b_i|, |\Delta A_{ij}|, |\Delta G_j(u_e)|, |\Delta G_j'(u_e)|, |U_i|\} < \delta$$

where

$$\Delta G_j'(u_j) = \frac{d(\Delta G_j)(u_j)}{du_j},$$

there is an *asymptotically stable* equilibrium $\tilde{u}_e$, of system (7), such that $|u_e - \tilde{u}_e| < \varepsilon$.

To summarise, robustness means that the system (1) is not overly sensitive to small perturbations. This is very important from a practical viewpoint as robustness ensures that small errors encountered in practical implementations of associative memories will not affect in an adverse manner the accuracy of the desired stored memories. Robustness ensures that the locations of the desired

asymptotically stable equilibria, which are used as memories in the neural networks, are not affected adversely by small perturbations.

The equilibrium u_e of the system (7) is *exponentially stable* iff [Wang et al. (1994)]: 1) u_e is an equilibrium point of both (7) and (1); 2) the matrix $-\mathbf{b} + \mathbf{AG}'$ is Hurwitz stable; 3)

$$\max\{|\Delta b_i|_\infty, |\Delta A_{ij}|_\infty, |\Delta A_{ij}|_1, |\Delta G_j'(u_e)|_\infty, \} < K_0,$$

where $\frac{1}{K_0} = 2|\mathbf{P}|_\infty(1 + |\mathbf{A}|_\infty + |\mathbf{G}'|_\infty)$, and $\mathbf{P} = \mathbf{P}^T$ is a positive definite symmetric matrix, determined by $\mathbf{PA} + \mathbf{A}^T\mathbf{P} = -\mathbf{E}$, where $\mathbf{E} \in \mathbb{R}^{n \times n}$ is the identity matrix, $\mathbf{A} = [A_{ij}]_{i,j=1}^n$, $\mathbf{b} = \text{diag}(b_1, \ldots, b_n)$, and where

$$\mathbf{G}'(u) = \left(\frac{dG_1(u_1)}{du_1}, \ldots, \frac{dG_n(u_n)}{du_n}\right)^T.$$

Note that K_0 is a positive number, that is determined by system (1) and is independent of the system perturbations. It is an admissible bound for robust stability.

The equilibrium $\tilde{u}_e$ of the system (7) is *exponentially stable* iff [Wang et al. (1994)]: 1) the matrix $-\mathbf{b} + \mathbf{AG}'$ is Hurwitz stable; 2)

$$\max\{|\Delta b_i|_\infty, |\Delta A_{ij}|_\infty, |\Delta A_{ij}|_1, |\Delta G_j'(u_e)|_\infty, \} < K,$$

where $\frac{1}{K} = 4|\mathbf{P}|_\infty(1 + |\mathbf{G}|_1 + |\mathbf{A}|_\infty)$, where again $\mathbf{P} = \mathbf{P}^T$ is a positive definite symmetric matrix, determined by $\mathbf{PA} + \mathbf{A}^T\mathbf{P} = -\mathbf{E}$, ($\mathbf{E} \in \mathbb{R}^{n \times n}$ is the identity matrix), $|\mathbf{G}|_1 = \sup_{|x-x_e|_\infty < \varepsilon} |\mathbf{G}'|_\infty$, $|\tilde{u}_\epsilon - u_\epsilon| < \min(\epsilon, 1/(4|\mathbf{G}|_2|\mathbf{A}|_\infty|\mathbf{P}|_\infty))$, $\mathbf{A} = [A_{ij}]_{i,j=1}^n$ is a $n \times n$ - matrix, and $\mathbf{b} = \text{diag}(b_1, \ldots, b_n)$.

Note that in the definitions given above, it is supposed that there exists an equilibrium of the perturbed system (7) that remains close to the corresponding equilibrium of the unperturbed system (1) for $t > 0$. For small perturbations, these assumptions hold true [Mitchel et al. (1989)], [Mitchel et al. (1977)]. [Wang et al. (1994)] give a detailed study of the stability conditions relating to perturbed neural networks.

In specific applications involving adaptive schemes for learning algorithms in neural networks, the interconnection patterns (and external inputs) are changed to yield an evolution of different sets of desired asymptotically stable equilibrium points with appropriate domains of attraction. One can derive a series of conditions, that can be used as constraints to guarantee that the desired equilibria always have the desired stability properties [Mitchel et all., 1989].

5. MODEL REDUCTION OF NONLINEAR SYSTEMS: EMPIRICAL GRAMIANS AND BALANCED TRUNCATION

Let us consider nonlinear systems of the most general form:

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{f}(t, \mathbf{x}(t)) + \mathbf{B}(t)\mathbf{u}(t) \quad (8) \\ \mathbf{y}(t) &= \mathbf{h}(t, \mathbf{x}(t)) \end{aligned}$$

where $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^n$ and $\mathbf{h} : \mathbb{R}^n \to \mathbb{R}^q$ are non-linear functions and the function $\mathbf{u}(t) \in \mathbb{R}^n$ is regarded as an input signal to the system (8) and the function $\mathbf{y}(t) \in \mathbb{R}^q$ is an output signal. In this case, $\mathbf{f}(t, \mathbf{x}(t))$ is called a dynamical term (or drift term) and $\mathbf{B}(t)\mathbf{u}(t)$ is called a source term (or diffusion term).

The unperturbed Hopfield network model (1) corresponds to (8) if the components of $\mathbf{f}(t, \mathbf{x}(t))$ are given by

$$f_i(t, \mathbf{x}(t)) = -b_i x_i + \sum_{j=1}^{n} A_{ij} G_j(x_j)$$

and the source term in (8) is given by the external input $U(t)$. In a similar manner, one can identify the different terms in the perturbed Hopfield model (7).

Suppose that the equilibrium point is reached when $\mathbf{u}(t) = 0$. Consider the vicinity of an isolated asymptotically stable equilibrium point (steady–state solution) which is supposed to be a constant solution and is chosen for simplicity at $\mathbf{x} = 0$, i.e. $\mathbf{f}(t, 0) \equiv 0$. It is also assumed that the system does not leave the region of attraction of this equilibrium point when the input is applied for the initial data used. If the system exhibits multiple steady–state solutions, then the analysis may be applied separately in the vicinity of each solution provided that extra care is taken to ensure that the system does not leave the region of attraction of the corresponding (asymptotically stable) equilibrium point.

Let also $\mathbf{x}^{ilm}(t)$ be the solution of (8) with $\mathbf{u} \equiv 0$:

$$\dot{\mathbf{x}}(t) = \mathbf{f}(t, \mathbf{x}(t)), \qquad \mathbf{x}^{ilm}(0) = c_m T_l e_i. \quad (9)$$

It is assumed that the initial condition in (9) does not take the system outside the region of attraction of the equilibrium point $\mathbf{x} = 0$. Then the 'state-space average' of the 'nonlinear' fundamental solution may be defined as:

$$\langle \Theta(t) \rangle = \frac{1}{rs} \sum_{m=1}^{s} \sum_{l=1}^{r} \sum_{i=1}^{n} \frac{1}{c_m} x^{ilm}(t) e_i^T T_l^T \quad (10)$$

where $\mathbf{M} \equiv \{c_1, c_2, \ldots, c_s\}$ is the set of s positive constants, $\mathbf{T}^n \equiv \{T_1, T_2, \ldots, T_r\}$ – is the set of r orthogonal $n \times n$ matrices and $\mathbf{E}^n \equiv \{e_1, e_2, \ldots, e_n\}$ is the set of standard unit vectors in $\mathbb{R}^n$. So for the system in (8), the *nonlinear* controllability gramian is defined as:

$$P = \int_0^{\infty} \langle \Theta(-\tau) \rangle^{-1} B(-\tau) B^T(-\tau) \langle \Theta(-\tau) \rangle^{-1T} d\tau \quad (11)$$

where $\langle \Theta(t) \rangle$ is as described in (10) and the *nonlinear* observability gramian is defined as [Condon et al. (2004)]:

$$Q = \int_0^{\infty} z^T(\tau) z(\tau) d\tau,$$

$$z(t) = \frac{1}{rs} \sum_{i,l,m} \frac{1}{c_m} y^{ilm}(t) e_i^T T_l^T \quad (12)$$

$y^{ilm}(t)$ is the output which corresponds to an initial state $x^{ilm}(0) = c_m T_l e_i$ and a zero source term.

Let T be the matrix that transforms both P and Q into diagonal form S as follows:

$$TPT^* = S, \quad (T^{-1})^* Q T^{-1} = S, \quad (TPQT^{-1} = S^2)$$

The states of the system are then ordered according to decreasing values of the diagonal entries in S. Once balanced, a Galerkin projection $\Pi = [I, 0]$, where Π is $k \times n$ projection matrix and I is $k \times k$ unit matrix, is then employed to project the transformed system onto the states corresponding to the k largest singular values (i.e. the k largest values of the diagonal matrix S where k is the desired dimension of the reduced-order model).

The final reduced nonlinear model that corresponds to (8) takes the form:

$$\dot{\mathbf{z}}(t) = \Pi T \mathbf{f}(t, T^{-1}\Pi^* \mathbf{z}(t)) + \Pi T \mathbf{B}(t)\mathbf{u}(t)$$
$$\mathbf{y}(t) = \mathbf{h}(t, T^{-1}\Pi^* \mathbf{z}(t)) \quad (13)$$

6. PERTURBED NONLINEAR NETWORKS AND MODEL REDUCTIONS

The model reduction technique of Section 5 will now be applied to the perturbed neural networks.

6.1 The case of free subsystems

In the case of vanishing $U_i(t)$, the reduced nonlinear model that corresponds to (4) is given by

$$\dot{z}_i = -\bar{b}_i z_i + \bar{A}_{ii} \bar{G}_i(\bar{p}_i), \quad (14)$$

or in vector form

$$\dot{\mathbf{z}}_i = -\bar{\mathbf{B}} \mathbf{z} + \bar{\mathbf{A}} \bar{\mathbf{G}}(\bar{\mathbf{p}}). \quad (15)$$

where $\bar{\mathbf{B}} = \vec{B}\mathbb{E}$, $\vec{B} = (\bar{b}_1, \ldots, \bar{b}_n)^T$ (here, again $\mathbb{E}$ is the $n \times n$ unit matrix) and the reduced model parameters are expressed through the initial ones as follows:

$$\bar{\mathbf{A}} = \Pi T \tilde{\mathbf{A}} T^{-1} \Pi^*, \qquad \mathbf{z} = \Pi T \mathbf{p} \quad (16)$$
$$\bar{\mathbf{B}} = \Pi T \tilde{\mathbf{B}} T^{-1} \Pi^*, \qquad \bar{\mathbf{G}} = \Pi T \tilde{\mathbf{G}}.$$

Here, Π is $k \times n$ Galerkin projection matrix and T is the transformation matrix that casts both the gramians (11) and (12) into diagonal form (cf. Section 5).

If it is possible to force the solutions $\phi_i(t, t_0, p_{i0}, \varepsilon)$ of (4) with $U_i(t = 0)$, $i = 1, \ldots, n$ to remain as close as desired to the stable equilibrium $p_i = 0$ for all $t > t_0 > 0$, by choosing p_{i0} to be sufficiently close to $p_i = 0$, then it will be possible to force the solutions $\bar{\phi}_i(t, t_0, p_{i0}) = \Pi T \tilde{\phi}_i(t, t_0, p_{i0})$ of (15) to remain as close as desired the same stable equilibrium point for all $t > t_0 > 0$. So the equilibrium $p_i = 0$ of (4) with zero external inputs, will be *also* a stable equilibrium point for (14). The proof can be done in a similar manner to those in [Condon et al. (2008)].

Analogously, if $p_i = 0$ is an unstable equilibrium point for eqn. (4) with zero external inputs, then it it will be unstable equilibrium point for eqn. (14) as well.

Finally, if $p_i = 0$ is exponentially stable for eqn. (4) with zero external inputs, then it it will be exponentially stable equilibrium point for eqn. (14) as well.

The proofs are in a similar manner as in [Condon et al. (2008)].

6.2 The case of neural networks

Now let us consider the case of *interconnected systems (neural networks)*.

Applying again model order reduction (as it is outlined in Section 5), we get a reduced nonlinear model, that corresponds to (7) in the form

$$\dot{z}_i = -\bar{b}_i z_i + \sum_{j=1}^n \bar{A}_{ij} \bar{G}_j(\bar{p}_j) + \bar{U}_i(t). \qquad (17)$$

or in vector form

$$\dot{\mathbf{z}}_i = -\bar{\mathbf{B}}\mathbf{z} + \bar{\mathbf{A}}\bar{\mathbf{G}}(\bar{\mathbf{p}}) + \bar{\mathbf{U}}(t), \qquad (18)$$

and the reduced model parameters are expressed through the initial ones (using the Galerkin projection matrix Π and the transformation matrix T) as follows:

$$\bar{\mathbf{A}} = \Pi T \tilde{\mathbf{A}} T^{-1} \Pi^*, \qquad \bar{\mathbf{U}} = \Pi T \tilde{\mathbf{U}}, \qquad (19)$$
$$\bar{\mathbf{B}} = \Pi T \tilde{\mathbf{B}} T^{-1} \Pi^*, \qquad \bar{\mathbf{G}} = \Pi T \tilde{\mathbf{G}}.$$

If the equilibrium point $\mathbf{p} = 0$ for eqn. (7) is asymptotically stable in the case of zero external input $U_i(t) = 0$, then it is asymptotically stable equilibrium point for eqn. (17) as well.

Indeed, if there exist constants $\sigma_{i1} > 0$ and $\sigma_{i2} > 0$ for some $\tilde{r}_i > 0$, such that $\sigma_{i1} p_i^2 < p_i G_i(p_i) < \sigma_{i2} p_i^2$, $|p_i| < r_i$, then one can find constants $\tilde{\sigma}_{i1} > 0$ and $\tilde{\sigma}_{i2} > 0$, such that:

$$\tilde{\sigma}_{i1} z_i^2 < z_i \tilde{G}_i(z_i) < \tilde{\sigma}_{i2} p_i^2, \qquad |z_i| < \tilde{r}_i. \qquad (20)$$

If the interconnections for eqn. (4) satisfy the estimate: $u_i A_{ij} G_j(x_j) \le |u_i| a_{ij} |u_j|$, for all $|u_i| < r_i$, $|u_j| < r_j$ and $i, j = 1, \ldots, n$, where a_{ij} are real constants, then the interconnections for eqn. (17) will satisfy the estimate: $z_i \tilde{A}_{ij} \tilde{G}_j(z_j) \le |z_i| \tilde{a}_{ij} |z_j|$, for all $|z_i| < \tilde{r}_i$, $|z_j| < \tilde{r}_j$ and $i, j = 1, \ldots, n$, where $\tilde{a}_{ij}$ are suitably chosen real constants. Secondly, if the test matrix $\mathbf{S} = [s_{ij}]$ for eqn. (4) is negative definite, then the test matrix $\tilde{\mathbf{S}} = [\tilde{s}_{ij}]$ for eqn. (17) will be negatively definite as well.

Detailed proofs for the stability results of neural networks can be found in [Mitchel et al. (1989)] and for the effect of model reduction on the stability of the Hopfield type network models – in [Condon et al. (2008)].

Summarizing, if $\mathbf{u} = 0$ is an exponentially stable equilibrium of the perturbed neural network (7), then it is an exponentially stable equilibrium of the reduced neural network (18), for the case when all the external inputs are zero: $U_i(t) \equiv 0$, $i = 1, \ldots, n$.
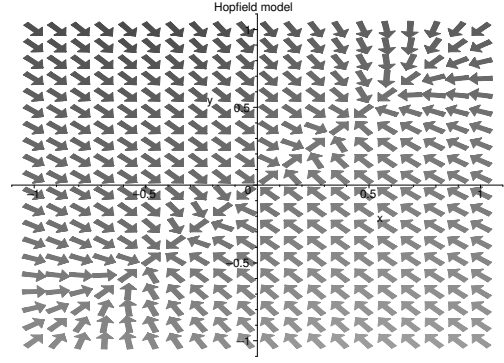

Figure 2: Phase portrait for the unperturbed Hopfield neural network model

7. EXAMPLE

For illustration of some of the stability results given above, we will take a specific test case, taken from [Hopfield (1984); Mitchel et al. (1989)]. For this case

$$A = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad g_i(x_i) = \frac{2}{\pi} \arctan\left(\frac{\lambda\pi}{2} x_i\right),$$

where $\lambda = 1.4$. Choosing the specific values of b_1 and b_2 to be $b_1 = b_2 = 1.1$, the model equation (1) takes the form:

$$\dot{u}_1 = -1.1 u_1 + g_2(u_2) + U_1(t)$$
$$\dot{u}_2 = -1.1 u_2 + g_1(u_1) + U_2(t). \qquad (21)$$

In the case of zero input $U_1(t) = U_2(t) = 0$ we have three equilibrium points for eqn. (21): $(-0.454, -0.454)$, $(0, 0)$ and $(0.454, 0.454)$ (as it is plotted in Fig.2).

By using the direct Lyapunov methods, one can prove that the equilibrium point $(0, 0)$ is not globally asymptotically stable. The precise proof can be found in [Mitchel et al. (1977)] and [Mitchel et al. (1989)].

The translated nonlinearities $G_i(w_i)$, $i = 1, 2$ for the equilibrium points $(\pm 0.454, \pm 0.454)$ are given by:

$$G_i(w_i) = \frac{2}{\pi} \arctan\left(\frac{\lambda\pi}{2}(w_i \pm 0.454)\right)$$
$$+ \frac{2}{\pi} \arctan\left(\frac{\lambda\pi}{2}(\mp 0.454)\right) \qquad (22)$$

In this case, we have $0 < G_i(w_i)/w_i < \sigma_{i2}$, $\sigma_{i2} < b = 1.1$, when $|w_i| < 0.454$ and the test matrix S takes the form

$$S = \begin{pmatrix} \frac{1.1}{\sigma_{i2}} & -1 \\ -1 & \frac{1.1}{\sigma_{i2}} \end{pmatrix}.$$

Since all successive principal minors of S are positive, the equilibrium points $(\pm 0.454, \pm 0.454)$ are asymptotically stable.

Consider the perturbed system, that corresponds to (21) in the form:

$$\dot{u}_1 = -1.1 u_1 + (1 + \epsilon) g_2(u_2) + U_1(t)$$
$$\dot{u}_2 = -1.1 u_2 + (1 + \epsilon) g_1(u_1) + U_2(t), \qquad (23)$$

where ϵ is a perturbation parameter.

Eqn. (23) can be linearised by using a Taylor series expansion of arctanx about the origin: $\arctan x = x - \frac{x^3}{3} + o(x^4)$. For sufficiently small ϵ, one of the eigenvalues of the linearised system is positive while the other is negative and hence the origin is a saddle point for both the perturbed and unperturbed systems. For the other equilibrium points when small perturbations are applied, the equilibrium points remain asymptotically stable.

In general, however, as shown in [Wang et al. (1994)], the perturbed neural network system may not have the same asymptotically stable points as the unperturbed system.

For large scale systems, when empirical balanced truncation is applied, the same conclusions can be made.

8. CONCLUSIONS

The paper has examined the stability properties of large-scale dynamical systems. In particular, it has focussed on nonlinear neural networks with perturbations. For this type of system, the properties of both the individual neurons and the interconnecting structure are employed to determine results for the system. The paper proceeds to ascertain the effect of model reduction technique on the resultant robustness of the equilibrium points of the system.

Further publications will explore the effect of model order reduction on the stability properties of perturbed nonlinear neural networks with feedback.

REFERENCES

A. C. Antoulas, *Approximation of Large-Scale Dynamical Systems* (Advances in Design and Control Series), SIAM, Philadelphia (2003).

Borisyuk A., Friedman A., Ermentrout B. and TermanD., *Tutorials in Mathematical Biosciences I: Mathematical Neuroscience*, Lecture Notes in Mathematics **1860**, Springer-Verlag, Berlin, 2005.

Condon M. and Ivanov, R. Nonlinear systems-algebraic gramians and model reduction. *J. Nonl. Sci.* 2004, Vol.14, 2004, pp 405-414.

M. Condon, G. G. Grahovski, Stability and Model Order Reductions of Nonlinear Systems with Applications to Neural Networks I: Hopfiel Type Models, Preprint 2008.

John S. Denker, Ed., *Neural Nemorks for Computing*, American Institute of Physics Conf. **151**, Snowbird, UT, 1986.

Leah Edelstein-Keshet, *Mathematical Models in Biology*, Classics in Applied Mathematics **46**, SIAM, Philadelphia, 2005.

Gray WS, Scherpen JMA. Hankel Operators and Gramians for Nonlinear Systems. *Proceedings of the 37th IEEE Conference on Decision and Control (CDC'98)*, Tampa, Fl, USA 1998, pp. 1416-1421.

M. Griffin, G. Tahara, K. Knorpp, R. Pinkham, and B. Riley, An 11- million transistor neural network execution engine, in *ISSCC91, Dig. Tech. Papers, Feb. 1991, pp. 180–181*.

Lj. T. Grujic, A. A. Martynyuk, and M. Ribbens-Pavella, *Stability of Large-Scale Systems Under Structural and Singular Perturbuiions*, Nauka Dumka, Kiev, USSR, 1984.

Hahn J, Edgar TF. An Improved Method for Nonlinear Model Reduction Using Balancing of Empirical Gramians. *Computers and Chemical Engineering* 2002; Vol. 26(10), pp. 1379-1397.

G. E. Hinton and J. A. Anderson, Eds., *Parallel Models of Associative Memory*, Erlbaum, 1981.

J. J. Hopfield *Neural Networks and Physical Systems with Emergent Collective Computational Abilities*, Proc. NatL Acad. Sci. USA **79** (1982), 2554–2558.

J. J. Hopfield, *Neurons with graded response have collective computationa properties like those of two-state neurons*, Proc. Nat. Acad. Sci. USA **81** (1984), 3088-3092.

J. J. Hopfield and D. W. Tank, Computing with neural circuits: A model, *Science,* vol. 233, pp. 625-633, 1986.

A. Iwata, Y. Yoshida, S. Matsuda, Y. Sato, and N. Suzumura, An artificial neural network accelerator using general purpose 24 bits floating point digital signal processors, in *Int. Joint Conf. Neural Networks, 1990, vol. II, pp. 43–46*.

T. Kohonen, *Self-Organization and Associative Memory*, New York: Springer-Verlag, 1984.

Lall S, Marsden JE, Glavaski S. A subspace approach to balanced truncation for model reduction of nonlinear control systems. *International Journal of Robust and Nonlinear Control* 2002; Vol. 12, pp. 519-535.

J.-H. Li, A. N. Michel, and W. Porod, Qualitative analysis and synthesis of a class of neural networks, *ZEEE Trans. Circuits Syst., 533-541, 1986.*

R. K. Miller and A. N. Michel, *Ordinary Differential Equations*, New York: Academic, 1982.

A. N. Michel and R. K. Miller, *Qualitative Analysis of Large Scale Dynamical Systems*, New York: Academic, 1977.

A. N. Michel, J. A. Farrell, W. Porod, Qualitative Analysis of Neural Networks, *IEEE Trans. Circ. Sys, Vol. 36, No. 2, (1989), 229–243*.

Moore B. Principal Component analysis in linear systems: Controllability, Observability and model reduction. *IEEE Trans. on Automatic Control* 1981; AC-26(1).

Phillips JR. Projection-based approaches for model reduction of weakly nonlinear, time-varying systems, *IEEE Transactions on computer-aided design of integrated circuits and systems* 2003, Vol. 22, No. 2.

D. Pomerleau, G. Gusciora, D. Touretzky, and H. Kung, Neural network simulation at warp speed: How we got 17 million connections per second, in *IEEE Int. Conf. Neural Networks, 1988, vol. II, pp. 143–150.*

Rewienski M and White J. A trajectory piecewise-linear approach to model order reduction and fast simulation of nonlinear circuits and micromachined devices, *IEEE Transactions on Computer-aided Design of Integrated Circuits and Systems* 2003, Vol. 22, No. 2, pp. 155-170.

Scherpen JMA. Balancing of nonlinear systems. *Systems and Control Letters* 1993; Vol. 21, pp.143-153.

K. Wang and A. N. Michel, Robustness and Perturbation Analysis of a Class of Nonlinear Systems with Applications to Neural Networks, *IEEE Trans. Circ. Sys, Vol. 41, No. 1, (1994), 24–32.*

J. Wawrzynek, K. Asanovic, and N. Morgan, The design of a neuromicroprocessor, *IEEE Trans. Neural Networks, vol. 4, pp. 394–399, May 1993*.

AUTHOR BIOGRAPHIES

MARISSA CONDON is a lecturer at DCU. Her research interests include the development of simulation techniques for nonlinear and high-frequency circuits and mathematical modelling of biological systems. Her webpage is `http://www.eeng.dcu.ie/∼condonm.html`.

GEORGI GRAHOVSKI is a post-doctoral researcher at DCU. He has research interests in integrable systems, solitons, Lie groups and Lie algebras, differential geometry and biophysical models in addition to nonlinear circuit simulation techniques and nonlinear neural networks.

An Application-Oriented Synthetic Network Traffic Generator

Carsten Albrecht, Christoph Osterloh, Thilo Pionteck, Roman Koch, and Erik Maehle
Institute of Computer Engineering
University of Lübeck
Ratzeburger Allee 160
23538 Lübeck
Germany
{albrecht,osterloh,pionteck,koch,maehle}@iti.uni-luebeck.de

Abstract— **Design space exploration and detailed analysis in the field of hardware design applies simulation in many variants. A frequently used method is stochastic simulation where systems are stimulated by randomised input. Synthetic traffic traces mainly form the load for stochastic simulation of network computing devices. The generator presented here utilises two well-known models to meet the features of a majority of applications and traffic sources. Based on application-specific parameter sets, the traffic models stochastically generate packet flows which are merged to an aggregated stream. Nevertheless, all packets can always be identified and are not resolved to a data mass representing the load of a link.**

Keywords— **Traffic Generator, Packet Flow Models, Synthetic Workload Generation**

I. Introduction

Growing bandwidth and complexity of network applications and protocols, as well as issues of network security and integrity cause more and more complex packet processing tasks. They have to be performed particularly at the edges of sub-networks. The resulting need for increasingly powerful and flexible processing devices is, however, hard to satisfy, because the general increase in processing power is exhausted by the increase in utilised bandwidth. In order to exploit the evolving bandwidth potential, it is therefore inevitable to investigate architectures tailored to specific application areas. Simulation of hardware designs to examine their potential concerning performance and power is a suitable and wide-spread technique for design-space exploration and evaluation of the applied algorithms.

Simulation results highly depend on the quality of the stimulation. In the field of network computing, packet traces forming the data to be processed within a device are needed as stimulations. With regard to the scope of application the examined device has to be tested and investigated by realistic but also challenging stimuli to show behaviour and performance characteristics in heavy-load scenarios.

Since a decade, hardware devices such as FPGAs (Field Programmable Gate Arrays) provide more flexibility. While they were just reprogrammable case-by-case they are now even partially reprogrammable while they execute a certain task. These run-time reconfigurable systems are able to adapt their system behaviour and to outperfom traditional systems [Compton and Hauck, 2002]. Applied to network computing as a discipline of hardware design new features have to be tested and evaluated. That is why usual benchmark suites and packet generators are not sufficient anymore to stress the devices for any purpose. Beside usual requirements on data rates, and protocol stacks, the adaptive features of run-time reconfigurable network devices need certain conditions that exercise and test their capabilities. In particular, the traffic traces demanded here have to be realistic and have to include changes over long and short terms. Additionally, it is necessary to distinguish traffic requirements on a rather fine-granular per-flow basis to adapt the system to it. A sample system that requires these features of a traffic generator tool for analysis is described in [Albrecht et al., 2006], [Pionteck et al., 2006].

The traffic generator presented in the following uses well-known traffic models with application-specific parameter sets for generating flows. The term *flow* is defined here as 5-tuple of used protocol within the Internet Protocol (IP) packet, source address, destination address, source port, and destination port. A fixed time-out value T_0 describes the longest possible period between two packets of the same flow [Hohn et al., 2005]. The ports are given by the application originating the traffic.

The paper is organised as follows: First, the basic packet models are introduced in Section II and then are mapped to applications in Section III. Based on that, the concept and user interface of the synthetic traffic generator are described in Section IV. Section V shows generated traffic and sketches its application in the system-design phase. Before Section VII finalises this work and gives final remarks some existing and frequently used packet generators are described and compared in Section VI.

II. Packet Flow Models

Network traffic is still frequently modelled using the Poisson model although many examinations have shown that the packet inter-arrival time is in general not exponentially distributed [Paxson and Floyd, 1995]. Consequently, there are numerous models for different kinds of network traffic. They have at least two
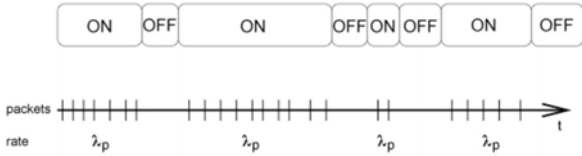
Fig. 1. Flow generated by ON/OFF model.

Fig. 2. Flow generation using b-model.

common features: self-similarity and long-range dependency. Self-similarity in the area of network traffic is the quality of keeping similar structures independent of scale and section. The requirement for self-similarity is shown in [Leland et al., 1994]. Long-range dependency is the property that there is a high dependency within long time intervals [Babic et al., 1998] and, thus, have high auto-correlation values over long periods of time. Further on, as a third property of the generated traffic which is closely related to long-range dependency, traffic should be heavy tailed. This leads to the Pareto principle where a huge number of small values add less to the average than a few large values. This effect is also known as $80 - 20$ law. The Pareto distribution is the most famous heavy-tailed distribution. Beside others the following two models provide the necessary properties and are able to cover a large number of applications generating network traffic.

A. ON/OFF Model

The ON/OFF model, also called packet-train model [Jain and Routhier, 1986], generates self-similar and long-range dependent traffic. The model bases on a finite-state machine with two states, ON and OFF. Packets are generated in bursts during the ON phases. ON phases last a time that follows a general distribution whereas the durations of OFF phases are exponentially distributed.

Figure 1 shows bursts generated by an ON/OFF model. The first row shows an alternating sequence of states, the second one the time of packet generation. The packets are basically transferred with the infinite transfer rate λ_p [Siriwong et al., 2007]. The parameters concerning number and size of packets as well as the distribution of ON and OFF phases depend on the application generating the traffic.

An extension to this single stream model is the N-burst model [Lipsky et al., 2001]. Here, multiple sources produce packets which are loss-less buffered in a router. The router is modelled as a service station with a Markov-modulated Poisson process.

B. b-Model

Another well suitable model for generating traffic with the properties described above is the b-model [Wang et. al., 2002]. In contrast to the ON/OFF model, the b-model can be applied to generate aggregated traffic streams such as background traffic for network studies, e.g. [Thid et al., 2006]. Furthermore, single sources such as file accesses can also be modelled.
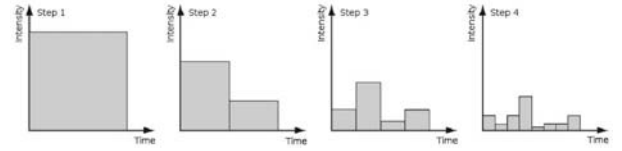
The algorithm is a divide-and-conquer mechanism

that splits the amount D of data into two parts. b acts as a weight so that one part gets $b \times D$ and the other $(1 - b) \times D$. The splitting is repeated until the required granularity is achieved. Figure 2 depicts the process of generating a b-modelled stream.

III. APPLICATIONS

Network traffic is generated by various applications. Figure 3 shows the distribution of the applications which are responsible for most of the traffic. The distribution of applications is given in [Luo and Marin, 2005]. HTTP, FTP, SSH, POP3 and SMTP form more than half of the TCP (Transmission Control Protocol) traffic. Combining this statistic with daily network statistics of e.g. www.caida.org, the set of applications and protocols has to be extended by IPSEC which is here included in the 'others' section. Nowadays, IPSEC achieves a top rank in the statistics. Many institutions and companies provide secure access to their infrastructure by applying so-called IPSEC tunnels. These tunnels contain again a lot of the previously named applications.

For modelling a significant part of usual network traffic all these applications have to be mapped to models utilisable for generating synthetic traffic streams. Although the degree and frequency of user interaction and so their parameter sets and generated traffic profile differ almost all those applications can basically be mapped to a single model: the ON/OFF model. For generating IPSEC traffic the b-model is chosen because it is also designed for aggregated traffic. Of course it would be possible to mix tunnel traffic from a set of sources, too, but in this case the b-model reduces the computing effort.
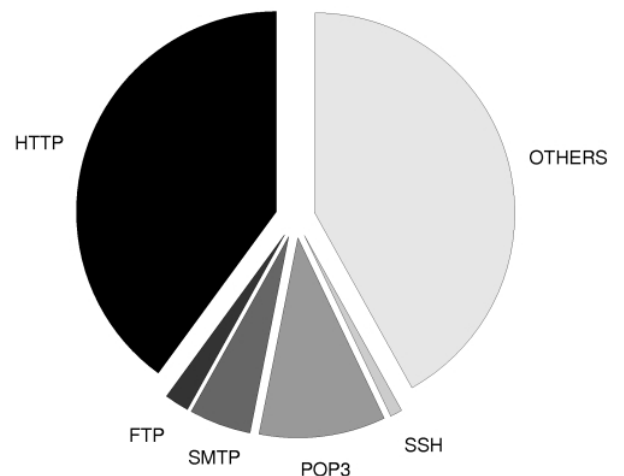

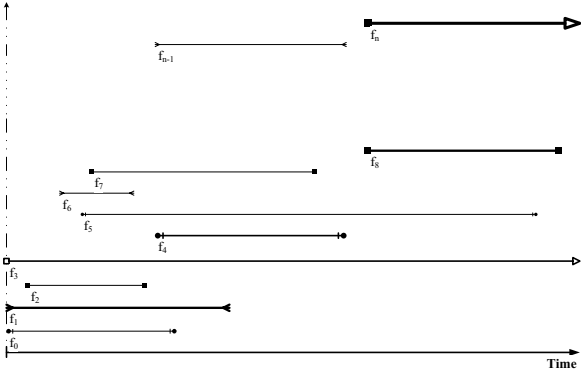Fig. 3. Application-level traffic-source distribution of the TCP protocol.

Fig. 4. Typed flow live-cycle diagram.

The parameter sets for the ON/OFF model to generate application-specific traffic can be taken from [Luo and Marin, 2005].

IV. SYNTHETIC TRAFFIC GENERATOR

Based on the models and applications of Sections II and III a synthetic traffic generator is built. Its aim is to produce an aggregated traffic stream composed of multiple sources with the opportunity to identify any packet origin. So, the result is more than only a bunch of data that has to be put on a link in a certain time period.

A. Concept

Figure 4 shows the principle of the generator. For the creation of a traffic stream multiple flows f_i can be defined. Each flow f_i has to be parametrised by a tuple $(A_{f_i}, B_{f_i}^{max}, D_{f_i}, t_{f_i})$ the elements of which are listed in Table I.

TABLE I: Flow parameters and their description.

A_{f_i} - application or type of flow
$B_{f_i}^{max}$ - link limit or maximum bandwidth
D_{f_i} - amount of data
t_{f_i} - starting point, i.e. offset to t_0

The application sets up the model. The link limit or maximum bandwidth of the flow represents the history of the flow. If it has used links with small capacities such as DSL or WLAN its bandwidth is generally reduced. Further on, the total amount of data transferred within this flow is defined. D combined with B_{max} delivers the lower boundary of flow live time. Finally, each flow has its own starting point t_{f_i} as an offset to the origin of time t_0.

The path from application level to packet level in the flow generation requires a packet size distribution. Some examinations of applications provide histograms for packet size distributions. For general TCP/IP traffic the major peaks of packet sizes are frequently at 40 Bytes, the smallest possible value, at about 576 Bytes and at 1500 Bytes. 1500 Bytes is a usual value of the maximum transfer unit (MTU) of an TCP/IP scenario [Sprint, 2008], [CAIDA, 2008]. Gaps can be closed by linearly interpolated data. In case
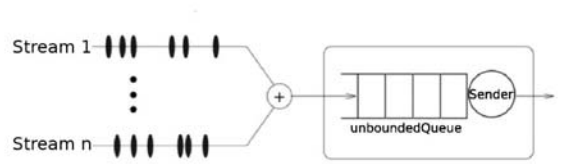

Fig. 5. Aggregation scheme for multiple flow sources.

that an application lacks a packet size distribution a uniform distribution for packet sizes is assumed.

Another important issue is the combination of multiple flows to form an aggregated stream. [Hohn et al., 2005] uses an infinite queue for buffering and merging multiple streams. The procedure for merging used here is simply a merge sort. All packets get a global timestamp with respect to the stream offset t_0. It defines when they have to be scheduled to the buffer. So, the infinite buffer finally contains all packets in ascending order of their global timestamps. Figure 5 shows the assembly process. n streams are put into the buffer on the entrance site. Inside, the resulting aggregated stream is produced by emptying the queue over a defined link with a given bandwidth. Packet timestamps and global clock are synchronised so that packets do not leave the queue before their time is reached. This may lead to intermissions but also to congestion, i.e. packets leave the queue later than their timestamps announce. This procedure is realistic and simulates the processes that are performed on any system injecting data into a system. Large data bursts have always to be buffered and sent step-by-step over the target link. Congestion in real world scenarios could lead to packet loss. Since the packet flows are generated based on parameters derived from traffic monitored within a real network, congestion and packet loss is basically taken into account. Therefore, the ideal and loss-less assumptions have to be chosen. Moreover, this technique represents the processing procedure of routers and switches where packets are labelled with a unique and ascending identifier on the ingress side which comprises multiple sources or interfaces, respectively. On the egress side the packets are transmitted in the order of their identifiers so that the packet order is kept unchanged. Quality of service (QoS) features may break this order by classifying the traffic in at least two classes with different policies and bandwidth guarantees. Here, only the order of packets within the same class is kept. Packets of different classes can be rearranged to achieve the QoS requirements.

B. Software Tool

Based on this concept a tool is realised to generate aggregated or single-source traffic streams. The back-end is formed by the implementation of the models presented in Section II. For the implementation of these stochastic processes a pseudo-random number generator is used for making decisions stochastically. The implementation applies the Newran03 library [Davies, 2006] which is used in other established
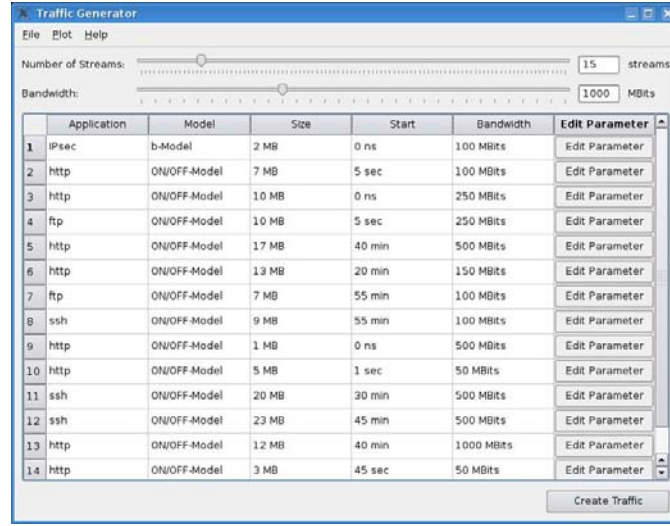
(a)Main window provides overview of generated flows.
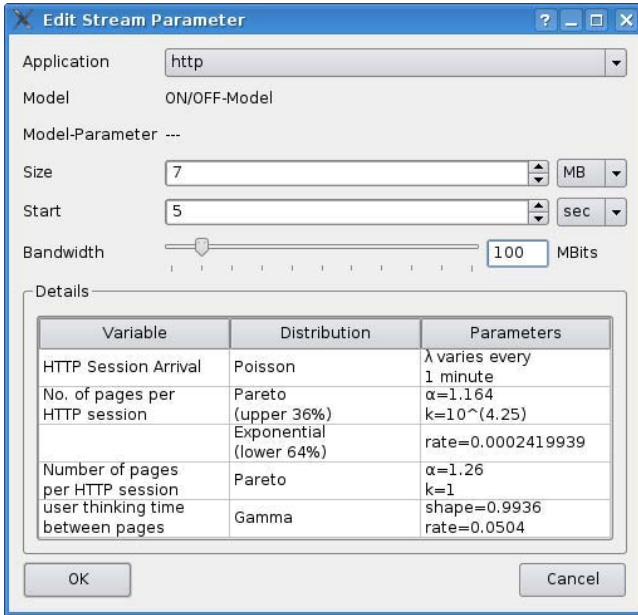
(b)Sub-menu for flow parametrisation including help on flow parameters and standard values.
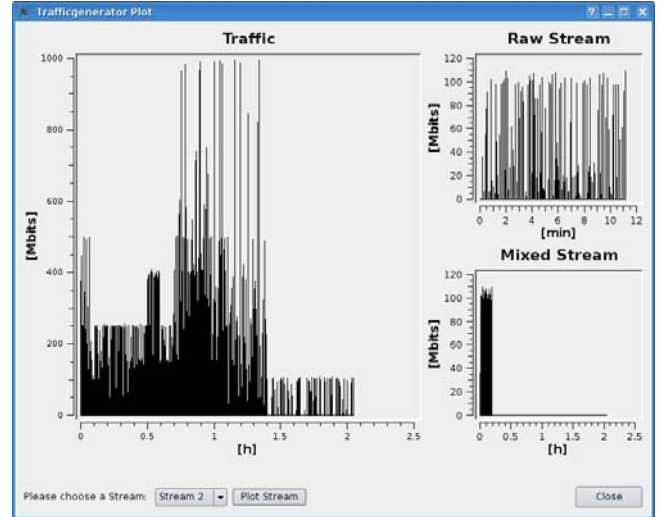
(c)Plotting aggregated and individual flows before and after merge process.

Fig. 6. Views of Traffic Generator.

simulation tools, too.

The front end of the traffic generator is a graphical user interface to ease configuration and parametrisation of the generator system and its sub-systems. The graphical user interface utilises the Qt library.

Figure 6 shows different user views on the trace configuration. In the root or main window shown in Figure 6(a) the tuple $(A_{f_i}, B_{f_i}^{max}, D_{f_i}, t_{f_i})$, see Table I, is defined. The first argument A_{f_i} includes a completely parametrised model so that it is not necessary but feasible to adapt the model parameters for own purposes. Model parameters are presented in a sub-menu with an own window shown in Figure 6(b). For user guidance the different parameters are described and the default values are given. The complete traffic configuration can be saved in a configuration file for later re-use. Because of the stochastic processes, different traces can be generated with the same settings. This is of course an important feature for the evaluation of stochastically driven simulations.

After generating a trace based on the given configuration the data distribution is visualised as shown in Figure 6(c). The large diagram on the left side of Figure 6(c) shows the cumulated data distribution of the aggregated traffic trace. For further analysis single streams can be depicted with more details on the right side. A stream is selected by the combobox [Trolltech AS, 2007] at the bottom. The diagram in the upper right corner shows the originally generated data distribution of the selected stream whereas in the lower right corner the embedded and adapted distribution is shown.

The output is given as a file in a currently proprietary format providing timestamp, application identifier, flow
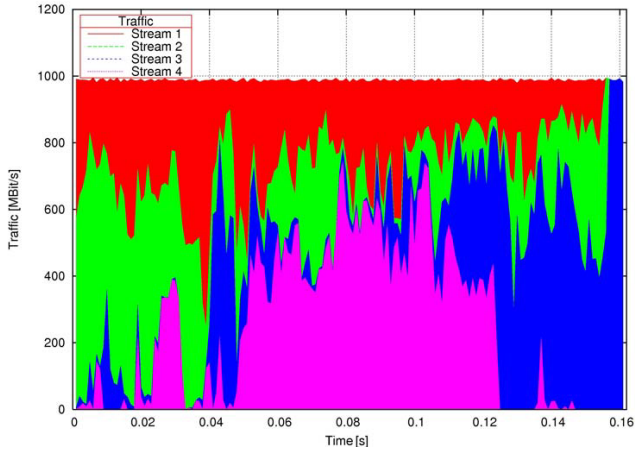
(a)Aggregation of four flows consuming complete link capacity each.
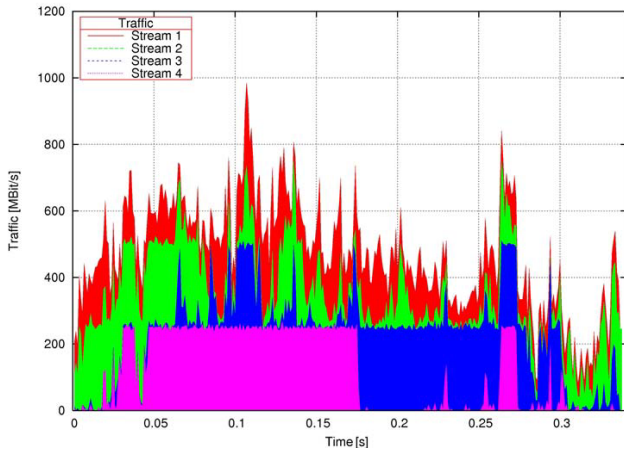
(b)Aggregation of four flows with different bandwidth limits.

Fig. 7. Synthetic b-modeled traffic samples with and without bandwidth limits for certain flows.

identifier to re-assign any packet to its originating flow description, see Figure 6, and packet size. For an improved integration into other environments the output capabilities are extended to the `pcap` format and an XML scheme.

The configuration shown in Figure 6 is utilised in the following to create the aggregated stream of Figure 8.

V. Application Example

A sample system that requires specific traffic to examine all functionalities of the device is a run-time reconfigurable coprocessor for network processors utilised for, e.g., edge routers. It provides the opportunity to off-load time-consuming payload processing so that the network processor gains headroom to serve its tasks more quickly. The coprocessor [Albrecht et al., 2006] adapts its system configuration to the flow requirements. An examination of self-adaptation, for example, is given in [Albrecht et al., 2008].

Here, the coprocessor supports tunnel modes for IPSEC partially combined with IPCOMP by providing accelerators for the cryptographic algorithms AES, DES, and 3DES as well as the compression algorithm Deflate. Tunnels are aggregated flows just as the com-

plete flow so that all streams of Figure 7(a) are generated by the b-model. Assuming that each stream is assigned to at least one of these algorithms, pipelining more than one is possible and required. It is easy to see that the requirements of the aggregated flow change over time. Just a short traffic trace is shown, but the significant difference is obvious.

Another feature of the generator is shown in Figure 7(b). Basically, these streams are the same as in Figure 7(a) but the bandwidth limit is reduced to a quarter of the maximum which is 1000 Mbit/s in this case. Peaks need now longer to be transferred so that the overall duration grows.

As mentioned before, traffic profiles change from daytime to daytime in composition and intensity. In Figure 8 a synthetic trace composed by 15 streams with different sources is shown. It is similar to daily profile with its increasing shape in the first half and a decreasing one in the second. All plots of Figure 8 show the same traffic using different time resolutions. This sequence of figures shows very well two important features of real Internet traffic which have to be met by a generator: self-similarity and burstiness. Starting with a time unit of $1s$ in Figure 8(a), time resolution is scaled up by a factor of 100 in Figure 8(b), and again by a factor of 10 in Figure 8(c). The aggregated traffic stream closely resembles itself in all three scales. Although the stream is depicted with a total scaling factor of 1000 averaging effects on the bandwidth in larger intervals are not visible. The shape including bursts and intermissions is kept over all time scales, i.e. its self-similar and bursty nature becomes obvious.

VI. Related Work

Synthetic traffic or workload generators are used in almost all fields of system and performance evaluation. So, the traffic generator presented here is of course not the first one. Aiml, granularity and, thus, applicability of the existing approaches, however, strongly differ.

[Vishwanath and Vahdat, 2006] presents the traffic generator Swing. It relies on a multi-layered model that represents user input or user interaction with the application, sessions of a single application, connections within a session, for example connections to different target hosts, and, finally, the network characteristics. Swing first monitors a packet link and evaluates the monitored data by applying this multi-layered model. Then, it starts to produce synthetic traffic based on the extracted information. The results meet well the real traffic characteristics. Nevertheless, the generator depends on an existing real environment or at least a test bed where data can be monitored.

In contrast to the previous generator, Tmix [Weigle et. al., 2006] is targeted to the network simulator *ns-2*. Again the parameters for generating traffic are extracted from a monitored traffic trace that is recorded on a representative link. The model reduces application behaviour to connection behaviour, i.e. sub-parts of an application as described before. These connections are modelled by so-called epochs which is a triplet that de-
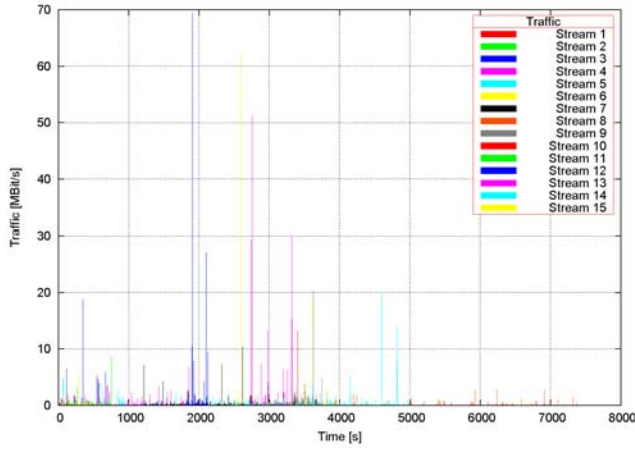
(a) Aggregated traffic with a time resolution of 1s.
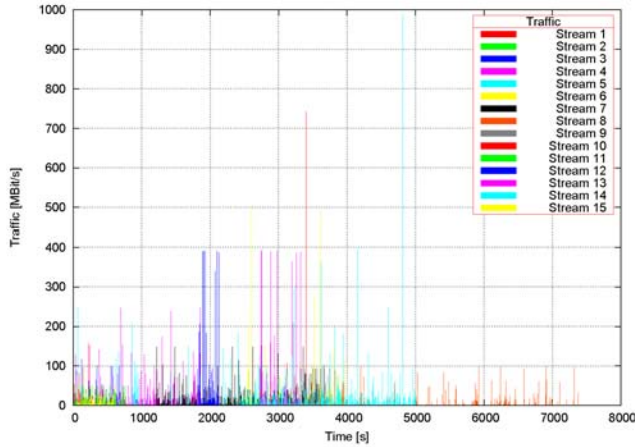
(b) Aggregated traffic with a time resolution of 10μs.
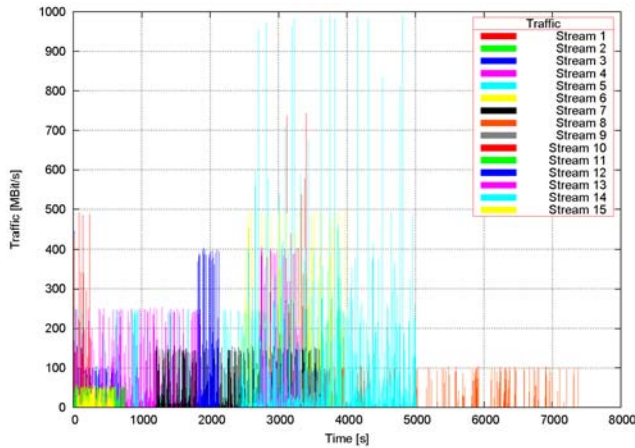
(c) Aggregated traffic with a time resolution of 1μs.

Fig. 8. Traffic plot with different time resolutions.

scribes the amount of data in both directions and the pause until the next epoch starts. This model resembles the ON/OFF model chosen here. Applications that do not match have not a representative here.

Further on, there exist approaches that only generate traffic of a single-source application such as HTTP [Barford and Crovella, 1998] or FTP [Ishac, 2001]. All these approaches are able to reproduce traffic as a single stream with the characteristics of their input parameters. Unfortunately these approaches are limited in the sense of generating traffic with changes in the profile, e.g. daily data distributions.

VII. CONCLUSION

Simulation is one of the leading techniques to evaluate new hardware designs. Examinations highly depend on the quality of stimuli driving the simulation model. Therefore, traffic generators providing traffic streams as realistic as possible are needed for synthetic workload generation in the area of network computing. The generator presented here is based only on two models which are well examined. The tool allows the generation of packet traces as coarse-grained volumes of data as well as a combination of fine-grained flows with varying sources and parameters. The transition between these two characteristics is smooth. This variant of traffic generation is important to assemble link load providing characteristics to analyse the features of the device-under-test. In particular, the recently arising field of run-time reconfigurable devices demands workload patterns that stress their adaptivity. So, beside the available functionalities it is important to build traces with self-defined characteristics and combinations of sources. The traffic generator presented here is a suitable solution to close this gap.

VIII. ACKNOWLEDGEMENT

REFERENCES

[Albrecht et al., 2006] Albrecht C., Foag J., Koch R., and Maehle E. 2006. *DynaCORE - A Dynamically Reconfigurable Coprocessor Architecture for Network Processors.* In Proceedings of the 14th Euromicro Conference on Parallel, Distributed and Network-based Processing (PDP). IEEE Computer Society. Montbeliard-Sochaux, France.

[Albrecht et al., 2008] Albrecht C., Roß P., Koch R., Pionteck T., and Maehle E. 2008. *Performance Analysis of Bus-Based Interconnects for a Run-Time Reconfigurable Co-Processor Platform.* In Proceedings of the 16th Euromicro Conference on Parallel, Distributed and Network-based Processing (PDP). IEEE Computer Society. Toulouse, France.

[Babic et al., 1998] Babic G., Vandalore B., and Jain, R. 1998. *Analysis and Modeling of Traffic in Modern Data Communication Networks.* In The Computing Research Repository, cs.NI/9809056, 1998.

[Barford and Crovella, 1998] Barford P., and Crovella M. 1998. *Generating representative Web workloads for network and server performance evaluation.* In ACM SIGMETRICS Performance Evaluation Review, Vol. 26, Issue 1, 151-160.

[CAIDA, 2008] Cooperative Association for Internet Data Analysis (CAIDA). 2008. www.caida.org.

[Compton and Hauck, 2002] Compton K., Hauck S. 2002. *Reconfigurable Computing: A Survey of Systems and Software.*

In ACM Computing Surveys, Vol. 34, No. 2, ACM Press, New York, NY, USA.

[Davies, 2006] Davies R. 2006. *Newran03 - a random number generator library* http://www.robertnz.net

[Hohn et al., 2005] Hohn N., Veitch D., and Ye T. 2005. *Splitting and Merging of Packet Traffic: Measurement and Modelling.* Performance Evaluation, Vol. 62, Issues 1-4, 164-177, 2005.

[Ishac, 2001] Ishac J. *FTP Traffic Generator.* Technical Report, Case Western Reserve University.

[Jain and Routhier, 1986] Jain R., and Routhier, S. A. 1986. *Packet Trains - Measurement and a New Model for Computer Network Traffic.* In IEEE Journal on Selected Areas in Communications, Vol. 4, No. 6, 986-995, 1986.

[Leland et al., 1994] Leland W. E., Taqqu M. S., Willinger W., and Wilson D. V. 1994. *On the self-similar nature of Ethernet traffic (extended version).* In IEEE/ACM Transactions on Networking, Vol. 2, No. 1, 1-15.

[Lipsky et al., 2001] Lipsky L., Jobmann M., Greiner M., and Schwefel, H.-P. *Comparison of the Analytic N-Burst Model with Other Approximations to Telecommunications Traffic.* In the Proceedings of the IEEE Symposium of Network Computing and Applications, 122-132, Cambridge, MA, USA.

[Luo and Marin, 2005] Luo S., and Marin G. A. 2005. *Realistic Internet Traffic Simulation Through Mixture Modeling and a Case Study.* In the Proceedings of the 37th Conference on Winter simulation, 2408-2416.

[Paxson and Floyd, 1995] Paxson V., and Floyd S. 1995. *Wide Area Traffic: the Failure of Poisson Modeling.* In IEEE/ACM Transactions on Networking, Vol. 3, No. 3, 226-244.

[Pionteck et al., 2006] Pionteck T., Koch R., Albrecht, C. 2006. *Applying Partial Reconfiguration to Networks-on-Chips.* In Proceedings of 2006 International Conference on Field Programmable Logic and Applications. Madrid, Spain.

[Siriwong et al., 2007] Siriwong K., Lipsky L., and Ammar R. 2007. *Study of Bursty Internet Traffic.* In the Proceedings of the IEEE Symposium of Network Computing and Applications, 53-60, Cambridge, MA, USA.

[Sprint, 2008] Sprint Academic Research Group. 2008. ipmon.sprint.com.

[Thid et al., 2006] Thid R., Sander I., and Jantsch A. 2006. em Flexible Bus and NoC Performance Analysis with Configurable Synthetic Workloads. In Proceedings of the 9th EUROMICRO Conference on Digital System Design (DSD). 681-688. IEEE Computer Society. Cavtat, Croatia.

[Trolltech AS, 2007] Trolltech AS. 2007. *Qt 4.3.* Whitepaper. Norway.

[Vishwanath and Vahdat, 2006] Vishwanath K. V., and Vahdat A. 2006. *Realistic and Responsive Network Traffic Generation.* In Proceedings of the 2006 Conference on Applications, Technologies, Architectures, and Protocols For Computer Communications, Pisa, Italy, 111-122. ACM, New York, NY, USA.

[Wang et. al., 2002] Wang M., Chan N. H., Papadimitriou S., Faloutsos C., and Madhyastha T. M. 2002. *Data Mining Meets Performance Evaluation: Fast Algorithms for Modeling Bursty Traffic.* In Proceedings of the 18th International Conference on Data Engineering, 507-516, IEEE Computer Society. Washington, DC, USA.

[Weigle et. al., 2006] Weigle M. C., Adurthi P., Hernández-Campos F., Jeffay K., and Smith F. D. 2006. *Tmix: a Tool for Generating Realistic TCP Application Workloads in NS-2.* SIGCOMM Computer Communication Review, Vol. 36, Issue 3, 65-76.

Author Biographies

Carsten Albrecht received his Diploma degree in Computer Science from the University of Lübeck in 2002. In the same year, he joined the Institute of Computer Engineering, University of Lübeck. His current research topics are application-specific management of dynamically reconfigurable systems and network-on-chips. Further research interests include multi-threaded processor design and network processor architectures.

Christoph Osterloh studies Computer Science at the University of Lübeck with the objective of receiving his Diploma degree this year. His current research topics are hardware design and performance analysis in the area of reconfigurable computing and robotics.

Dr.-Ing. Thilo Pionteck received his Diploma degree in Electrical Engineering in 1999 and his PhD from the Technical University of Darmstadt in 2005. In the same year, he joined the Institute of Computer Engineering, University of Lübeck where he leads the "Reconfigurable Computing" research group. His research interests include design of dynamically reconfigurable systems, topology adaptive network-on-chips and network processors.

Roman Koch received his Diploma degree in Computer Science from the University of Lübeck in 2003. He is currently a PhD student at the Institute of Computer Engineering, University of Lübeck where he holds a position as a research assistant. His current research topic are design and prototyping of dynamically reconfigurable systems.

Prof. Dr.-Ing. Erik Maehle received his Diploma and Doctoral degree in computer science from the University of Erlangen-Nürnberg in 1977 and 1982, respectively. After subsequent positions as a postdoc at the IBM Zurich Research Laboratory and the University of Erlangen-Nürnberg he became a Professor at the Universities of Augsburg in 1987 and Paderborn in 1989. Since 1994 he is Director of the Institute of Computer Engineering at the University of Lübeck. He is Vice-Chair of the IFIP Working Group 10.3 Concurrent Systems, steering committee member of the German Special Interest Group on Parallel Processing PARS, member of the Steering Committees of the German Informatics Society (GI) on Computer and System Architecture ARCS as well as Fault-Tolerance and Dependability VerFe and speaker of GI Specialist Area Computer Engineering. His research interests include parallel and fault-tolerant computing, reconfigurable computing, robotics and embedded systems.

AN ALGORITHM FOR SPATIAL DATA CLASSIFICATION AND AUTOMATIC MAPPING BASED ON "SPIN" CORRELATIONS

Milan Žukovič
Department of Mineral Resources Engineering
Technical University of Crete
Chania 73100, Greece
Email: mzukovic@mred.tuc.gr

Dionissios T. Hristopulos
Department of Mineral Resources Engineering
Technical University of Crete
Chania 73100, Greece
Email: dionisi@mred.tuc.gr

KEYWORDS

spatial statistics, non-Gaussian distribution, Ising model, optimization, simulations

ABSTRACT

We employ the Ising model from statistical physics in the problem of spatial data classification. We use a multiple-class discretization of the sample values. The proposed algorithm predicts the class identity at unmeasured points based on Monte Carlo simulations that are conditional on the observed data (sample). The algorithm aims to minimize the deviation between the normalized correlation energies of the sample and the entire domain. A hierarchical scheme is used, in which points predicted to belong in lower-level classes retain their identity in the inference of the higher-level classes. The method is non-parametric and thus suitable for application to non-Gaussian data. The method is investigated using real data of surface elevation over a large part of the territory of North America. The effects of the ratio of training to prediction points, the number of classes, and the initial conditions are investigated. Potential extensions of the model are also discussed.

INTRODUCTION

The nowadays availability of large quantity of remotely-sensed geo-referenced data, such as land cover, terrain elevation, population, meteorological variables, atmospheric pollution, etc., creates increasing demands for efficient processing and analysis. The information contained in the data can provide reliable policy and management tools for application in diverse areas, such as ecosystem management, environmental policy, the design of real-time hazard warning systems, and various other decision making tasks. However, the data may be derived from different sources, measured by various systems using different methodologies and space-time resolutions. Furthermore, the data coverage is often incomplete due to different reasons, such as limited resources (material, human, technical), equipment limitations (detection level, resolution), meteorologi-

cal conditions (observations hindered by clouds). This situation leads to the missing data problem. Therefore, in order to use standard tools for the analysis of space-time data, there is a need to fill the data gaps and unify the resolution. These tasks involve down-scaling (refining) of the data with sparse resolution. They can be performed by means of well established interpolation and classification techniques (Atkinson et al., 1999). However, considering the ever-increasing size of spatial data, some classical methods used in geostatistics (e.g., kriging), e.g. (Wackernagel, 2003), can be impractical due to the high computational complexity involved. Furthermore, methods like kriging usually assume a jointly Gaussian distribution, an assumption that is often not justified by the data. In addition, application of such methods typically requires considerable human (subjective) input (Diggle et al., 2007).

A recently proposed approach focuses on the modelling of spatial correlations by means of short-range interactions, inspired from models of statistical physics (Hristopulos, 2003; Hristopulos and Elogne, 2007). These works focused on the development of computationally efficient spatial dependence models that are applicable to gridded and scattered Gaussian data. The objective of the present study is to extend the scope of the interaction-based representation to non-Gaussian data. Herein, we formulate a classification model inspired from the Ising spin model used in statistical physics and demonstrate its application on remotely sensed earth surface elevation data. The proposed model is computationally fast, it is non-parametric (i.e., it makes no assumption on the data distribution), and it requires minimal input by the user. Hence, it is suitable for the automatic classification and mapping of large, non-Gaussian data.

DEFINITION OF THE PROBLEM

Let us consider a set of sampling points $G_s = \{\vec{r}_i\}$, where $\vec{r}_i = (x_i, y_i) \in \mathbb{R}^2$ and $i = 1, ..., N$. These points are assumed to be scattered on a rectangular grid $\widetilde{G}$ of size $N_G = L_x \times L_y$, where L_x and L_y are respectively the horizontal and vertical dimensions of the rectangle (in terms of the unit length) and $N < N_G$. Let z_i be a value attributed

to the point $\vec{r}_i$. Then, the set $Z_s = \{z_i \in \mathbb{R}\}$ represent the sample of the process. Let $G_p = \{\vec{r}_p\}$ be the set of prediction points where $p = 1, \ldots, P$, such that $\tilde{G} = G_s \cup G_p$. For many applications Z_s can be represented as a realization of a continuous underlying random field $Z(\vec{r}_i)$. In the following we discretize the continuous distribution using a number of classes, $\mathcal{C}_q$, $q = 1, \ldots, N_c$. The classes are defined with respect to a set of threshold levels t_k, $k = 1, \ldots, N_c + 1$, where $t_1 = \min(z_1, \ldots, z_N)$ and $t_{N_c+1} = \max(z_1, \ldots, z_N)$. Each class $\mathcal{C}_q$ corresponds to an interval as follows: $\mathcal{C}_q = (t_q, t_{q+1}]$ for $q = 2, \ldots, N_c - 1$, $\mathcal{C}_1 = (-\infty, t_2]$, and $\mathcal{C}_{N_c} = (t_{N_c}, \infty)$. We define the *indicator field* $I_Z(\vec{r})$ to take integer values $q = 1, \ldots, N_c$ equal to the appropriate class index. In particular, $I_Z(\vec{r}) = q$ implies that $z(\vec{r}) \in \mathcal{C}_q$. The prediction problem is then posed as a classification problem, i.e., each point in G_p is assigned a class label. In order to estimate the class identity of Z at the prediction points, we use the well-studied Ising model, e.g. (McCoy et al., 1973). Once all the prediction points have been assigned to a class, a map of the process Z can be generated consisting of equivalent class (isolevel) contours.

THE ISING MODEL

Let us assume a set of variables s_i^q ("spins"), which can take the value $s_i^q = 1$ ("spin-up") or $s_i^q = -1$ ("spin-down"). The Ising model considers pairwise interactions between the spins, expressed by the following Hamiltonian:

$$H = -\sum_{i,j} J_{ij}\, s_i\, s_j - \sum_i h_i s_i. \qquad (1)$$

The first term corresponds to the "spin-spin exchange" interaction energy. The coupling strength J_{ij} controls the strength as well as the type of the interaction: if $J_{ij} > 0$ it is "ferromagnetic" or if $J_{ij} < 0$ it is "antiferromagnetic". The second term corresponds to the symmetry-breaking bias, which is caused by the presence of a site dependent "external field" h_i. The latter controls the overall distribution of the "spin" values between 1 and -1 (the magnetization). The coupling strength is usually considered to be uniform and its range limited to nearest neighbors. However, the model can be generalized to include also non-uniform, longer-range couplings.

The probability density function of the spin configurations is expressed following the Gibbs formalism as:

$$f = \frac{e^{-H/k_B T}}{Z[H]}, \qquad (2)$$

where k_B is the Boltzmann s constant and T the temperature. The partition function $Z[H]$ is a normalization factor obtained by summation of $e^{-H/k_B T}$ over all possible "spin

configurations". Hence, it is only a function of the model parameters J_{ij} and h_i, not of a particular configuration. In the forward problem, the coupling strength and the polarizing field are known and one is interested in the most probable spin configurations. In the inverse problem, the "spins" are known from the sampled values, and the estimation focuses on inferring the model parameters (e.g., by means of the maximum likelihood method). Following that, the optimal prediction of the "spin" values at unsampled locations can be performed by maximizing the conditional (on the data) probability f (or, equivalently by minimizing H) with respect to the unknown values. Unfortunately, the normalizing constant Z is in most cases intractable by analytical means, while its numerical evaluation is a computational bottleneck. Possible approaches to circumvent this problem, such as the maximum pseudo-likelihood approach (Besag, 1975) or various Markov Chain Monte Carlo estimation techniques (Chen et al., 1997), can be either very inaccurate or prohibitively slow respectively.

THE MULTILEVEL CLASSIFICATION ALGO-RITHM

Non-Parametric Nearest-Neighbor Model

In the following we use the ideas motivating the Ising Hamiltonian (Eq. 1). We propose a hierarchical scheme in which the sample, G_s^q and prediction, G_p^q grids are progressively updated for increasing class index q. For the lowest class $G_s^1 = G_s$, $G_p^1 = G_p$, where G_s and G_p are the initially defined grids. Let N_q denote the number of known spins at level q. The discretization is binary with respect to each threshold value, i.e., $s_{i_q}^q = -1$ if $z_{i_q} \leqslant t_{q+1}$ and $s_{i_q}^q = 1$ if $z_i > t_{q+1}$ for $i_q = 1, \ldots, N_q$. For $q > 1$ the sample (prediction) grid is augmented (diminished) by the points $\vec{r}_l$ for which $s_l^{q-1} = -1$. It follows that $N_1 = N$ and $N_{q>1} \geqslant N$. The set $S_s^q = \{s_{i_q}^q\}$, $q = 1, \ldots, N_c$; $i_q = 1, \ldots, N_q$ includes all the spin values for class index q. Let $\tilde{S}^q = S_s^q \cup S_p^q$ denote the "spin" values at the prediction grid $\tilde{G}$. In this framework, the Ising model can be used to represent spatial interactions between the spins $\tilde{S}^q$ with respect to the specific threshold.

In this study, we restrict the scope of the Ising model to the simplest energy functional: we set the polarizing field uniformly to zero, i.e., $h_i = 0, i = 1, ..., N$ and limit the exchange interactions to uniform strength, "ferromagnetic," i.e. $J_{ij} = J > 0$, with nearest neighbor coupling range. The choice of zero polarizing field does not allow to explicitly control the volume fraction of "up" versus "down" spins. As explained below, this is achieved in the simulations by selecting the initial "spin" values so as to reflect the "up-down" distribution of the sample.

The choices made above help to avoid the parameter inference problem and suggest a non-parametric approach.

This approach utilizes a cost function, $U(S_p^q|S_s^q)$, that describes the deviation (the error) between a suitably normalized energy of the simulated distribution, $\tilde{C}^q$, and its counterpart, C_s^q, for the sample distribution.

$$U(S_p^q|S_s^q) = \begin{cases} \left(1 - \tilde{C}^q/C_s^q\right)^2, & \text{for } C_s^q \neq 0, \\ \\ (\tilde{C}^q)^2, & \text{for } C_s^q = 0, \end{cases} \quad (3)$$

where $C_s^q = \langle s_i^q s_j^q \rangle_{G_s^q}$ is the spin pair correlation of the q−level sample and $\tilde{C}^q = \langle s_i^q s_j^q \rangle_{\tilde{G}}$ is the spin pair correlation over the entire grid.

Given the above, the classification problem of estimating S_p^q is reduced to finding the optimal configuration $\hat{S}_p^q$ corresponding to the minimum of the cost function (3) at a fixed temperature T:

$$\hat{S}_p^q = \underset{S_p^q}{\arg\min} \, U(S_p^q|S_s^q). \quad (4)$$

Hierarchical Classification Strategy

We propose a hierarchical algorithm that proceeds sequentially by increasing the class index. The binary discretization and the classification are first performed with respect to the first (lowest) class and then repeated sequentially for the remaining classes. The "gaps" in the prediction grid, G_p are gradually filled as the algorithm proceeds through consecutive levels. At each level, all the locations identified as having -1 spin values at the lower levels, are used as input (sample data) for the current classification. The reduced prediction grid, G_p^q for the class index q contains P^q points so that for $q > q'$ it holds that $P^q \leqslant P^{q'}$ and $P^1 = P$. The classes $\mathcal{C}_q$ can be defined as desired and do not need to be uniform. The algorithm uses the *rejection ratio*, which is defined by:

$$\rho = \frac{\text{number of rejected states}}{\text{number of simulated states}}.$$

The algorithm consists of the following steps:

Algorithm

(i) *Initialize the indicator field on the entire grid by means of $\hat{I}_Z(\tilde{G}) = \text{NaN}$*
(ii) *set the class index to $q = 1$*
(iii) while [1] $q \leqslant N_c - 1$ *discretize Z_s with respect to t_{q+1} to obtain S_s^q*
(iv) *given the data S_s^q, calculate the sample correlation energy C_s^q*
(v) *assign initial values to the spins at G_p^q, i.e., generate $\hat{S}_p^{q\,(0)}$*
(vi) *calculate the initial values of the simulated correlation $\tilde{C}^{q\,(0)}$ and the cost function $U^{(0)}$; initialize the simulated*

states index $i = 1$
(vii) *initialize the rejection ratio $\rho \to 0$; and the rejected states index $i_r = 0$*
(viii) while [2] $\rho < 1$ *repeat the following updating steps:*
   (a) generate a new state $\hat{S}_p^{q\,(i+1)}$ by perturbing $\hat{S}_p^{q\,(i)}$
   (b) calculate $\tilde{C}^{q\,(i+1)}$ and $U^{(i+1)}$
   (c) if $U^{(i+1)} < U^{(i)}$ accept the new state; $i_r \to 0$.
      else keep the "old" state; $i_r \to i_r + 1$; endif
   (d) $\rho \to i_r/P^q$; $i \to i + 1$;
end [2]
(ix) *assign the -1 "spins" to the q level, i.e., $\hat{I}_Z(\{\vec{r}_i\}) = q$ if $\hat{S}^{(i_{\max})}(\{\vec{r}_i\}) = -1$, $\{\vec{r}_i\} \in \tilde{G}$*
(x) *increase class index, $q \to q + 1$, return to step (iii)*
end [1]
(xi) *for $q = N_c$ set $\hat{I}_Z(\{\vec{r}_i\}) = N_c$ $\forall \vec{r}_i$ $(i = 1, \ldots, N_G)$ such that $\hat{I}_Z(\{\vec{r}_i\}) = \text{NaN}$.*

This algorithm returns an indicator field $\hat{I}_Z = I_Z(G_s) \cup \hat{I}_Z(G_p)$, which consists of the original sample discretization and the class estimates at G_p. The indicator values at the sampling sites are exactly reproduced, because the initial state respects these values and the updating steps skip over the sites in S_S^q. Below we refer to $I_Z(G_s)$ as the training set. The generation of new states in step (viii) is realized using single-spin Metropolis updating at zero temperature. The algorithm may sweep through the lattice several times, until the rejection ratio exceeds the threshold value. The $T = 0$ constraint means that there is no stochastic selection of unfavorable spins. Hence, the candidate "spin" for the update is flipped unconditionally only if it lowers the cost function. This is called a "greedy" Monte Carlo algorithm (Papadimitriou et al., 1982) and guarantees convergence, which is usually very fast. In comparison, in simulated annealing T is slowly lowered starting from an initial high-temperature state. This approach is much slower computationally, but the resulting configuration is less sensitive to the initial state $\hat{S}_p^{q\,(0)}$. The sensitivity of the greedy algorithm is known to be especially pronounced in high-dimensional spaces with non-convex energies. In such cases, the greedy algorithm is more likely to get stuck in local minima, instead of converging to the global one. However, this is not a concern in the current problem. In fact, targeting the global minimum of the cost fnction U strongly emphasizes the sample correlation energy per "spin" pair C_s^q, ignoring that the latter is influenced by sample-to-sample fluctuations.

The initial configuration can be selected in a number of ways. Since the proposed model aims to provide a fast and automatic classification mechanism, the initial configuration should minimize the relaxation path (in state space) to the equilibrium. It should also be selected preferably with little or no user intervention. Assuming a certain degree of spatial continuity, common in geospatial data sets, $\hat{S}_p^{q\,(0)}$

is determined based on the "augmented sample" states S_s^q in the immediate neighborhood of the individual prediction points. The neighborhood of $\vec{r}_p$ is determined by an $m \times m$ stencil (where $m = 2l + 1$) centered at $\vec{r}_p$. Then, $\hat{s}_p^{q\,(0)}$ is assigned by majority rule, based on the prevailing value of its neighbors from S_s^q that are situated inside the stencil. The stencil size is chosen automatically, reflecting the local sampling density and the "spin" values distribution. Namely, it is adaptively set to the smallest size, that contains a finite number of sampling points with one value (1 or -1) prevailing.

The updates are accepted unconditionally as long as they lower the cost function. The grid is swept randomly, point by point. The simulation terminates, for a given class index, if one complete sweep (i.e., P_q tried spin flips) through G_p^q does not produce a single successful update. This scheme implies an overall decreasing number of steps with increasing q in line with the size reduction of G_p^q. The parametrization required by the algorithm thus involves only the definition of the class intervals. The number of classes depends on the study objective: if the goal is to determine exceedance levels a binary classification is sufficient. For environmental monitoring and decision making purposes a moderate number (e.g., eight) of classes is often sufficient.

DATA AND SIMULATIONS

The data used in this study represent the surface elevation, $Z(\vec{r})$, on a 5-minute latitude/longitude grid, over a part of the territory of North America (approx. $80 - 110°$ W, $55 - 40°$ N). The data form a rectangle of 400×200 pixels (NOAA, 1988). Elevation values are in whole meters, representing the elevation of the center of each cell, with a resolution of 1 meter. Some summary statistics: number $= N_G = 80\,000$, $z_{\min} = 1$, $z_{\max} = 3790$, $\bar{z} = 774.41$, $z_{0.50} = 441$, $\sigma_z = 713.17$. The value of the skewness coefficient is 1.37 and of the kurtosis coefficient 4.07. As evidenced from the above statistics, the data are non-Gaussian and positively skewed. The histograms, corresponding to the respective class widths considered in the study, will be presented later.

From the complete data set, we generate a sample Z_s of size $N = (1 - p)N_G$ by randomly removing $P = pN_G$ values (validation points). For different degrees of thinning, $p = 0.33, 0.5$ and 0.66, we generate 100 different configurations of removed points. The values at the prediction points are then classified using the multilevel classification algorithm. In the classification performance evaluation, the indicator values at the prediction points $I_Z(G_p)$ will be compared with the estimates $\hat{I}_Z(G_p)$, obtained after removing the points G_p from the data. We use two resolutions and respective class intervals. First, a resolu-

tion of 500 m segregates the data into eight classes $\mathcal{C}_q = [500(q - 1), 500q]$, $q = 1, \ldots, 8$. Second, a finer resolution of 250 m resulting in $N_c = 15$ classes corresponding to the intervals $\mathcal{C}_q = [250(q - 1), 250q)$, $q = 1, \ldots, 15$ is used.

In order to evaluate the classification performance, we calculate the misclassification rate as a fraction of misclassified pixels, defined by

$$F = \frac{1}{P} \sum_{p=1}^{P} \left[1 - \delta\big(I_Z(\vec{r}_p), \hat{I}_Z(\vec{r}_p)\big) \right], \qquad (5)$$

where $I_Z(\vec{r}_p)$ is the true value at the validation points, $\hat{I}_Z(\vec{r}_p)$ is the classification estimate and $\delta(I, I') = 1$ if $I = I'$, $\delta(I, I') = 0$ if $I \neq I'$.

The computations are performed in the Matlab® programming environment on a desktop computer with 1.93 GB of RAM and a the processor Intel®Core ™2 CPU 6320 @ 1.86 GHz 1.86 GHz.

RESULTS

Resolution 500 m - 8 classes
The isolevel map and the histogram of the original complete data, discretized according to the vector of thresholds corresponding to this resolution, are shown in Fig. 1. In the map, the elevations in the range of $0 \leqslant Z < 500$ m (first class) dominate, covering about 55% of the area, while those above 3500 m correspond only to 0.1%. Fig. 1(b) presents the histogram of the class values for the original data (left histogram bar) as well as the histograms of the data reconstructed by the classification for the three degrees of thinning. The histograms resulting by application of the classification algorithm to the training sets with $p = 0.33, 0.5$ and 0.66, are shown by the respective bars $2 - 4$ (moving from left to right). The histograms represent mean values obtained from 100 realizations. The match between the distributions of the original and the reconstructed data is excellent. As mentioned above, this was achieved without explicit control of the data volume fraction ($h_i = 0$ in Eq. 1).

Fig. 2 helps to visualize the classification results in terms of reconstructed maps. The isolevel maps on the left are obtained from a single realization (the first from the set of one hundred). The maps on the right represent the standard deviations of the class indices obtained from the 100 realizations. We observe that the reconstructed maps provide a close visual match to the original map, shown in Fig. 1(a), especially at lower p. Misclassification mainly occurs along the borders between different classes. The effect of class borders is better demonstrated in the plots of the right column, which show the spatial distribution of the standard deviations of the estimates. Note that the
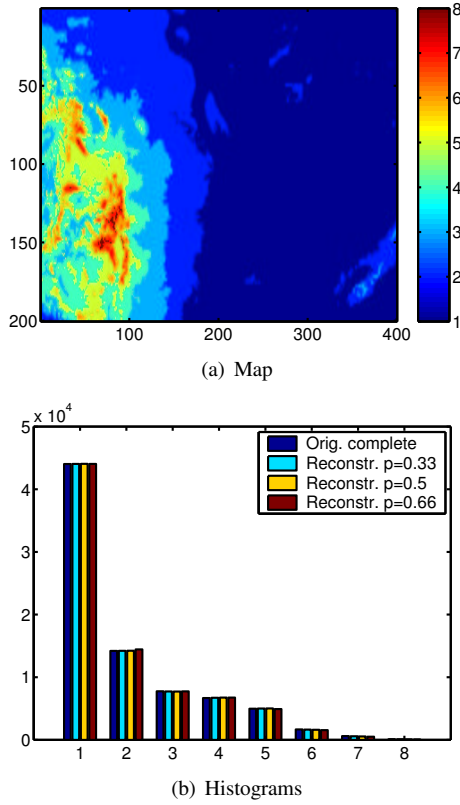
(a) Map

(b) Histograms

Figure 1: (a) 8-class isolevel map of the original complete data with the resolution 500 m. (b) Group of histograms. From left to right: the original complete data and the reconstructed ones from the thinned data with $p = 0.33, 0.5$ and 0.66, respectively. The histograms of the reconstructed data represent mean values, obtained from 100 realizations.

(a) p=0.33 Reconstructed map

(b) p=0.33 Standard deviations

(c) p=0.5 Reconstructed map

(d) p=0.5 Standard deviations

(e) p=0.66 Reconstructed map
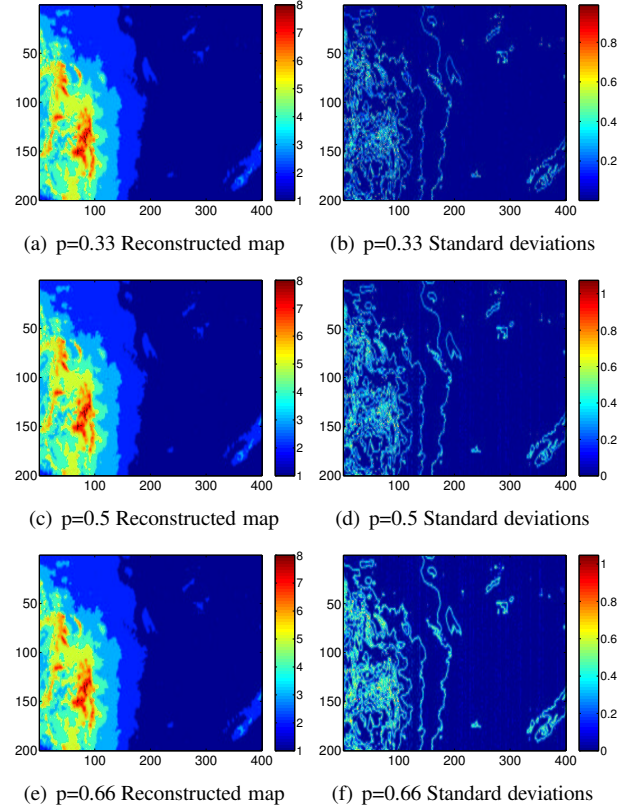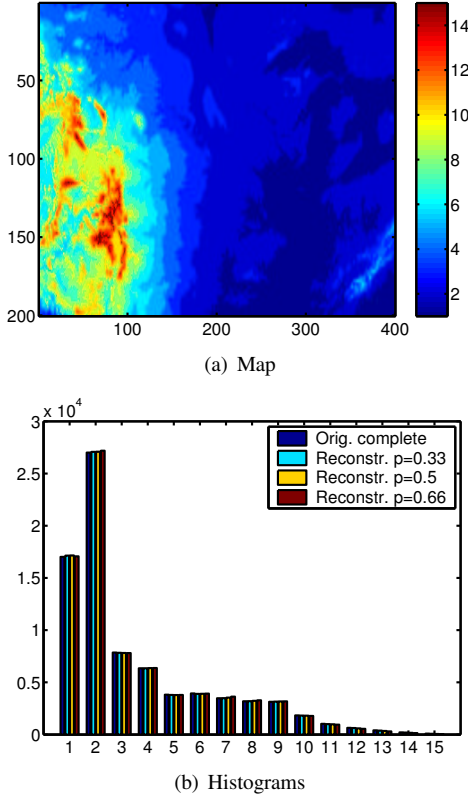
(f) p=0.66 Standard deviations

Figure 2: Examples of class maps obtained from individual realizations (a,c,e) and standard deviations of the class indices calculated from 100 different realizations (b,d,f). The data are reconstructed from samples with thinning degrees $p = 0.33, 0.5$ and 0.66, respectively. An 8-level classification scheme is used.

sampling points locations G_s (vary from realization to realization) bring zero contribution by construction, due to honoring the values at G_s. The spatial patterns in the original map, however, are reconstructed reasonably well even at $p = 0.66$.

Table 1 presents quantitative measures of the classification performance. It includes the mean value of the misclassification rate $\langle F^* \rangle$ and its standard deviation STD_{F^*}. The optimization s computational performance is measured in terms of mean number of Monte Carlo steps needed to achieve the optimum $\langle N_{MC} \rangle$, the optimization CPU time $\langle T_{cpu} \rangle$, and the cost function mean value $\langle U^* \rangle$ at termination. As expected, the misclassification rate increases with p.

The increase of $\langle N_{MC} \rangle$ with p is due to the fact that $\langle N_{MC} \rangle$ is a measure of the "spin" system relaxation time, while an increasing p translates into higher N and, thus, also larger state-space size 2^N. Since N_q decreases with

increasing q, due to the progressive filling of the gaps, the Metropolis sampler in general speeds up as q increases. The relaxation time is shortened by proper choice of the initial state. To give an approximate estimate, 2.08×10^5 Monte Carlo steps at $0.66 \times 400 \times 200$ prediction points correspond to roughly 4 grid sweeps. There are inter-level differences in the value of $U(S_p^q | S_s^q)$. Nevertheless, even at $p = 0.66 \max(U^*) < 7 \times 10^{-4}$. The average CPU time needed for the optimization at $p = 0.33, 0.5$ and 0.66 is approximately 6, 8, and 14 seconds, respectively.

Resolution 250 m - 15 classes

Next, we repeat the classification experiment using a resolution of 250 m. The isolevel map 3(a) has higher resolution than the eight-level one in 1(a). The most and least represented classes are the second and the last ones, which contain approximately 34% and 0.1% of the values, respectively. Due to the higher resolution an increase in the mis-

Table 1: Eight-level classification performance measures: mean values of the misclassification rate $\langle F^* \rangle$ and its standard deviation STD_{F^*}, mean number of Monte Carlo steps $\langle N_{MC} \rangle$, optimization CPU time $\langle T_{cpu} \rangle$, mean value of cost function at termination $\langle U^* \rangle$. The averaging is performed over 100 realizations.

8 classes	$p = 0.33$	$p = 0.5$	$p = 0.66$
$\langle F^* \rangle$ [%]	5.27	5.79	6.48
STD_{F^*}	0.12	0.10	0.12
$\langle N_{MC} \rangle$	9.10e+4	1.35e+5	2.08e+5
$\langle T_{cpu} \rangle$ [s]	5.55	7.95	13.60
$\langle U^* \rangle$	8.84e-5	1.44e-4	1.83e-4

(a) Map

(b) Histograms

Figure 3: (a) 15-class isolevel map of the original complete data with the resolution 250 m. (b) Group of histograms. From left to right: the original complete data and the reconstructed ones from the thinned data with $p = 0.33, 0.5$ and 0.66, respectively. The histograms of the reconstructed data represent mean values, obtained from 100 realizations.

classification rate is expected. Nevertheless, as we can see in Fig. 4, both the class distributions and the visual patterns are recovered quite well in all cases. The statistics of the

(a) p=0.33 Reconstructed map

(b) p=0.33 Standard deviations

(c) p=0.5 Reconstructed map

(d) p=0.5 Standard deviations

(e) p=0.66 Reconstructed map
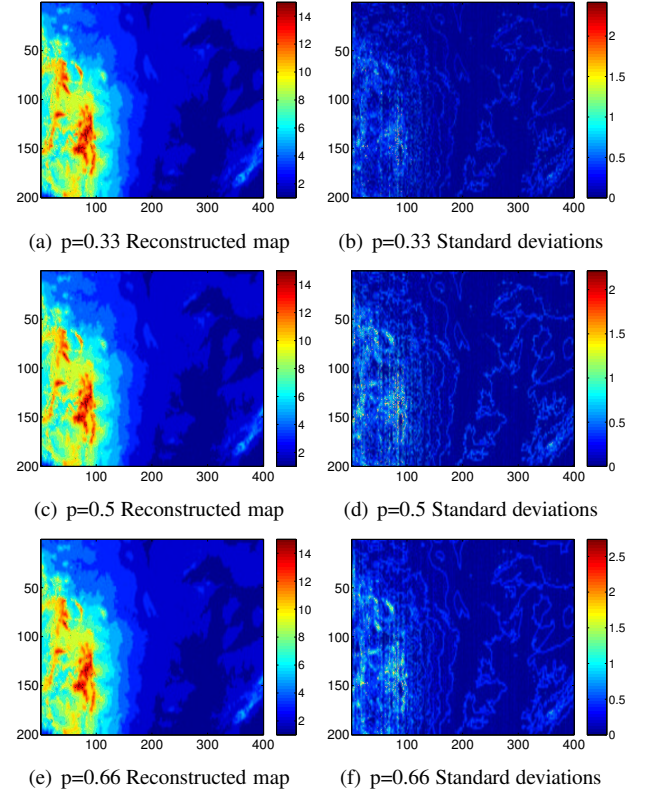

(f) p=0.66 Standard deviations

Figure 4: Examples of class maps obtained from individual realizations (a,c,e) and standard deviations of the class indices calculated from 100 different realizations (b,d,f). The data are reconstructed from samples with thinning degrees $p = 0.33, 0.5$ and 0.66, respectively. An 15-level classification scheme is used.

15-level optimization are given in Table 2.

CONCLUSIONS

We presented a novel classification model suitable for automatic mapping of large spatial data sets and validated it on the real world surface elevation data. The model is inspired from statistical physics and is based on minimizing distance between the normalized discrete correlation energies of the sample and the prediction domain data. It is computationally fast, universal with regard to the data distribution (makes no assumptions), and is almost automatic. The only parametrization in the proposed approach involves only the number of classes.

We have also used a deterministic updating scheme (results not shown here), in which the lattice is swept through sequentially. This scheme converges faster than the stochastic updating. For comparison, using the deterministic updating for the eight-level classification with thinning degree $p = 0.33$ we obtain $\langle T_{cpu} \rangle = 4.11$ sec,

Table 2: Fifteen-level classification performance measures. The same quantities are tabulated as in Table 1.

15 classes	$p = 0.33$	$p = 0.5$	$p = 0.66$
$\langle F^* \rangle$ [%]	11.93	12.93	14.23
STD_{F^*}	0.19	0.16	0.15
$\langle N_{MC} \rangle$	1.58e+5	3.16e+5	5.13e+5
$\langle T_{cpu} \rangle$ [s]	10.20	16.43	29.15
$\langle U^* \rangle$	9.07e-5	1.50e-4	1.97e-4

$\langle N_{MC} \rangle = 4.03 \times 10^4$ in contrast with $\langle T_{cpu} \rangle = 5.55$ sec and $\langle N_{MC} \rangle = 9.10 \times 10^4$ for the stochastic scheme. The misclassification rate is almost identical, $\langle F^* \rangle \approx 5.26\%$. The disadvantage of the deterministic scheme is that it generates certain artifacts. One of them is increased sensitivity to the initial conditions used, leading to high variability near the domain edge where the updating originates. Although we did not observe it at $p = 0.33$, it becomes more pronounced as p increases.

The model is demonstrated on a regular grid, however, the extension to irregularly spaced sampling points is straightforward. The interaction constant J_{ij} in Eq. (1) can be defined through a kernel function (such as the radial basis function), and the interaction neighborhood (nearest neighbors), for example, as pairs of points whose Voronoi cells have a common boundary. Further possible extensions could include incorporation of further-neighbor or/and "multi-spin" correlation energy in the Hamiltonian. Another possibility is to consider the q-state Potts model, instead of the sequential Ising one. Then the simulations could be performed simultaneously at all levels, however, as we also observed in our tests, the increased state space would result in much slower relaxation and hence much longer optimization times. The advantage would be in the initial state selection, since the "spins" of a range of states within the stencil would participate in the majority vote.

ACKNOWLEDGMENTS

REFERENCES

Atkinson P. M., Tate, N. J. (Eds.) (1999). *Advances in Remote Sensing and GIS Analysis*. John Wiley & Sons.

Diggle P. J., Ribeiro, P. J. (2007). *Model-based Geostatistics. Series: Springer Series in Statistics*. Springer.

McCoy B. M., Wu, T. T. (1973). *The Two-Dimensional Ising Model*. Harvard University Press, Cambridge Massachusetts.

Besag, J. (1975). Statistical analysis of non-lattice data. *The Statistician*, 24:179–195.

Chen, M. H., Shao, Q. M. (1997). On Monte Carlo methods for estimating ratios of normalizing constants. *Ann. Statist.*, 25:1563–1594.

Hristopulos, D. T. (2003). Spartan Gibbs random field models for geostatistical applications. *SIAM J. Scient. Comput.*, 24(6):2125–2162.

Hristopulos, D. T. and Elogne, S. N. (2007). Analytic properties and covariance functions for a new class of generalized Gibbs random fields. *IEEE Trans. Inform. Theor.*, 53(12):4467–4679.

Data Announcement 88-MGG-02, Digital relief of the Surface of the Earth. NOAA, National Geophysical Data Center, Boulder, Colorado, 1988.

Papadimitriou C. H., Steiglitz, K. (1982). *Combinatorial Optimization*. Prentice Hall.

Wackernagel H. (2003). *Multivariate Geostatistics*. Springer.

AUTHOR BIOGRAPHIES

MILAN ŽUKOVIČ was born in Svidnik, Slovakia and obtained his PhD degree in applied physics from Kyushu University, Japan in 2000. He pursued his research in the field of condensed matter physics at Kyushu University for 2 more years, before he assumed a position in automotive industry within Yazaki Corporation. Since October 2006, he has been involved in the research in modeling of spatial random fields as a Marie-Curie postdoc at Technical University of Crete, Greece. His e-mail address is : mzukovic@mred.tuc.gr.

DIONISSIOS HRISTOPULOS was born in Neo Faliro (Pireas), Greece. He holds a Diploma in Electrical Engineering (1985) from the National Technical University of Athens (Greece) and a PhD in Physics from Princeton University (1991). After two years of mandatory military service in Greece, he joined the Department of Environmental Sciences and Engineering at the University of North Carolina (Chapel Hill, USA). In 2000 he moved to the Pulp and Paper Research Institute of Canada to work in the field of Paper Physics. For research published with T. Uesaka, they were awarded the 2003 Johannes Van den Akker Prize for Advances in Paper Physics. Since 2002 he has been with the Department of Mineral Resources Engineering, at the Technical University of Crete. He recently became Professor of Geostatistics. His email is dionisi@mred.tuc.gr and his personal webpage at http://www.mred.tuc.gr/home /hristopoulos/dionisi.html.

SIMULATION OF THE HYDRAULIC CIRCUIT OF A WHEEL LOADER BY USING THE BOND GRAPH TECHNIQUE

Gregorio Romero, Jesús Félez, M. Luisa Martínez, J. José del Vas
Engineering Graphics and Simulation Group
Department of Mechanical and Manufacturing Engineering
Universidad Politécnica de Madrid
C\José Gutierrez Abascal Nº2, CP. 28006, Madrid (Spain)
E-mail:{gregorio.romero; jesus.felez; luisa.mtzmuneta}@upm.es, jdelvas@industriales.upm.es

KEYWORDS

Bond Graph, hydraulic circuit, wheel loader, valves, pump, hydraulic cylinder.

ABSTRACT

Hydraulic systems are widely used in Engineering because they are highly versatile and can be combined with other domains of physics (mechanics, electronics, etc.). For this reason, it is of vital importance to study and develop them with a view to obtaining mechanical-hydraulic systems that are increasingly efficient. The beginning of this paper briefly analyses the method usually used to perform simulation in the hydraulic domain. Subsequently, after performing an analysis of different Bond-Graph technique-based works, different characteristic elements will be individually analysed. Finally, the different elements analysed will be interconnected and a simulation of a hydraulic circuit will be performed in order to obtain a series of results that will enable the different rationales set out in this paper to be validated. Finally, a series of conclusions will be reached based on the work presented.

1. INTRODUCTION

Hydraulic systems modelling and simulation has been performed using a whole range of techniques, although currently a lot of territory remains to be researched in this area of Engineering. Traditionally, algebraic-differential equations have been proposed based on the corresponding laws of physics (Cobo et al. 1998.). They have then been solved in various environments like MATLAB, Simulink, and MAPLE, to name but a few. One of the drawbacks of this procedure lies in the fact that obtaining algebraic-differential equations is usually complex as well as the procedure for solving them, and on many occasions so much time is spent that real time simulation cannot be performed while costs also increase significantly.

Another possible option is to generate the model from zero using specific software, either by the finite elements method, using block diagrams, or using graphic techniques. These types of simulations are often oriented towards specific applications, frequently technological ones, and are therefore mainly focused on obtaining graphic or numerical results (*Hydro+Pneu*, *OHC-Sim*, *HOPSAN*, *LVSIM*) and move away from obtaining equations for the model.

Although there is still room for it to be improved and optimised in different areas of Engineering, the Bond-Graph technique (Karnopp et al. 1990), lets a whole range of systems be modelled and then simulated (Gawthrop and Smith 1992; Borutzky and Cellier 1996; Félez et al. 2000). Hydraulic systems can be dealt with in this way, which speeds up the process considerably, and reduces the time needed for modelling and obtaining the algebraic-differential equations. This is reflected in a reduction of process-related costs and an increase in efficiency. In the hydraulic scope of this technique the study has focused on fluid propagation characteristics (Borutzky 1995) or on simulating different industrial applications (Bunes and Einang 2000; Kun et al. 1999; Wanli and Fumin 2005,), some of them in combination with other domains such as thermodynamics (Thétiot and Zouaoui 1998). Regarding the simulation of a hydraulic circuit similar to that of the machine in question, some preliminary work has to be done (Krishnaswamy and Li 2006; Nadjm-Tehrani and Strömberg 1999) but the main aim is not the model itself but to optimise and regulate the model.

As can be seen, a lot remains to be researched and improved in the study of hydraulic systems using the Bond-Graph technique. The contribution of this paper to the existing bibliography is that it shows how the different hydraulic parts can be modelled and then connected in order to simulate a wheel loader. To do this, *Bondin* software (Romero et al. 2005; Romero et al. 2005), is used; It is an enhanced version of another program previously developed (Félez et al. 1990) by one of the authors.

Finally, this paper is intended to show that it is not always necessary to simulate a specific set of parts at a determined instant, which would cause the simulation to be excessively slow. However, it is possible to obtain a series of behavioural laws based on the results of the simulation that will improve computational time.

2. VALVES

As a prior step to developing the different models, apart from other ports that can be used, it should be pointed out that there is an Inertia-type port, and another Capacitor-type port that represent the compressibility of the fluid and its mass, respectively, the values of which are listed below.

- Compressibility:

$$K = \frac{B}{A \cdot L} = \frac{4 \cdot B}{\pi \cdot D^2 \cdot L} \qquad (1)$$

> B: Bulk fluid module.
> D: Inside diameter of the pipe section.
> L: Pipe length.

- Mass:

$$I = \frac{\rho \cdot L}{A} = \frac{4 \cdot \rho \cdot L}{\pi \cdot D^2} \qquad (2)$$

> ρ: Fluid density.
> D: Inside diameter of the pipe section.
> L: Pipe length.

These equations partly differ from those given by other authors (Rosenberg and Karnopp 1983) as it has been assumed there is no deformation in the pipe.

2.1. Distributor valves

This type of valve is a hydraulic component designed to let the fluid pass through a pipe, prevent it passing or change its direction in any kind of circuit. They are usually assigned the function of regulating the starting, stopping and change in the direction of movement of the cylinder rods. Although this is the basic task of these valves they are also used for other functions such as directing the fluid to other valve controls, to the tank discharge or to the manometers, for example.

In these valves the passage of fluid is either total or zero. That is, either all the flow reaches the inlet pipe or none passes through; the same occurs with the outlet pipes.

Classic directional valves, also called distributors, can have two, three or more positions, and have two, three or more internal communication circuits. The positions are schematically represented by squares, those with two or three positions being mostly used, while the circuits are indicated by flow lines drawn on the inside of the squares. The arrows on these lines indicate the direction of flow, although in many applications the direction can be opposite to that shown.
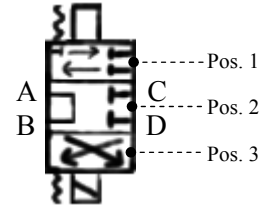

Figure 1: Graphic representation of a 3x2 valve

This valve can be modelled using the Bond Graph technique by representing in a single model the different internal communication circuits between the different inputs (A, D) and outputs (B, C), and then fitting in as many resistances as are needed to configure the different positions.
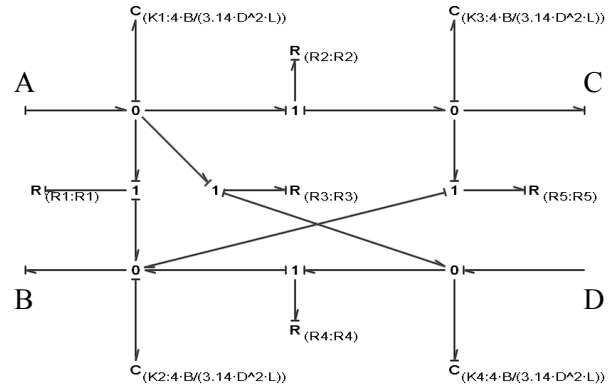

Figure 2: 3x2 valve model using Bond-Graph.

In the above figure the assembly of the valve model to the rest of the model or other elements would be done through the bonds at the ends (A, B, C, D).

The C capacitor ports represent fluid compressibility and must be placed so that the causality in the type R ports is appropriate. The type R ports are entrusted with ensuring the fluid passes one way or another by means of its value. All of them carry effort causality on the port with the purpose of defining the effect on the flow they let pass, since it is verified that:

$$Q = \frac{P}{R} \Rightarrow \begin{Bmatrix} Si\ R \downarrow \Rightarrow Q \uparrow \\ Si\ R \uparrow \Rightarrow Q \downarrow \end{Bmatrix} \qquad (3)$$

In the above expression P stands for the effort existing in the bond that reaches port R. therefore, depending on the difference of efforts port R defines the flow that passes through the communication circuit. If R has a very low value an open pass is being simulated, while if it is very high, the pass is closed, as the flow defined by the resistance port will tend to zero. Thus, by controlling the value of R, the flow passing through each valve circuit is also controlled, which means the behaviour of a valve of this type is being simulated.

The type R ports are associated with type 1 junctions to be able to define the flow volume that will flow through

the rest of the bonds, and since the 1 junction is of this type the flow volume will be maintained. However, the C ports are associated with type 0 junctions since they indicate a loss of volume.

Below is the code used to change the different positions of this type of valve.

```
if (position == 1)
{
  R1=10e15;
  R2=10e2;
  R3=10e15;
  R4=10e2;
  R5=10e15;
}
else if (position == 2)
{
  R1=10e2;
  R2=10e15;
  R3=10e15;
  R4=10e15;
  R5=10e15;
}
else if (position == 3)
{
  R1=10e15;
  R2=10e15;
  R3=10e2;
  R4=10e15;
  R5=10e2;
}
```

2.2. Non-return valve

This type of valve is capable of letting the fluid pass in one direction but not in the opposite direction. It is closed by using spherical, conical or plane seats, accompanied by a return spring that obliges the fluid flowing in the free direction to overcome a minimum pressure, almost always of a small value, to be able to produce the opening.

Figure 3: Graphic representation of a non-return valve

In the Bond-Graph technique this type of valve can be modelled by placing a type R port between the input and the output to let the fluid pass in one direction but not in the opposite direction and by placing a type 1 port to determine the direction of flow and thus be able to govern the value of resistance R.
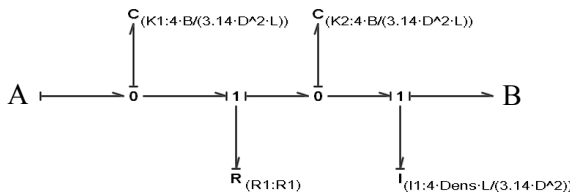
Figure 4: Non-return valve model using Bond-Graph.

As with the previous valve a type R port must be inserted to let the fluid pass or not. In the above figure,

when the volume goes from left (A) to right (B), the value of resistance R must be very low (only the value of the fluid load loss as it passes through the valve) in order to let the fluid flow freely, while if the direction of volume is reversed (from right to left) the value of R must be high so that no volume can flow.

In order to define the flow volume communicated to the rest of the model the causality needs to be of effort on the R ports. This is achieved by adding two type C ports to represent fluid compressibility. It is also necessary to add a type 1 port to represent the fluid mass that is in motion. Because this latter port is causality of flow on this port, inserting it causes the causality of the bond at point A to be the same as at point B, thus causality is not changed on the route.

Therefore, the direction of flow of the fluid can be determined either by the flow of the type 1 port or by comparing the efforts existing in the type C ports. Thus, the direction of volume will be positive providing the inertia flow is positive or when the effort in the first capacitor is greater than in the second one.

Shown below is the code used to change the value of the type R port automatically.

```
if (effort(K1)>=effort(K2))
{
  R1=10e2;
}
else if (effort(K1)<effort(K2))
{
  R1=10e15;
}
```

As can be seen, when the effort at the right is greater than the effort at the left the fluid tries to flow from right to left and its passage is prevented due to the high values assigned to R, while if the effort at the left is greater than the effort at the right, passage is allowed, with value R being the corresponding load loss through the valve.

2.3. Pressure limiter valve

Pressure limiter valves control the maximum pressure in the circuit by means of an electrical input signal that is proportionally converted into a pressure on the fluid.

According to figure 5, the fluid should logically pass from point A to point B. However, under conditions of overpressure in the circuit (point B) the valve opens and lets the fluid go directly to point C, that is, the tank. By this means, the maximum pressure at point B of the circuit is limited.
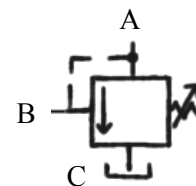

Figure 5: Graphic representation of pressure limiter valve

To make the simulation model in a Bond-Graph the three branches have to be inserted; one input (A) and two outputs (B, C), and a type R port inserted to let the fluid pass or not to the tank branch (C), where the pressure is zero. The value of this R port will depend on the effort existing in the bond node of the three branches, which means an additional port needs to be inserted in order to determine it. This other port is a type C port and represents fluid compressibility; on the other hand it serves to make the causality in the type R ports the appropriate type and on the other hand allow obtaining the pressure related to the compression of the fluid.
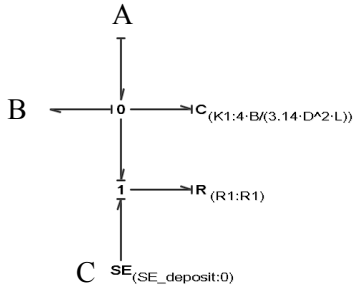
Figure 6: Pressure limiter valve model in Bond-Graph.

In the above figure the type R port regulates the volume crossing the branch over which it is placed. Therefore, if the value of R s very high all the volume entering through branch A leaves through branch B, and as a result the limiter valve is closed. To the contrary, if the pressure in the circuit (branch B and logically branch A, in which exist the same pressure) reaches a maximum value, the value of R needs to be a minimum so as to let all the volume pass to the tank branch (C).

Shown below is the code used to change the value of the type R port automatically depending on the maximum value of the pressure.

```
if (effort(K1)>limit_effort)
{
  R1=10e2;
}
else if (effort(K1)<=limit_effort)
{
  R1=10e15;
}
```

It can be seen that when the effort in the capacitor is greater than a maximum value the valve opens *(effort(K1)>limit_effort)* and if that is not the case, it closes *(effort(K1)<=limit_effort)*.

2.4. One-way flow limiter valve

The main purpose of the valves regulating volume or flow is to control the amount of fluid sent to the actuators or working parts. When constant flow pumps are installed in the circuit, the maximum volume flowing through a circuit can never be more than what the pump is capable of generating, which means the flow regulators usually limit the volume to this value

sending the rest of the fluid to the tank.

This type of valve can regulate the flow either in one direction or both. In this case, a valve that regulates flow in one direction but not in the other will be described.

Figure 7: Graphic representation of a one-way flow limiter valve

It can be modelled in a Bond-Graph by placing the two branches in parallel, each with its type R ports between the input and the output to regulate the passage of fluid. The R port placed in the upper branch must let all the fluid pass in one direction (from A to B) and prevent it from doing so in the opposite direction (from B to A), while the task of the port placed in the lower branch will be to regulate the volume passing through the valve when the fluid is circulating in a particular direction (from B to A).
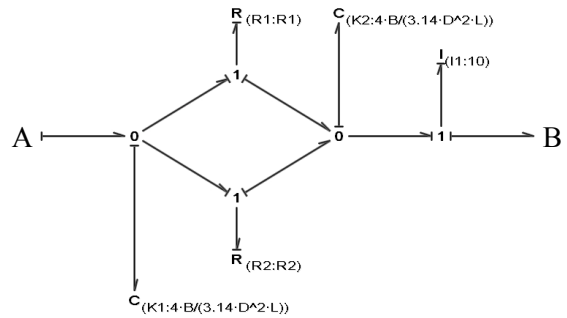

Figure 8: One-way flow limiter valve model in Bond-Graph.

As with the non-return valve, the direction of flow can be determined by inserting a type 1 port to represent the fluid mass that is in motion or by comparing the efforts existing between the input and the output.

According to the above figure, port R1 represents the non-return and port R2 represents the narrowing. Thus, if the direction of flow is from A to B, resistance R1 must have a very low value so the fluid can pass to that branch instead of passing through the other (resistance R2), thereby not producing any narrowing whatsoever. If the direction of flow is from B to A, resistance R1 must have a very high value to make the fluid pass through the other branch (resistance R2) and so produce the load loss in the narrowing.

Ports C1, C2 and I1 are built into the model to make the causality appropriate in each of the R ports, and so regulate the volume circulating depending on the direction of flow of the fluid. The port values are those seen in the other models, with the exception of the I1 port, which in this case has a low value as its purpose is to obtain appropriate causality in port R1, as well as to impose the conditions governing the values of the R

ports.

In order for this type of valve to function properly the code needed is shown below, where what has been explained above can be clearly seen:

```
if (effort(K1)>=effort(K2))
{
  R1=10e2;
  R2=9500;
}
else if (effort(K1)<effort(K2))
{
  R1=10e15;
  R2=9500;
}
```

Therefore, when the flow of the type 1 port is positive I *(flow(I1)>=0)* the fluid circulates from A to B without any narrowing since the fluid can flow freely through the upper branch, while when the flow is negative, *(flow(I1)<0)* narrowing occurs when the fluid is made to flow through the lower branch.

3. HYDRAULIC CYLINDER

Cylinders are the working parts of hydraulic circuits that are most frequently used in machines. Using the flow and pressure provided by the pumps the work is developed in these parts through a straight-line forward and backward movement that takes place in the different phases of a cycle.

However, they can be classified into two types: simple acting and double acting. The first of the two types uses the force from the hydraulic circuit to carry out one of the movements, and an external force like that due to the action of a spring or to the force of gravity to carry out the opposite movement. The second type uses hydraulic force to effect both movements, whether or not helped by external forces.
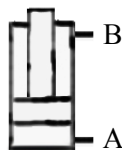

Figure 9: Graphic representation of a hydraulic cylinder.

In a Bond-Graph, a hydraulic cylinder is modelled by initially putting in place the input and output branches and then the mechanical forces appearing on the piston rod, which in figure 9 are the weight and the friction between the cylinder and its housing. It is important to bear in mind that two domains are being worked in, hydraulics and mechanics. This means that hydraulic power has to be transformed into mechanical power and vice-versa, using two TF elements placed at the input and output of the cylinder.

In figure 10 the K1 port placed next to input A represents fluid compressibility and also has the task of making the causality in port R1 be of flow entering the system. This causality must be so because this port will define the limits of the piston displacement (*Lmax* and *Lmin*), so that when the piston reaches the ends, value R1 is very high and so interrupts the fluid flow. To obtain the distance run by the cylinder from its initial position a type 1 port has been assigned to it, over which, in addition, the effect of gravity appears.
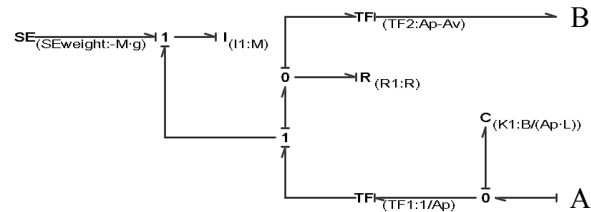

Figure 10: Hydraulic cylinder model in Bond-Graph.

Regarding the TF elements, the one placed at the beginning of the distance displacement serves to transform the hydraulic domain variables to the mechanical domain by using the piston area (*Ap*), while the one placed at the end serves to make the opposite transformation by taking the piston area less that of the piston rod (*Ap-Av*).

4. PUMP AND ENGINE

An engine is an element belonging to the field of rotational mechanics that transforms one kind of energy, either electrical, thermal or hydraulic into mechanical energy at the output shaft. From a Bond-Graph point of view it is represented as a flow source in units of radians/second, regardless of the type of engine, as we are dealing with an angular velocity source.
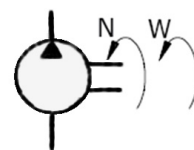

Figure 11: Graphic representation of a hydraulic pump.

As for hydraulic pumps (fig.11), they are an element, which after being driven by an engine to an angular velocity ω with a torque *N*, generates a flow *Q* of fluid under a specific pressure *P*. The basic characteristic of a hydraulic pump is its cylinder capacity per rotation, which defines the volume of fluid generated by each rotation of the pump shaft. In the Bond Graph technique these are represented by a TF type element, whose ratio is the cylinder capacity per rotation, and it is this that makes the change from the physical domain of rotational mechanics (input) to the hydraulic domain (output). It is then connected to the input and output branches using a type 1 node, by means of which the flow volume of both branches will be maintained but not the pressure, which at the input is atmospheric

pressure and at the output is the input pressure plus the pressure contributed by the pump.

5. HYDRAULIC CIRCUIT OF THE MACHINE

The mission of a wheel loader's hydraulic circuit is to distribute the flow volume and pressure supplied by the pumps to the different hydraulic cylinders. Due to the work developed through the straight line forward and backward movement that takes place in the different phases of a cycle, these cylinders are responsible for loading and unloading the material as well as rotating the machine. The circuit comprises an oil tank, two pumps whose job is to drive the flow through the whole circuit, different distributor, non-return and pressure limiter valves, filters and hydraulic cylinders, as may be seen in the following figure.
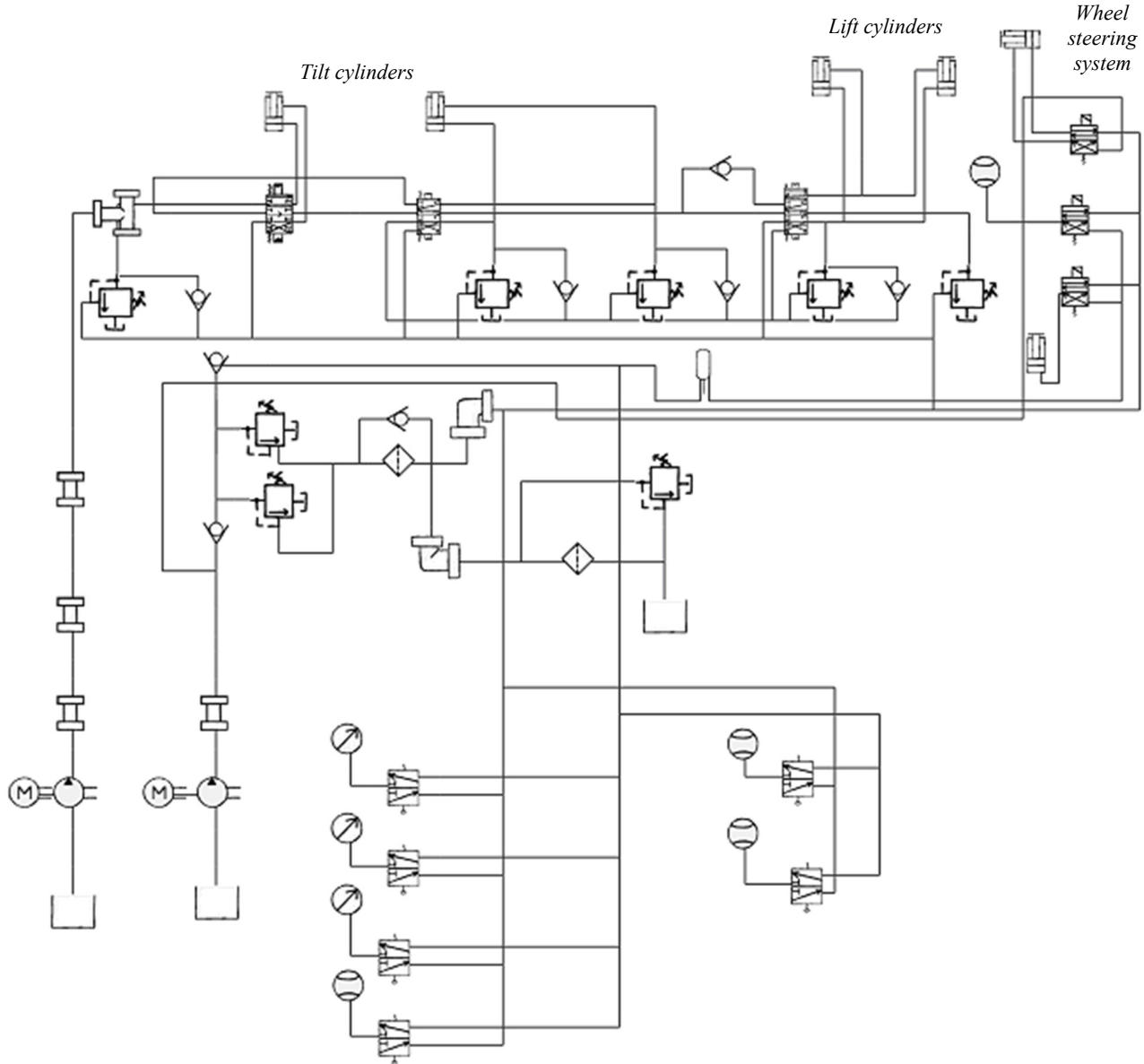

Figure 12: Wheel loader hydraulic circuit model with *Bondin* software.

The hydraulic circuit shares common elements of the system entrusted with producing the vertical movement of the bucket and the loading and unloading system, thereby allowing the corresponding actuators to function simultaneously and move in one direction or the other by means of the different distributor valves.

In addition to the elements mentioned throughout this paper, type R elements have been included in order to be able to simulate the secondary losses due to the fluid circulating through the pipe, the elbows and the joints.

$$R = 8 \cdot \rho \cdot \xi \cdot \frac{Q}{\pi^2 \cdot D^4} \qquad (4)$$

ξ : *Coefficient of secondary losses in each element.*
Q: *Flow.*

6. RESULTS

In the simulation, the pipe sections have been deemed to have a constant diameter of 8 cm and the engine moving the pumps to have linear acceleration during the first 0.5 seconds, the instant at which a constant speed of 350 rad/s is reached, corresponding to the supply of a 2 litre per second flow rate.

The simulation lasts for 20 seconds. In the first half of the time the cylinders will be moved in one direction and in the other half, in the opposite direction. It must be said that because the displacements of the lifting and tilting systems have different limits, the cylinders of both systems will not move in unison during the whole cycle.

Figures 13 and 14 show the movement of the cylinder responsible for producing the vertical movement of the bucket and the cylinder responsible for loading and unloading, respectively.
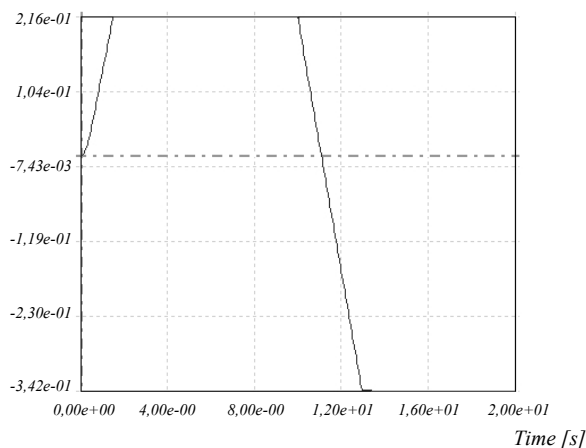
Cylinder displacement [m]

Figure 13: Tilting cylinder. displacement
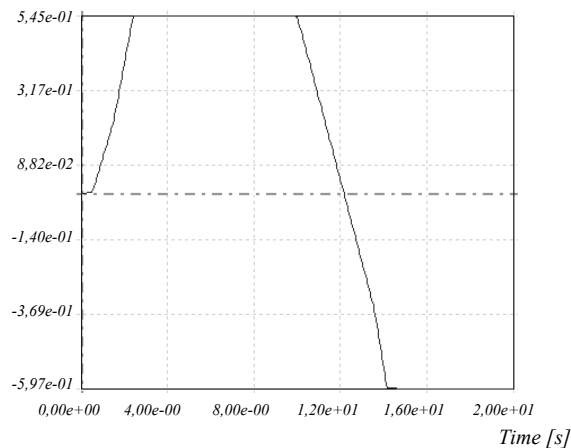
Cylinder displacement [m]

Figure 14: Lifting cylinder displacement.

In figure 13, it can be seen how the rod moves in a linear form practically from the beginning until it reaches the limit of its displacement, and then starts to move again when the direction of circulation of the fluid is reversed. Although the pump continues to supply flow, the cylinder stops in both cases, as when it reaches the limit of its displacement the pressure limiter valve comes into operation.

In figure 14, it can be seen how the rod also moves in a linear form practically from the beginning until it reaches the limits of its displacement. It can also be seen how the lifting cylinder movement undergoes a change in gradient approximately after the first second. This is because the other cylinder, the tilting cylinder, has concluded its displacement and therefore all the flow driven by the pump is used to move only the cylinder represented.

In order to better understand the previous two graphs, the speed of the tilting cylinder has been represented where it can be seen how the cylinder speed becomes constant almost from the very moment the cylinder begins to move in one direction or the other, and how the speed is zero when it reaches the end of its displacement (fig. 15).
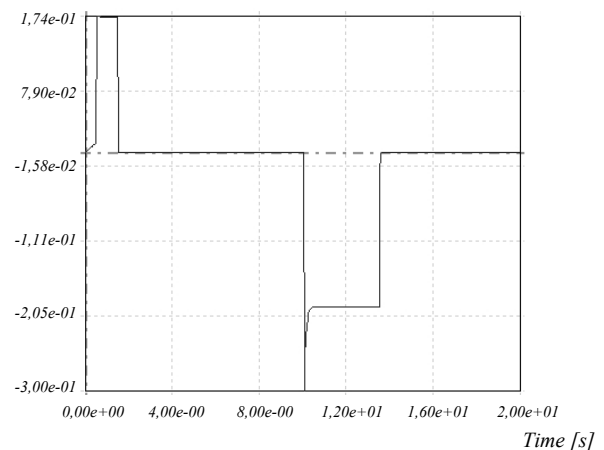
Cylinder speed [m]

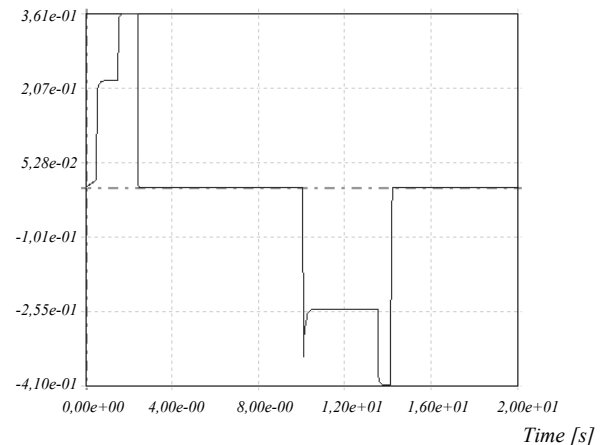Figure 15: Tilting cylinder rod speed.

Cylinder speed [m]

Figure 16: Lifting cylinder rod speed.

Likewise, figure 16 shows the lifting cylinder speed where it can be seen that when the tilting cylinder has stopped its speed accelerates because the fluid was previously distributed through both cylinders but is now only distributed through one.

When a complete simulation of all the hydraulic circuit has been performed the behaviour of the different cylinders can be analysed from the different results obtained. In this way an expression can be found to determine the speed of the different actuators as a function of the different control inputs (fig. 17).
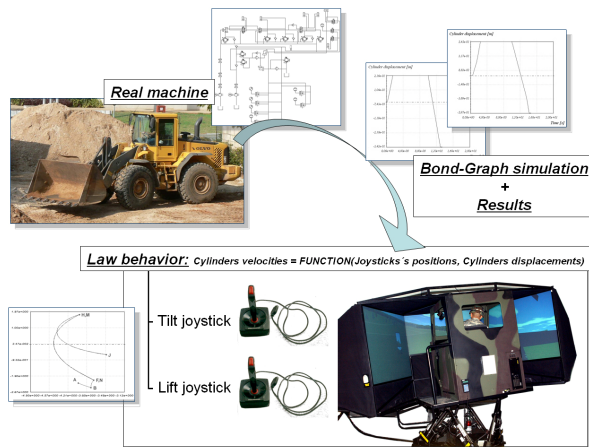

Figure 17: Outline of how behavioural law is obtained.

Therefore, instead of proceeding to perform a simulation of the full circuit it is enough at any instant to implement these expressions to obtain the movement of the bucket (Romero et al. 2006).

These expressions can be written simply as,

$$Velocity_{Tilt\_cylinder} = A \cdot Tilt_{Joystick\_value}$$
$$Velocity_{Lift\_cylinder} = B \cdot Lift_{Joystick\_value} \quad (5)$$

where A and B are constants that are dependent on the corresponding cylinder being out of the limits or not, and if the other cylinder is moving at the same time or not.

7. CONCLUSIONS

Various models of elements developed by the Bond Graph technique have been presented. The validity of these models is evident from the models themselves and the simulations performed. A complex model has also been developed by "encapsulating" various sub-models prepared beforehand. In conclusion, it may be said that in simulations that need a real time solution, a full simulation of the entire hydraulic circuit can be performed, and then the model can be simplified by just analysing the different results obtained. Next one or more expressions can be obtained to determine the speed of the different actuators as a function of the inputs.

8. REFERENCES

Borutzky, W. 1995. "Discontinuities in a Bond Graph Framework". *Journal of the Franklin Institute*, Elsevier Science Ltd, Vol. 332B, Nº. 2. pp. 141-154.

Borutzky, W. and Cellier, F. 1996. "Tearing in Bond Graphs with Dependent Storage Elements". *Proceedings. of CESA'96*, pp. 1113 - 1119

Bunes, O. and Einang, P.M. 2000. "Comparing the performance of the common rail fuel injection system with the traditional injection system using computer aided modelling and simulation". ENSUS 2000, *International Conference on Marine Science and Technology for Environmental Sustainability*. Newcastle.

Cobo, M. et al. 1998. "Modeling, Identification and real-time control of bucket hydraulic system for a Wheel type loader earth moving equipment". *Journal Mechatronics vol. 8*. pp 863-885.

Félez, J. et al. 1990. "Bondyn: A Bond Graph based simulation program". *Trans. ASME. J. Dyn. Syst. Measurement and Control*. Vol.112 , pp.717-727.

Félez, J. et al. 2000. "Deriving simulation models from bond graphs with algebraic loops. The extension to multibond graph systems". *Journal of the Franklin Institute*. Vol. 337 , pp.579-600.

Gawthrop, P.J. and Smith, L.S. 1992. "Causal augmentation of Bond Graphs with algebraic loops". *J. of the Franklin Institute*. Vol. 329, No.2, pp.291-303.

Karnopp, D.C. et al. 1990. "System Dynamics: A Unified Approach". John Wiley & Sons, Inc., Second edition.

Krishnaswamy, K. and Li, P.Y. 2006. "Bond Graph Based Approach to Passive Teleoperation of a Hydraulic Backhoe". Transactions of the ASME. *Journal of Dynamic Systems, Measurement and Control*. Vol. 128. pp 176-185.

Kun, D. et al. 1999. "Simulation Study on the Hydraulic Jacking system for LIAOHAI2 Platform". *Proceedings of the 3rd International Symposium on Fluid Power Transmission and Control (ISFP'99)*. Beijing. International Academic Publishers. pp 18-22.

Nadjm-Tehrani, S. and Strömberg, J.E. 1999. "Formal Verification of Dynamic Properties in an Aerospace Application". Journal *Formal Methods in System Design*, Springer Netherlands. Vol. 14. Nº2. pp 135-169.

Thétiot, R. and Zouaoui, F. 1998, "Automatic Construction of Processes from a Bond Graph Representation". *Qualitative Reasoning Group* research QR-98. Northwestern University. pp 131-136.

Romero, G. et al. 2005. "Optimised procedures for obtaining the symbolic equations of a dynamic system by using the Bond-Graph technique". *International Conference on Bond Graph Modeling and Simulation ICBGM'05*. New Orleans. SCS Publishing, Simulation Series. Vol.37, No.1, pp.51-58.

Romero, G. et al. 2005. "Systematic reduction of dynamic equations to a minimal set in systems modelled with Bond Graphs". *International Conference on Integrated Modeling and Analysis in Applied Control and Automation IMAACA'05*. Marseille. pp.59-65.

Romero, G. et al. 2006. "Kinematic analysis of mechanism by using Bond-graph language". European Conference on Modelling and Simulation ECMS 2006. Bonn . pp 155-165.

Wanli, L. and Fumin, Z. 2005. "Analysis on Dynamic Characteristic of a High-pressure Water Generator". *Sixth International Conference on Fluid Power Transmission and Control* (ICFP' 2005). Hangzhou.

Website of Hydro+Pneu software.
< http://www.pneuhydro.com/ >

Website of Software for Fluid Power Technology.
< http://journal.fluid.power.net/issue6/software6.html >

Website of Linköpings Universitet.
< http://www.flumes.ikp.liu.se/tools/hopsan/index_en.xf/ >

Website of Lab-Volt Systems, Inc.
< http://www.labvolt.com/display_left.cfm?view=Topics >

9. AUTHOR BIOGRAPHIES

GREGORIO ROMERO received his Mechanical Engineering and Doctoral degrees from the UNED. (Spain) in 2000. He got his PhD Degree from the Technical University of Madrid in Spain in 2005 working on simulation and virtual reality, optimizing equations systems. He has worked as Assistant Professor at the Technical University of Madrid in Spain (UPM) since 2001. He is developing his research in the field of simulation and virtual reality including simulation techniques based on bond graph methodology integrating computer graphics and virtual reality techniques to simulation in real time. He has published more than 35 technical papers and has been actively involved in over 20 research and development projects and different educational projects.

JESÚS FÉLEZ received his Mechanical Engineering and Doctoral degrees from the University of Zaragoza in 1985 and 1989. He started as Associate Professor at the Technical University of Madrid in Spain (UPM) in 1990 and became Full Professor in 1997. His main activities and research interests are mainly focused on the field of simulation, computer graphics and virtual reality. His research includes simulation techniques based on bond graph methodology integrating computer graphics and virtual reality techniques, mainly addressed towards the development of simulators. He has published over 50 technical papers and has been actively involved in over 25 research and development projects. He has served as thesis advisor for 30 master's theses and four doctoral dissertations.

M. LUISA MARTÍNEZ received her Mechanical Engineering and Doctoral degrees from the Technical University of Madrid (Spain) in 1990. She got her PhD Degree in 1997 working on variational geometry. In 1990 she started to work as Associate Professor at the Technical University of Madrid in Spain (UPM). Her thesis was focused on variational geometry. She usually works in the field of computer graphics, virtual reality and CAD. During this time she has been involved in different educational projects and pilot activities promoted by the European Commission and other Spanish institutions. She has published over 40 technical papers and has been actively involved in over 25 research and development projects.

JUAN JOSÉ DEL VAS received his Mechanical Engineering degree from the U.P.M. (Spain) in 2006. He has worked as researcher at the Technical University of Madrid in Spain (U.P.M.) since 2005. He developed his research in the field of simulation techniques of systems based on bond graph methodology and in environmental projects. He has been involved in others technical papers.

An XML Based Simulation Method for Extended Queuing Networks

Andreas W. Liehr, Klaus J. Buchenrieder
Institut für Technische Informatik
Universität der Bundeswehr München
D-85577 Neubiberg, Germany
Email: {andreas.liehr|klaus.buchenrieder}@unibw.de

KEYWORDS

Extended Queuing Network Models, Extensible Markup Language, Event Based Simulation Method

ABSTRACT

Extended Queuing Networks (EQN) are popular models for performance simulation of computer systems and communication networks. The EQN simulator developed in this work provides a simple and efficient simulation environment, which can be seamlessly integrated by XSLT transformations into modern system-design environments. Novel in this approach, is the consistent exertion of XML for all programmed modules of the simulator and the specification of the EQNs itself. The simulation method developed profits from XML Schema and provides a framework for constructing and solving standard Queuing Networks (QNM) as well as EQNs. The application of the XML Path Language (XPath) warrants a simple, robust and error resistant simulation, that can be interfaced with numerous programming languages with an XML interface. The paper presents the simulation method, the underlying XML model, model validation and the analysis of results. In the contribution, we also provide implementation details and experimental results for the prototype simulator.

INTRODUCTION

Since the invention and introduction of Extended Queuing Network Models by Charles Sauer (Sauer et al., 1980; Sauer and MacNair, 1983) numerous advances for specification, simulation and use of EQNs have been brought forward. Sinclair and Mandala provide a capacious list of EQN simulation tools and ascents in the field (Sinclair and Madala, 1986). A more recent approach, relies on Java for model description and adopts HLA-based distributed simulation (D'Ambrogio et al., 2006; Gianni and D'Ambrogio, 2007).

Multipresent research (Musovic et al., 2005; Gu and Petriu, 2005; Woodside et al., 2005; Balsamo et al., 2006) on performance simulation and power estimation, profits from QNMs and EQNs, even though commercial support for native EQN simulation is unavailable. The gap between proprietary representations within simulation tools and simulation models in XML can be bridged by theoretical methods developed by Xu, D'Ambrogio and Liehr (Xu and Lehmann, 2004; D'Ambrogio and Iazeolla, 2005; Liehr and Buchenrieder, 2007). These academic foundations lead us to develop an event driven EQN simulator, that works on XML representations of a simulation model. Through the design of our EQN description in XML and a dexterously designed XML Schema definition, the syntactical and semantical verification reduces to the validation of the XML representation against the XML Schema. This enables us to suspend the simulation process in every state by writing the current state to an XML file and to an overarching resume file of the simulation process. This resume constitutes the baseline for the independently conducted analysis.
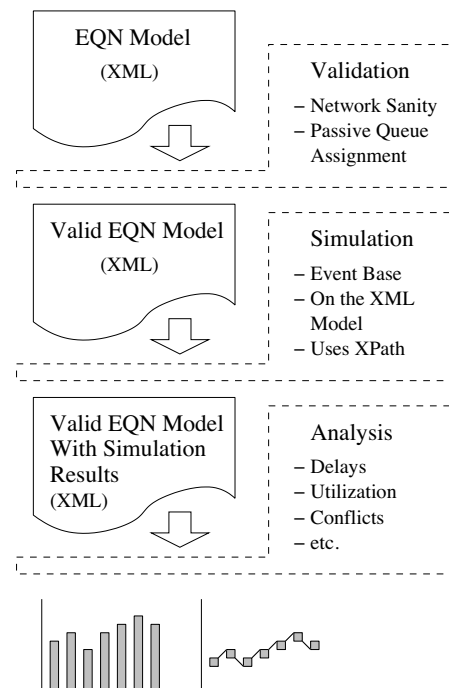

Figure 1: The XML Based EQN Simulator

The design of our event-driven simulation method splits the process into a validation phase, a simulation phase and an analysis phase as presented in Figure 1. While there is no need to define the desired simulation results prior to analysis, the structure of the gained simulation results foster a fast gathering of required informa-

tion, concerning arbitrary aspects (like average waiting time for jobs, overall throughput on distinct points of the EQN, etc.) of the simulation model.

The remainder of this paper is organized as follows: The following section introduces the syntax of our XML representation for EQNs and the benefits of the XML Schema. Section 3 introduces our simulation algorithm and in the following section we present details of the analysis phase. The concluding sections bring forward our prototype implementation and provide experimental results.

THE XML MODEL

The EQN structure used in our work, resembles the definition originally proposed by Sauer, McNair and Salza (Sauer et al., 1980; Sauer and MacNair, 1983). The networks are composed of the following network elements: a set of $Nodes$, token pools, variables and connections between the elements of $Nodes$.

The set $Nodes_{PQ}$ contains all elements of the network which belong to passive queues except the token pool itself:

$$Nodes_{PQ} = \{ALLOCATION\_NODES,$$
$$CREATE\_NODES, DESTROY\_NODES, \quad (1)$$
$$RELEASE\_NODES\}$$

The set $Nodes$ contains all elements through which jobs can be routed:

$$Nodes = \{Nodes_{PQ}, ACTIVE\_QUEUES,$$
$$ROUTING\_NODES, ROUTING\_POINTS, \quad (2)$$
$$SET\_NODES, SOURCES, SINKS\}$$

In contrast to the original definition, we apply minor modifications concerning ingoing and outgoing connections, to allow for a validation step. The semantics of the modified and the original version, however remain identical.

Elements in $Nodes$ are restricted to exactly one outgoing connection per node except for sinks and routing nodes. Paths can be split only at routing nodes, containing a list with an arbitrary number of outgoing routes and a rule specifying the routing of the arriving jobs. The rule decides which route each job will take.
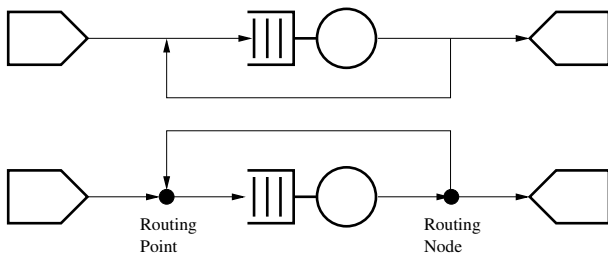

Figure 2: The Introduction of Routing Node and Routing Point

The set $Nodes_s$ denotes all elements with a successor, the jobs can be routed through. Because routing nodes have as many successors as routing decisions are possible at this point of network, the ROUTE elements of routing nodes, representing this possibilities, are included in this set:

$$Nodes_s = \{Nodes_{PQ}, ACTIVE\_QUEUES,$$
$$ROUTING\_POINTS, SET\_NODES, \quad (3)$$
$$SOURCES, ROUTING\_NODES/ROUTES\}$$

Furthermore, we introduce routing points to ensure that every element of $Nodes$ has exactly one incoming connection. Exceptions from this rule are made only for routing points merging a number of paths.

The set $Nodes_p$ contains all elements with a predecessor, through which jobs can be routed. Because routing points have as many predecessors as routes, the ROUTE elements of routing points are part of the set:

$$Nodes_p = \{Nodes_{PQ}, ACTIVE\_QUEUES,$$
$$ROUTING\_NODES, SET\_NODES, \quad (4)$$
$$SINKS, ROUTING\_POINTS/ROUTES\}$$

Figure 2 shows a traditional EQN and the semantically equivalent representation, using the modified notation, below. For simulation, it is stored in a single XML file with the top level element SIM_EQN. Listing 1 provides an example. It constitutes the initial state for the simulation of the EQN shown in Figure 3. The nodes of the EQN are stored as child elements of the element NETWORK.

Listing 1: The XML Notation for EQN Models

```
<SIM_EQN>
<NETWORK>
 <SOURCE Id='1' Rule='D:30' NId='2'>
  <JOB>
   <TOKENS/>
   <VARIABLES/>
  </JOB>
 </SOURCE>
 <ALLOCATION_NODE Id='2' NId='3' PQId='1' Rule
    ='D:1'/>
 <ACTIVE_QUEUE Id='3' NId="4" Rule='D:50'
    Scheduler='FCFS'/>
 <RELEASE_NODE Id='4' NId='5' PQId='1' Rule='
    D:1'/>
 <SINK Id='5'/>
</NETWORK>
<PASSIVE_QUEUES>
 <PASSIVE_QUEUE Id='1' Size='1' Tokens='1'
    Scheduler='FCFS'/>
</PASSIVE_QUEUES>
<VARIABLES/>
<GLOBAL_TIMER Time='0' Stop='10000'/>
<JOBS/>
<HISTORY/>
<NEXT_EVENTS>
  <EVENT Type="NEW_JOB" NId="1" Time="0"/>
</NEXT_EVENTS>
</SIM_EQN>
```

Each element of $Nodes_s$ has a unique identification attribute Id. A predecessor element from $Nodes_p$ points to its successors Id attribute with its attribute NId. Depending on the type, each node may contain additional

attributes or child elements such as: building rules, job templates, size information, scheduler type, etc. These attributes allow us to store the information of these nodes in our XML representation. As an example, consider the source in Listing 1: It contains a building rule and a template for jobs as a child element. Each node is assigned to a passive queue and contains the Id of the matching passive queue as attribute PQId.

A passive queue is stored as a child element named PASSIVE_QUEUE of the element PASSIVE_QUEUES. It is attributed with an Id, a maximal size, a scheduler type, the current number of tokens, and the token pool of the queue.
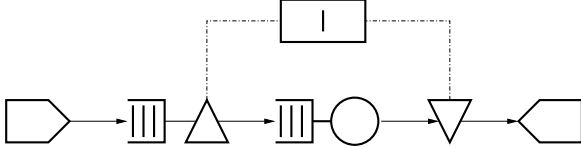

Figure 3: An EQN with one Active Queue and one Passive Queue

The element VARIABLES contains global variables as child elements. These may be required by routing nodes to perform complex routing decisions for the jobs.
The element GLOBAL_TIMER of the simulation model holds the current time stamp, followed by the time stop for the end of the simulation. Jobs generated by the source elements are stored inside the element JOBS. The HISTORY element provides a movement trace of jobs and the flow of tokens. The element NEXT_EVENTS contains information about pending creations and the finishing time for the servicing of jobs.

Properties of the Simulation Model
The model validation against the XML Schema guarantees three properties: (i) all elements of $Nodes_s$ from definition (3) have a predecessor and can be reached by jobs; (ii) every element of $Nodes_p$ from Definition (4) has exactly one successor node; (iii) every element of $Nodes_{PQ}$ is assigned to exactly one passive queue.
While (i) and (ii) are ensured by the network validation, (iii) is enforced by passive queue assignment validation.
Network Validation: The validation process ensures, that the connections between nodes in the network are well defined according to the definition of EQNs. This holds when the function, with $Nodes_p$ as source and $Nodes_s$ as target, is bijective. The mechanism involves two pairs of keys and key references in the XML Schema definition.
The key of the first pair, addresses the successor-definition within the network. It is constructed from the Id attributes of all elements in $Nodes_s$ (3) and thereby implies, that two elements of $Node_s$ can not have the same Id:

$$\forall m \in Nodes_s \; \nexists n \in Nodes_s :$$
$$m \neq n \wedge m.Id = n.Id \qquad (5)$$

The associated key reference is build from the NId attributes of all nodes having a successor, as defined in (4). This key reference ensures, that a successor for every element of $Nodes_p$ exists in $Nodes_s$:

$$\forall m \in Nodes_p \exists n \in Nodes_s : m.NId = n.Id \qquad (6)$$

Listing 2 shows an XML Schema definition for the first key/keyref pair.

Listing 2: An XML Schema Key Definition for EQN Nodes

```
<xsd:key name="networkKey">
 <xsd:selector xpath="./EQN:SINK | ./
     EQN:ACTIVE_QUEUE | ./EQN:ALLOCATION_NODE
     | ./EQN:RELEASE_NODE | ./EQN:DESTROY_NODE
     | ./EQN:CREATE_NODE | ./EQN:SET_NODE |
     ./EQN:ROUTING_NODE | ./EQN:ROUTING_POINT/
     EQN:ROUTES/EQN:ROUTE" />
 <xsd:field xpath="@Id" />
</xsd:key>
<xsd:keyref name="networkKeyRef" refer="
    EQN:networkKey">
 <xsd:selector xpath="./EQN:SOURCE | ./
     EQN:ACTIVE_QUEUE | ./EQN:ALLOCATION_NODE
     | ./EQN:RELEASE_NODE | ./EQN:DESTROY_NODE
     | ./EQN:CREATE_NODE | ./EQN:SET_NODE |
     ./EQN:ROUTING_NODE/EQN:ROUTES/EQN:ROUTE |
     ./EQN:ROUTING_POINT" />
 <xsd:field xpath="@NId" />
</xsd:keyref>
```

The second key is defined by the NId attributes of $Nodes_p$. It implies that two elements of $Nodes_p$ can not have the same successor attribute NId:

$$\forall m \in Nodes_p \; \nexists n \in Nodes_p :$$
$$m \neq n \wedge m.NId = n.NId \qquad (7)$$

The second reference key is defined by the same set of Ids as the first key. This warrants a predecessor in $Nodes_p$ for every element of $Nodes_s$:

$$\forall m \in Nodes_s \exists n \in Nodes_p : m.Id = n.NId \qquad (8)$$

From (6) and (7) follows, that the connections between nodes are a function f as defined in (9). Thereby, NId values of elements in $Nodes_p$ are the source X in function f. The Id values of elements in $Nodes_s$ are the target Y:

$$X = \{x | \exists m \in Nodes_p : x = m.NId\}$$
$$Y = \{y | \exists n \in Nodes_s : y = n.Id\} \qquad (9)$$
$$f : X \rightarrow Y, x \mapsto x := y$$

Definition (9), together with (5) imply:

$$\forall m \in Nodes_p \exists! n \in Nodes_s : m.NId = n.Id \qquad (10)$$

Consequently, f is surjective and the function f is injective as (8) and (9) imply. Hence, the combination of (8), (9) and (10) suggests f bijective.
The XML Schema allows only well formed EQNs according to the basic definition of EQNs. Nodes with an illegal number of predecessors, as shown in the left upper
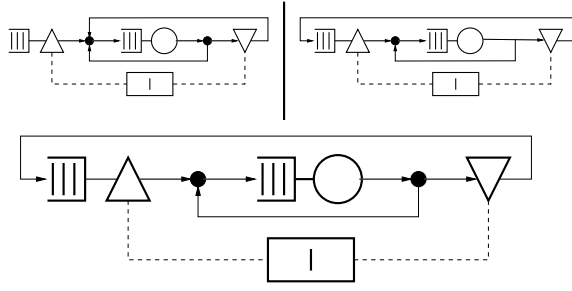
Figure 4: A Valid EQN and Two Invalid Modifications

part of Figure 4, will not be validated. The same holds for nodes with an illegal number of successors, as depicted in the upper right part of Figure 4.

Passive Queue Assignment Validation: A third key/keyref pair ensures, that every element of $Nodes_{PQ}$ is assigned to exactly one passive queue. The key is defined with the Id attributes of all passive queues.:

$$IDs = \{x | \exists m \in PASSIVE\_QUEUES : m.Id = x\} \quad (11)$$

Every passive queue has a unique Id:

$$\forall m \in PASSIVE\_QUEUES$$
$$\nexists n \in PASSIVE\_QUEUES : \quad (12)$$
$$m \neq n \wedge m.Id = n.Id$$

The key reference is defined by the PQId attribute of every element from $Nodes_{PQ}$.:

$$PQIDs = \{x | \exists m \in Nodes_{PQ} : m.PQId = x\} \quad (13)$$

From (12) and the definition of the key reference follows:

$$\forall m \in PQIDs \, \exists! \, n \in IDs : m.PQId = n.Id \quad (14)$$

This proves that every element of $Nodes_{PQ}$ is assigned to exactly one passive queue. Hence, the EQN constructions shown in Figure 5 are impossible.

While we only needed to write an appropriate XML Schema, the whole validation process is done by off the shelf XML validation tools. Because every part of the simulation result will be created by the XML manipulation methods, the result of the simulation is valid by construction, as long as the XML Schema is correct.
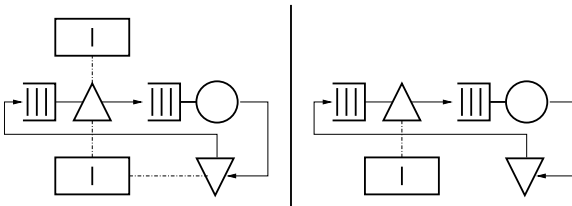

Figure 5: Two EQN with illegal assigned Passive Queues

THE SIMULATION PHASE

In the simulation phase, validated EQN models are executed on an event base. The simulation is synchronized with a global timer and extends from a current-time to the end-time prespecified in the XML representation. Since the current-time variable is advanced whenever a simulation event completes, the simulation process can be stopped and relaunched at any time. Hence, one can analyze partial simulations and continue thereafter.

The main_algorithm: For the explanation of the algorithm and improved readability we use Nassi-Schneidermann diagrams and XPath (Clark and DeRose, 1999) expressions, as illustrated in Figure 6.

First, the current time at the start of the simulation and the finishing time for the simulation are extracted from the XML representation. Afterwards a loop is initiated, that repeats until the time inside the simulation exceeds the predefined finishing time.

Inside this loop, the lists for triggered servers and passive queues are initialized. Then all events from NEW_EVENTS, that have reached their end-time, are extracted. Those events can be either from the type NEW_JOB or SERVED. When the type is NEW_JOB, a source releases the next job to the network. The method create_job (), explained below, is executed for this source as shown in the diagram.
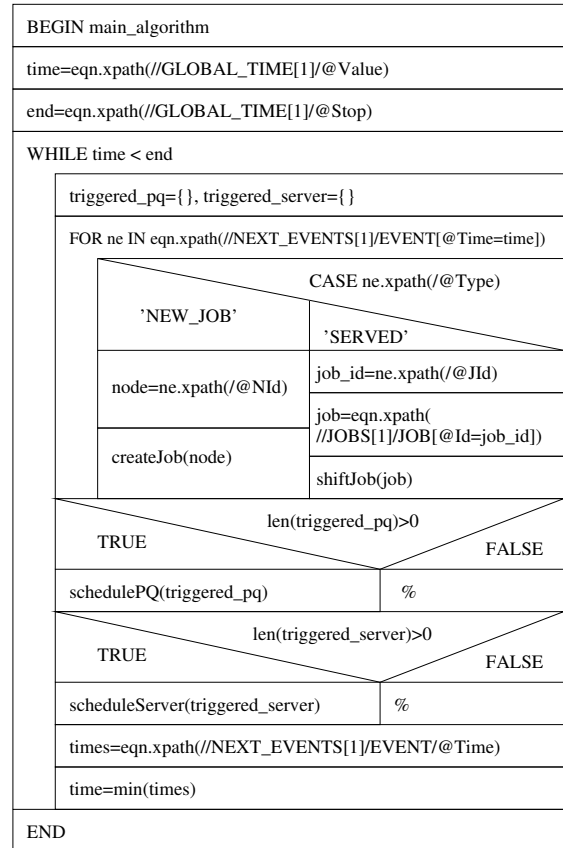

Figure 6: The Method main_algorithm

If an event of the type SERVED occurs, a server has finished the processing of the current job and shift_job () executes.

During the processing of jobs in the network, tokens are created or released into a token pool. In this case, the Id of the passive queue is appended to the variable schedule_pq and causes the activation of the appropriate scheduler. A scheduler of a queue, belonging to a server, is activated if the server finishes a job or jobs queue in front of an idle server.

The last XPath expression gathers the timestamps for all events that occur at a defined time in the network. Such events can be the generation of new jobs by sources and the end of the service time of a job at a server. The last line determines the minimum of these timestamps and sets the global time to this value before the loop is started again.

The method create_job: This method injects new jobs into the network when called by the main algorithm. It builds a copy of the job template provided by the source the job is created from. The copy is assigned to an XML element named PRESENT that represents the current state and position of this job. The PRESENT, representing the current time as start and stop time and the Id of the source as current position.

The time for job creation of this source is determined and an EVENT-element, holding this information, added to the list of next events. Last, the job is handed over to the shift_job () method.

Figure 7: The Method shift_job

The method shift_job: The method shift_job () implements the movement of jobs through the network, Figure 7. First, the successive node for the job and the type of this node is extracted by XPath. Second, a history item is created and appended to the simulation HISTORY. It contains the job Id, the node Id, the arrival-time and the leaving-time of the job on the current node.

Depending on the type of the successor node, a method containing a node specific algorithm is called. This modular structure enables the extension of the algorithm for new types of nodes and the selective modification of the

behavior of single node types without influencing the remainder of the simulation system.

THE ANALYSIS PHASE

During the simulation phase, the XML based EQN is autonomously simulated. In the analysis phase, the desired simulation results, such as waiting times, service times, resource utilization, throughput, etc. are specified by the user. Then, average, worst case, best case or complex interpretations of the demanded parameters are calculated.

In the following example, we derive the average waiting time for the jobs of the network shown in Figure 3.

Listing 3: Average Waiting Time for Jobs to be Served

```
items=eqn.xpath('/SIM_EQN/HISTORY/HI[@NId=3]')
sum=0
for item in items:
  sum+=item.xpath('@Stop')-item.xpath('@Start')
result=sum/len(items)
```

While the first XPath expression collects all entries in the history related to the allocation node, the for-loop sums up the total waiting time of its queue. The sum is then divided by the total number of jobs that were served by this server:

Listing 4: Number of Token Allocations from Passive Queue 1

```
items=eqn.xpath('/SIM_EQN/HISTORY/HI[@NId=2
    and @Tokens>0]')
result=len(items)
```

The code in Listing 4 produces the total number of allocation processes, connected to the passive queue with Id 1.

It is also possible to return to the simulation phase from the analysis phase. In this case only the attribute Stop of the element GLOBAL_TIMER must be altered. Partial simulations are carried out by specific time-intervals.

IMPLEMENTATION

The current version of the simulator is implemented in Python on a standard X86 architecture under Mac OSX. All XML manipulations are performed with the Python LXML library. For performance reasons, we split the XML model into several parts after import completes. The first part contains the network comprised of the static elements NETWORK and PASSIVE_QUEUES. The second part contains all active Jobs. The third part consists of the elements GLOBAL_TIMER, VARIABLES and NEXT_EVENTS. In each simulation step all elements belonging to HISTORY and jobs arriving at a sink are output. For analysis, module outputs are merged as soon as the simulation terminates or when interrupted by user interaction. This allows us to separate EQN creation, validation, simulation and analysis.

The current implementation enables the simulation of complex EQNs, as shown in Figure 8.

EXPERIMENTS

The EQN-based specification and simulation method was successfully tested with representative examples, such as the multiqueue network shown in Figure 8. It is composed of a CPU with a single-level cache, memory and a bus.
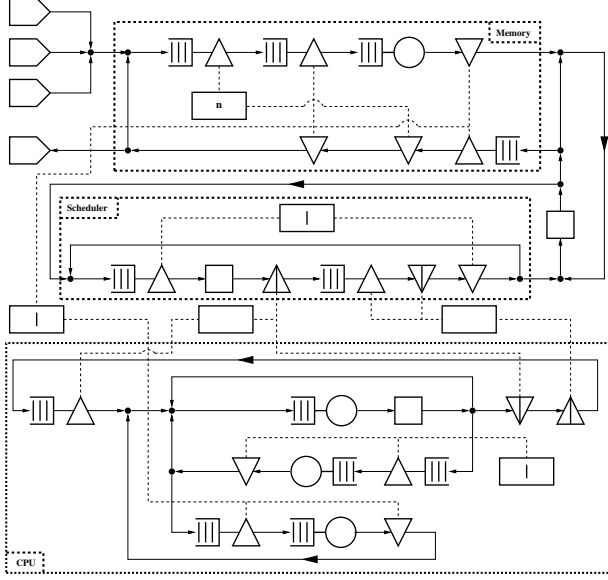

Figure 8: EQN Model for Experiments

When execution starts, three sources release jobs of different types into the system. Jobs from type one are compute intensive with little IO. Source one releases a type one job every 50 seconds into the system. IO intensive jobs of type two are generated and released at random time-intervals reaching from 1 to 10 seconds. Jobs from type three are very short but time-critical. Their generation interval spans from 5 to 50 seconds.

Numerous simulations for all active source combinations were carried out. In addition, different scheduling strategies for the CPU have been computed to demonstrate the simulator. In the analysis phase, we determined the average and the maximal time a job remained in the network. We also measured the average utilization of the system components during the simulated period.

Table 1: Execution Times and System Utilization for FCFS Scheduler

Sour-ces	⌀ 1 (s)	⊤ 1 (s)	⌀ 2 (s)	⊤ 2 (s)	⌀ 3 (s)	⊤ 3 (s)	Util. (%)
1	5.76	5.81	-	-	-	-	11.49
2	-	-	4.06	6.94	-	-	65.74
3	-	-	-	-	0.04	0.05	0.07
1;2	6.84	9.00	4.80	11.53	-	-	74.24
1;3	5.80	5.99	-	-	0.37	5.79	11.64
2;3	-	-	4.12	10.03	1.94	7.00	63.81
1;2;3	7.38	10.77	5.41	13.81	3.14	12.75	76.17

FCFS Scheduling: With FCFS as scheduling algorithm, jobs are served continuously in the order of arrival. As expected, the service time increases when multiple jobs await processing. Table 1 depicts the simulation results for the system with FCFS. The left column indicates the active sources. Column two contains the average execution time for type 1 jobs, column three holds the maximum execution time figured. The last column gives the average utilization of the system during the simulation interval. Clearly, the execution time for small jobs of type three increases as multiple sources are activated.

Round Robin Scheduling: As shown in Table 2, the average and maximal execution time for jobs from type three decreases noticeable for multiple active sources and Round Robin scheduling. As a result, a shorter time slot lowers the worst case execution time for jobs of this type even more.

Table 2: Execution Times and System Utilization for Round Robin Scheduler

Sour-ces	⌀ 1 (s)	⊤ 1 (s)	⌀ 2 (s)	⊤ 2 (s)	⌀ 3 (s)	⊤ 3 (s)	Util. (%)
Time Slot 20 ms							
1;2	14.89	31.80	4.62	9.03	-	-	74.83
1;3	5.73	5.85	-	-	0.04	0.67	11.49
2;3	-	-	4.23	11.51	0.13	0.49	61.42
1;2;3	13.19	26.24	4.38	10.33	0.12	0.53	68.0
Time Slot 100 ms							
1;2	15.20	29.06	4.39	10.34	-	-	72.08
1;3	5.77	5.93	-	-	0.05	0.16	11.57
2;3	-	-	4.63	10.16	0.51	1.82	65.34
1;2;3	18.30	50.01	5.56	13.74	0.57	2.37	77.76

Implementation Details: The EQN-based specification method and the simulator are implemented and executed with Python, version 2.5.2, on an Intel 2.2GHz Core 2 Duo CPU using a Mac OSX 10.5.2 operating system. While the first implementation with the PyXML package proved memory intensive and slow, we achieved a 13x speedup with an implementation based on the LXML Pythonic XML processing library in version 2.0apha6 and libxml2, version 2.6.21.

The maximal memory allocation during the simulation of the EQN of Figure 8 for $1.3x10^6$ job movements amounts less than 5 megabyte. The XML file, holding the simulation results, had a size of 56 megabytes. The simulation for this movement count required little over 27 minutes.

CONCLUSION AND OUTLOOK

In this paper, we propose a novel method to specify and simulate Extended Queuing Network Models entirely provided in XML. The method consists of a model validation, a simulation and an analysis phase. For all steps we use the power of XPath expressions to provide a simple and fault robust simulation. Model validation warrants that the presented EQN description fulfills the

properties, network sanity and singular passive queue assignment. In the analysis phase, we obtain system information from the simulation data. For excessive amounts of simulation data, due to model complexity , our method fosters distributed computation.

The methods and prototypes presented in this paper are highly appreciated by our industrial and scientific partners. The long-term goal of this research is to develop a system-specification environment in which the software, hardware and the system-architecture is captured in UML as presented in Liehr and Buchenrieder (2008). Based on this specification, a set of XML-described queuing networks is generated to allow for an efficient architectural exploration of the behavioral model. In this context, the EQN-simulator is an integral component for system-evaluation and performance-estimation.

Future work will include the enhancement of our model with the node types SPLIT_NODE, FISSION_NODE and FUSION_NODE to meet the full specification of EQNs. Furthermore, we intend to embed the simulator component into a web service that accepts XML based models and returns the simulation results. This will increase the usability of the approach in frameworks outside of the focus-domain of this research.
Another advancement persecuted is the implementation of a distributed multi-CPU method for the analysis of performance results.

REFERENCES

Balsamo, S., Marzolla, M., and Mirandola, R. (2006). Efficient performance models in component-based software engineering. In *SEAA '06 Proceedings of the 32nd Euromicro Conference on Software Engineering and Advanced Applications*, Cavtat/Dubrovnik, Croatia.

Clark, J. and DeRose, S. (1999). Xml path language (xpath) version 1.0. Technical report, World Wide Web Consortium.

D'Ambrogio, A., Gianni, D., and Iazeolla, G. (2006). jeqn a java-based language for the distributed simulation of queueing networks. In *Computer and Information Sciences - ISCIS 2006*, volume 4263/2006, pages 854–865, Istanbul, Turkey. Springer Berlin / Heidelberg.

D'Ambrogio, A. and Iazeolla, G. (2005). Design of xmi-based tools for building eqn models of software systems. In *23rd IASTED International Multiconference, Software Engeneering*, Innsbruck. ACTA Press.

Gianni, D. and D'Ambrogio, A. (2007). A language to enable distributed simulation of extended queueing networks. In *Journal of Computers*, volume 2.

Gu, G. P. and Petriu, D. C. (2005). From uml to lqn by xml algebra-based model transformations. In *WOSP '05: Proceedings of the 5th international workshop on software and performance*, pages 99–110, New York, NY, USA. ACM Press.

Liehr, A. W. and Buchenrieder, K. J. (2007). Generation of related performance simulation models at an early stage in the design cycle. In *14th IEEE International Conference on the Engeneering of Computer-Based Systems (ECBS)*, pages 7–14, Tucson, AZ, USA. IEEE Computer Society.

Liehr, A. W. and Buchenrieder, K. J. (2008). Performance evaluation of hw/sw-system alternatives. In *Design, Automation and Test in Europe (DATE) University Booth - Demonstration and Poster Exhibition*, Munich, Germany.

Musovic, A., Fischer, R., Nageldinger, U., Buchenrieder, K., and Lehmann, A. (2005). An eqn* based approach for high-level performance and power estimation. In *Conceptual Modeling and Simulation Conference, 2005. CMS 2005. 2nd 13M International Mediterranean Modeling Multiconference*, pages 193–198.

Sauer, C. H. and MacNair, E. A. (1983). *Simulation of Computer Communication Systems*. Prentice Hall.

Sauer, C. H., MacNair, E. A., and Salza, S. (1980). A language for extended queuing network models. *IBM Journal of Research and Development*, 24(6):747–755.

Sinclair, J. B. and Madala, S. (1986). A graphical interface for specification of extended queueing network models. In *ACM '86: Proceedings of 1986 ACM Fall joint computer conference*, pages 709–718, Los Alamitos, CA, USA. IEEE Computer Society Press.

Woodside, M., Petriu, D. C., Petriu, D. B., Shen, H., Israr, T., and Merseguer, J. (2005). Performance by unified model analysis (puma). In *WOSP '05: Proceedings of the 5th International Workshop on Software and Performance*, pages 1–12, New York, NY, USA. ACM Press.

Xu, Z. and Lehmann, A. (2004). Generating queuing network models from uml-based models for software performance prediction. In *SPECTS' 04: Proceedings of the 2004 International Symposium on Performance Evaluation of Computer and Telecommunication Systems*, San Jose, California, USA.

AUTHOR BIOGRAPHIES

ANDREAS W. LIEHR is a research assistant at the Universität der Bundeswehr München. He is currently working towards his doctoral degree in the field of performance prediction and power estimation in the development cycle of embedded systems.

PROF. KLAUS J. BUCHENRIEDER, PH. D. is professor at the Universität der Bundeswehr München, where he is head of the "Embedded Systems / Computers in Technical Systems" group.

MODELLING TUMOR-IMMUNE SYSTEMS WITH CELL-DEVS

Rhys Goldstein
Gabriel Wainer
Department of Systems and Computer Engineering
Carleton University
Ottawa, Ontario, Canada
Email: {rhys, gwainer} @sce.carleton.ca

KEYWORDS

DEVS formalism, Cell-DEVS, cellular automata, tumor growth, immune cells

ABSTRACT

Cell-DEVS, an extension of the DEVS formalism, is used to model tumor-immune systems that involve growing tumors interacting with immune cells. Tumors can be regarded as a core of necrotic cells, surrounded by dormant cells, surrounded in turn by proliferative cells. The growth of a tumor is effected by the division of its proliferative cells, but inhibited by nearby immune cells. We present a Cell-DEVS model representing a tumor-immune system. The use of Cell-DEVS is advantageous as it facilitates the formal specification and reuse of cellular models. The Cell-DEVS model was implemented and tested using the CD++ toolkit. Simulation results indicate that, in a qualitative sense, the desired behaviour of tumors and immune cells was captured.

INTRODUCTION

Tumor growth is a complex process; more complicated still when the response of an immune system to the tumor is considered as well. The combination of a growing tumor and the immune response will be referred to as a "tumor-immune system". Animal experiments have revealed many interesting aspects of tumor-immune systems. It has been observed, for example, that while inoculation with a certain number of tumor cells results in the rejection of the tumor by the immune system, a smaller number of tumor cells may lead to progressive tumor growth (Takayanagi et al., 2006). The modelling of tumor-immune systems is of interest as a means of investigating tumor growth and immunity.

Deterministic models based on differential equations have been designed for tumor-immune systems, but are regarded as inadequate for processes of considerable complexity. They are often restricted to 1-dimensional or radially symmetric tumor growth. Such limitations have been overcome by the adoption of stochastic cellular automata (Mallet and Pillis, 2006). In one case, a 3-dimensional cellular model was developed to simulate tumors growing over three orders of magnitude in radius (Kansal et al., 2002). Another cellular automaton has modelled the release of proteins by immune cells in response to a tumor (Takayanagi et al., 2006).

Facilitating the formal specification and reuse of cellular models, Cell-DEVS (Wainer and Giambiasi, 2002) is a compelling tool for the modelling and simulation of biological systems (Wainer et al., 2007). This paper demonstrates how the formalism can be applied, using a tumor-immune system model as an example. One such cellular automaton was chosen for this purpose (Huricha and Ruanxiaogang, 2003), which defines a 2-dimensional cell space through which immune cells wander in search of a tumor. Tumors generally form a core of necrotic cells, surrounded by a ring of dormant cells, surrounded in turn by a ring of proliferative cells. The tumor grows as the outermost proliferative cells divide. A growing tumor may overwhelm the immune system, or may be defeated by the immune cells. The Cell-DEVS model was implemented and tested using the CD++ toolkit (Wainer, 2002). Simulation results exhibited the desired qualitative behavior of tumors and immune cells.

After providing a brief overview of Cell-DEVS and a more detailed description of the model we based our work on, we describe the formal specification of a tumor-immune system model. Following this is a description of the CD++ implementation of the model. Several test results that demonstrate the behaviour of the model under simulation, showing how the Cell-DEVS formalism may aid in the modelling and simulation of biological systems such as these.

CELL-DEVS AND CD++

The DEVS formalism (Zeigler et al., 2000) provides a framework for the construction of hierarchical modular models, allowing for model reuse, and reducing development and testing times. Basic models, called "atomic models", are specified through transition functions. Multiple DEVS models can be integrated together to form hierarchical structural models, called "coupled models".

The Cell-DEVS formalism was defined as an extension to cellular automata (Wolfram, 2002) combined with DEVS. In Cell-DEVS, each cell in a cellular model is seen as a DEVS atomic model, and a procedure for coupling cells is defined based on the neighborhood relationship. Only the active cells in the cell space are triggered, independently from any activation period. Each cell of a Cell-DEVS model holds state variables and a local computing function, which updates the cell state by using its present state and its neighborhood.

A timed DEVS cell atomic model, associated with each cell in a cellular model, is specified as follows:

$$TDC = \langle X, Y, S, N, delay, d, \delta_{ext}, \delta_{int}, \tau, \lambda, D \rangle$$

The variable X defines the external inputs, Y defines the external outputs, and S is the cell state definition. The variable N represents the set of relative coordinates of each cell in the neighborhood. The $delay$ is the kind of delay used for the cell, and d is its duration. A $transport$ delay can be associated with each cell, which defers the outputs for the cell. A state change will be discarded if it is not steady during an $inertial$ delay. The local computing function τ is used to evaluate the future state of the cell. The remaining functions drive the cell's behavior: δ_{int} for internal transitions, δ_{ext} for external transitions, λ for outputs, and D for the state's duration.

A Cell-DEVS coupled model, representing an entire cell space, is specified as follows:

$$GCC = \langle X_{list}, Y_{list}, X, Y, n, [t_1, \ldots, t_n], N, C, B, Z \rangle$$

Here, X_{list} and Y_{list} are input/output coupling lists, used to define the model's interface. X and Y represent the input/output events. The n value defines the number of dimensions of the cell space, and $[t_1, \ldots, t_n]$ is the number of cells in each dimension. N is the neighborhood set. The cell space is defined by C, together with B, the set of border cells, and Z, the translation function.

CD++ (Wainer, 2002) is a modelling tool that implements the DEVS and Cell-DEVS formalisms. DEVS atomic models are programmed in C++, while both DEVS coupled models and Cell-DEVS models can be defined using a built-in specification language. CD++ makes use of the independence between modelling and simulation provided by DEVS, and different simulation engines have been defined for the platform.

TUMOR-IMMUNE SYSTEM MODEL

A cellular automaton, describing the growth of tumors interacting with an immune system, was previously designed and tested (Huricha and Ruanxiaogang, 2003). This original model, upon which the new Cell-DEVS version is based, consists of a 2-dimensional cell space. Each cell of the model represents a biological cell. The various types of cells are listed below.

- **Normal Cell**: a cell that is neither part of a tumor, nor a part of the immune system.

- **Immune Cell**: a cell that is part of the immune system. No distinction is made between the different types of immune cells that may respond to a tumor.

- **Proliferative Cell**: a tumor cell that divides, facilitating tumor growth.

- **Dormant Cell**: a tumor cell that was once proliferative, but no longer divides.

- **Necrotic Cell**: a tumor cell that was once dormant, but is now considered dead.

In a typical simulation, the initial cell space is composed primarily of normal cells. Dozens of immune cells are scattered throughout the cell space, and a single proliferative cell is situated in the center.

With the simulation underway, the single proliferative cell divides, converting adjacent normal cells into other proliferative cells. These converted cells then divide in turn, effecting the growth of the tumor. The innermost proliferative cells gradually become dormant cells, and the innermost dormant cells later become necrotic. The necrotic, dormant, and proliferative cells are expected to form three roughly concentric circles.

The immune cells wander randomly through the space of normal cells that surround the tumor. When an immune cell encounters a tumor cell, one of the two cells is destroyed. Such interactions may eventually lead to the death of the entire tumor.

CELL-DEVS MODEL SPECIFICATION

The tuple below is a Cell-DEVS coupled model that describes a tumor-immune system as a cell space.

$$TIS(t_1, t_2, d, p_{resisting}, p_{moving}, p_{curing},$$
$$p_{dividing}, p_{dying})$$
$$= \langle X_{list}, Y_{list}, X, Y, n, [t_1, t_2], N, C, B, Z \rangle$$

The parameters t_1 and t_2 are the dimensions of the cell space, and d is the delay for each cell. The parameters $p_{resisting}$, p_{moving}, and p_{curing} represent, respectively, the probabilities with which an immune cell is added to the model, moves from its present location, and cures a proliferative cell. Higher values of these probabilities favour the immune system. The parameter $p_{dividing}$ represents the probability with which a proliferative cell divides, and p_{dying} is the probability that a dormant cell becomes necrotic.

In our case, the model has neither inputs nor outputs, and is always 2-dimensional.

$$X_{list} = Y_{list} = X = Y = \varnothing$$

$$n = 2$$

The model uses a 5-by-5 cell extended Moore neighborhood, defined below.

$$N = \{[i, j] \mid i \in \{-2, -1, 0, 1, 2\}$$
$$\wedge \ j \in \{-2, -1, 0, 1, 2\}\}$$

The borders of the model are wrapped. This is convenient in that it ensures that immune cells incident on the border are not removed from the model completely. The disadvantage is that the model losses validity as the growing tumor nears the boundary.

$$B = \varnothing$$

The translation function Z is defined by the Cell-DEVS formalism, while each timed DEVS cell model is defined as follows.

$$C([i, j]) = \langle X, Y, S, N, delay, d, \delta_{ext}, \delta_{int}, \tau, \lambda, D \rangle$$

The input and output event sets X and Y are defined by the Cell-DEVS formalism.

The state associated with each individual cell consists of both a type and a direction. The type indicates whether the represented biological cell is a normal, immune, proliferative, dormant, or necrotic cell. The direction, which is either $[0, 0]$ or the coordinates of an adjacent cell, indicates either the way immune cells are moving or the way proliferative cells are dividing.

$$S = \{[type, direction] \mid type \in types$$
$$\wedge \ direction \in directions\}$$

$$types = \{normal, immune, prolif,$$
$$dormant, necro\}$$

$$directions = \{[i, j] \mid i \in \{-1, 0, 1\}$$
$$\wedge \ j \in \{-1, 0, 1\}\}$$

The model uses transport delays.

$$delay = transport$$

The variables δ_{int}, δ_{ext}, λ, and D are defined by the Cell-DEVS formalism. The following section describes the local computing function τ.

LOCAL COMPUTING FUNCTION

The term "state" refers to the value associated with each cell in the cell space. A state must be a value in the set S. A "state function" is a function that takes a pair of relative coordinates as an input, and results in the state of the cell associated with those coordinates. The state function s therefore maps values in the set N to values in the set S. The example equation below indicates that there is an dormant cell at a relative position of $[-2, 1]$.

$$s([-2, 1]) = [dormant, [0, 0]]$$

In any scope that contains the state function s, the functions $type$ and $direction$ will also be available. These functions provide the types and directions associated with neighboring cells. They are defined implicitly below.

$$s([i, j]) = [type([i, j]), direction([i, j])]$$

The local computing function τ maps the state function s, which carries the present states of neighboring cells, onto the new state of the cell with relative coordinates $[0, 0]$. It is defined below, using a conditional expression in which conditions are placed on the left of arrows that point to the corresponding results.

$$\tau(s) = \begin{pmatrix} type([0, 0]) = normal & \rightarrow \tau_{normal}(s) \\ type([0, 0]) = immune & \rightarrow \tau_{immune}(s) \\ type([0, 0]) = prolif & \rightarrow \tau_{prolif}(s) \\ type([0, 0]) = dormant & \rightarrow \tau_{dormant}(s) \\ type([0, 0]) = necro & \rightarrow \tau_{necro}(s) \end{pmatrix}$$

The local computing function depends on the functions τ_{normal}, τ_{immune}, τ_{prolif}, $\tau_{dormant}$, and τ_{necro}. Each of these were formally specified, and the definition of τ_{normal} is presented below as an example. Several other sets and functions are described first.

Two random value functions are used. The function $rand$ results in a value randomly selected from the argument set. The expression $rand(\{1, 2, 3, 4\})$, for example, would have a 25% chance of resulting in 2. The function $rand_U$ takes no arguments, and results in a real number randomly chosen between 0 and 1.

The variable adj represents the set of relative coordinates of all adjacent cells. In this specification, "adjacent cells" are the eight cells surrounding the one in question.

$$adj = directions - \{[0, 0]\}$$

The function $count_{near}$ results in the number of cells, with coordinates contained in the argument M, that are of type $type_{near}$. Among other things, it is used to determine whether there is a proliferative cell adjacent to an immune cell.

$$count_{near}(s, type_{near}, M)$$
$$= \sum_{[i,j] \in M} \left(\begin{array}{ll} type([i,j]) = type_{near} & \rightarrow 1 \\ type([i,j]) \neq type_{near} & \rightarrow 0 \end{array} \right)$$

The function $count_{incoming}$ results in the number of cells, with coordinates adjacent to $[k, l]$, that are of type $type_{incoming}$. In order to be counted, those adjacent cells must also be have a direction that points to $[k, l]$. Among other things, this function is used to prevent collisions between immune cells.

$$count_{incoming}(s, type_{incoming}, [k, l])$$
$$= \sum_{[i,j] \in L} \left(\begin{array}{ll} incoming([i,j]) & \rightarrow 1 \\ \neg incoming([i,j]) & \rightarrow 0 \end{array} \right)$$

$$incoming([i, j])$$
$$= (s([i, j]) = [type_{incoming}, [k - i, l - j]])$$

$$L = \{[k - i, l - j] \mid [i, j] \in adj\}$$

Demonstrating the $count_{incoming}$ function, Figure 1 shows a possible arrangement of types and directions of each cell in a neighborhood.

normal	normal	immune	normal	normal
[0, 0]	[0, 0]	[1, −1]	[0, 0]	[0, 0]
normal	normal	prolif	**normal**	immune
[0, 0]	[0, 0]	[1, 0]	**[0, 0]**	[−1, 0]
normal	normal	prolif	immune	normal
[0, 0]	[0, 0]	[0, −1]	[−1, −1]	[0, 0]
normal	normal	normal	normal	normal
[0, 0]	[0, 0]	[0, 0]	[0, 0]	[0, 0]
normal	normal	normal	normal	normal
[0, 0]	[0, 0]	[0, 0]	[0, 0]	[0, 0]

Figure 1: A hypothetical neighborhood of types and directions. Cell $[0, 0]$, at the center, is a proliferative cell pointing downwards. Cell $[1, 1]$ is indicated by boldface text.

There are three immune cells adjacent to the cell $[1, 1]$. The immune cell on the upper left and the one to the right are both approaching that cell. The one below, however, is pointing elsewhere. The cell $[1, 1]$ therefore has 2 incoming immune cells.

$$count_{incoming}(s, immune, [1, 1]) = 2$$

As specified above, the function τ_{normal} is used to evaluate the local computing function for a cell that is currently normal. A normal cell is considered to be "receiving" an immune cell if there is exactly one incoming immune cell. In this event, the cell becomes an immune cell that is "preparing" its next action. If a normal cell is not "receiving", then it is "praying" that is will not be overtaken by the tumor.

$$\tau_{normal}(s)$$
$$= \left(\begin{array}{ll} receiving(s) & \rightarrow \tau_{preparing}(s) \\ \neg receiving(s) & \rightarrow \tau_{praying}(s) \end{array} \right)$$

$$receiving(s)$$
$$= (count_{incoming}(s, immune, [0, 0]) = 1)$$

If a "praying" cell has any incoming proliferative cells, the previously normal cell is considered to be "mutating". It then becomes a proliferative cell itself, "plotting" its next division. A "praying" cell that is not "mutating" is "waiting".

$$\tau_{praying}(s)$$
$$= \left(\begin{array}{ll} mutating(s) & \rightarrow \tau_{plotting}(s) \\ \neg mutating(s) & \rightarrow \tau_{waiting}(s) \end{array} \right)$$

$$mutating(s)$$
$$= (count_{incoming}(s, prolif, [0, 0]) > 0)$$

An option to replenish immune cells was added to the Cell-DEVS version of the model. A "waiting" cell may

spontaneously become an immune cell. This happens with a probability of $p_{resisting}$, but only if the cell is surrounded by normal cells. Otherwise, the "waiting" cell remains a normal cell.

$$\tau_{praying}(s) = \begin{pmatrix} resisting(s) & \rightarrow \tau_{preparing}(s) \\ \neg resisting(s) & \rightarrow [normal, [0, 0]] \end{pmatrix}$$

$$resisting(s) = ((count_{near}(s, normal, adj) = 8) \\ \wedge (rand_U() < p_{resisting}))$$

Note that if $p_{resisting}$ is zero, immune cells will never spontaneously appear. Instead they will die off until there are none left, or until there are no proliferative cells left that can kill them.

The remainder of the local computing function was also specified in this manner. The formulae above, for example, indicate situations in which a normal cell receives an incoming immune cell. Similar logic is used to determine when an immune cell vacates its present location, leaving a normal cell in its place.

In the original cellular automaton, the radius of a tumor was restricted by a constant parameter. This parameter was omitted from the Cell-DEVS version to allow the interaction of the tumor and immune cells to determine the extent to which the tumor grows. In the Cell-DEVS model, immune cells only interact with proliferative cells. If they manage to destroy all proliferative cells, a cluster of necrotic cells may remain. This scenario is interpreted as an immune system victory. The alternative possibility is that the tumor reaches the boundary of the cell space. This result is interpreted as a tumor victory.

CD++ IMPLEMENTATION

The specified model was implemented using the CD++ toolkit, a task that involved re-writing the mathematical formulae in the built-in specification language. The $\tau_{wandering}$ formula, which determines the direction of an immune cell, provides an example of this translation.

$$\tau_{wandering}(s) = \begin{pmatrix} moving(s) & \rightarrow [immune, rand(adj)] \\ \neg moving(s) & \rightarrow [immune, [0, 0]] \end{pmatrix}$$

$$moving(s) = (rand_U() < p_{moving})$$

Below is a CD++ expression corresponding to the right-hand side of the definition of $\tau_{wandering}$.

```
if(
  ((uniform(0,1)) < #Macro(p_moving)),
  1 + ((trunc(uniform(0,8))+1)*0.0625),
  1)
```

The condition $moving(s)$ is captured by the sub-expression on the second line. If this condition is true, the value of the entire expression is taken from the third line. The integer 1 indicates the $immune$ cell type, while the random multiple of 0.0625 represents the random direction. If $moving(s)$ is false, a stationary immune cell results from the fourth line.

For testing purposes, a parameter was introduced to approximate the initial density of immune cells. Each cell was given a probability $p_{initial}$ of starting as an immune cell instead of a normal cell. This feature was defined in CD++ by the rule below. The `if` expression chooses between the immune cell type with value 1, and the normal cell type with value 0. The next line specifies the duration d of the delay. As indicated by the last line, the transition takes effect only at the beginning of a simulation, when all cells have been initialized with the value -1.

```
rule :
  { if(
      uniform(0, 1) < #Macro(p_initial),
      1,
      0) }
  #Macro(d)
  { (0,0) = -1 }
```

SIMULATION RESULTS

The four simulation results presented below demonstrate the uninhibited growth of a tumor, the movement of immune cells, the victory of a tumor over the immune system, and the victory of the immune system over a tumor. All tests were performed on cell spaces of 41-by-41 cells.

The first test investigates the growth of a tumor in the absence of immune cells. As shown in Figure 2, the simulation exhibited a growing tumor roughly shaped as three concentric circles.

The images in Figure 3 show the results of a simulation of immune cell movement. A p_{curing} value of 1 ensured that immune cells were never killed by proliferative cells, while a $p_{resisting}$ value of 0 ensured that immune were never added to the cell space. As there were 21 immune cells at the beginning and end of the simulation, the test served to partially verify that the issue of collision detection was adequately addressed.

(a) $time = 11$ (b) $time = 23$
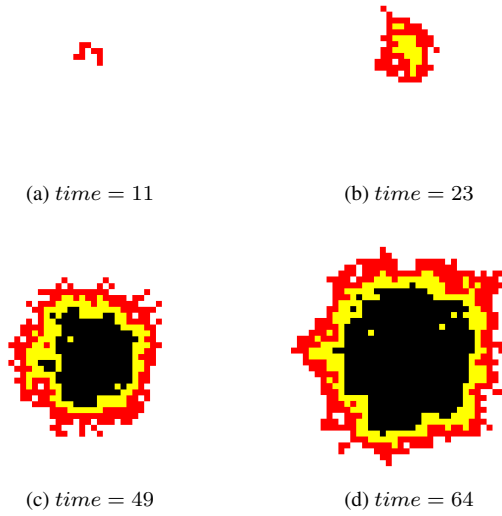

(c) $time = 49$ (d) $time = 64$

Figure 2: A simulation of tumor growth uninhibited by the immune system. In (a), the tumor consists of a small number of proliferative cells. In (b), the tumor has developed an inner cluster of dormant cells. These dormant cells, as shown in (c) and (d), later become an inner ring around a core of necrotic cells. The parameters were as follows: $p_{initial} = p_{resisting} = 0$, $p_{dividing} = p_{dying} = 0.5$.
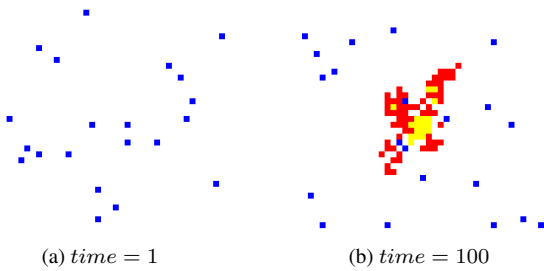

(a) $time = 1$ (b) $time = 100$

Figure 3: A simulation of immune cell movement. The image (a) shows the initial distribution of immune cells, and (b) shows a cell space with the same number of immune cells after 100 time units. These results were obtained with the following parameters: $p_{initial} = 0.01$, $p_{resisting} = 0$, $p_{moving} = p_{curing} = 1$, $p_{dividing} = 0.2$, $p_{dying} = 0$.

The final two tests demonstrate the interaction between growing tumors and immune cells. Figure 4 shows the results of a test in which the immune cells were defeated by a growing tumor. In the simulation presented in Figure 5, the parameters were changed such that the tumor was defeated by the immune system.

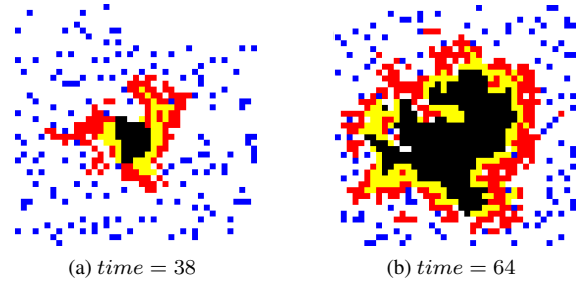(a) $time = 38$ (b) $time = 64$

Figure 4: A simulation in which a tumor overwhelms the immune system. In (a), the immune cells have cleared away a large section of proliferative cells on the upper side of the tumor. Later, in (b) the tumor is shown having regained this region. It thereafter proceeded to expand towards the boundaries of the cell space. The parameters for this test were chosen to favour the tumor: $p_{initial} = 0$, $p_{resisting} = 0.01$, $p_{moving} = 0.5$, $p_{curing} = 0.8$, $p_{dividing} = 0.6$, $p_{dying} = 0.6$.
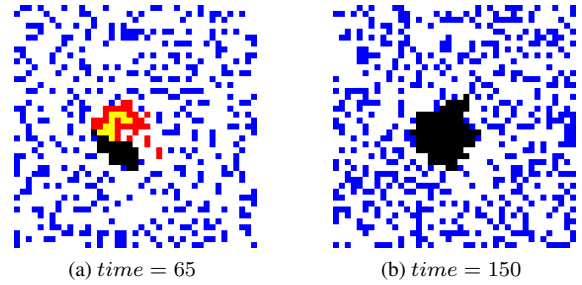

(a) $time = 65$ (b) $time = 150$

Figure 5: A simulation in which a tumor is defeated by immune cells. In (a), the proliferative cells were restricted to a single side of the otherwise dead tumor. The mass of necrotic cells increased roughly four times thereafter, but as shown in (b), the last proliferative cell was eventually destroyed. The parameters for this test were as follows: $p_{initial} = 0$, $p_{resisting} = 0.03$, $p_{moving} = 0.4$, $p_{curing} = 0.8$, $p_{dividing} = 0.4$, $p_{dying} = 0.8$.

CONCLUSION

A tumor-immune system model was specified using the Cell-DEVS formalism and implemented with CD++. Simulation results indicated that the model captured the intended qualitative aspects of tumor growth and immune system response.

Although similar simulations have previously been developed without Cell-DEVS, the use of this formalism was advantageous in that it facilitated the complete formal specification of the model. It was not necessary to explicitly indicate the advancement of time, nor the application of the local computing function to each cell in the cell space. These rules were part of the

simulation, not the model, and were therefore implicit in the formalism itself.

Another advantage of DEVS was not demonstrated in the paper, but is worth noting. Any two DEVS or Cell-DEVS models can be integrated by the specification of a link between the output of one model and the input of the other. Suppose, for example, that the Cell-DEVS model of the tumor-immune system was re-designed with the ability to accept immune cells as input values. Other DEVS models, perhaps representing blood vessels or lymph nodes, could be defined to output immune cells. All the models could be then be integrated, with the lymph nodes and blood vessels supplying immune cells to the site of a tumor.

Though inherently complex, biological systems can often be regarded as sets of interacting subsystems. The DEVS and Cell-DEVS formalisms offer a compelling approach to the modelling and simulation of such systems.

REFERENCES

Huricha, R. and Ruanxiaogang, X. (2003). A simple cellular automaton model for tumor-immunity system. *Proceedings. 2003 IEEE International Conference on Robotics, Intelligent Systems and Signal Processing*.

Kansal, A. R., Torquato, S., Harsh, G. R., Chiocca, E. A., and Deisboeck, T. S. (2002). Simulated Brain Tumor Growth Dynamics Using a Three-Dimensional Cellular Automaton. *Journal of Theoretical Biology*, 203(4):367–382.

Mallet, D. G. and Pillis, L. G. D. (2006). A cellular automaton model of tumor-immune system interactions. *Journal of Theoretical Biology*, 239:334–350.

Takayanagi, T., Kawamura, H., and Ohuchi, A. (2006). Cellular Automaton Model of a Tumor Tissue Consisting of Tumor Cells, Cytotoxic T Lymphocytes (CTLs), and Cytokine Produced by CTLs. *IPSJ Digital Courier*, 2:138–144.

Wainer, G. (2002). CD++: a toolkit to develop DEVS models. *Software, Practice and Experience*, 32(3):1261–1306.

Wainer, G. and Giambiasi, N. (2002). N-dimensional Cell-DEVS models. *Discrete Event Systems: Theory and Applications*, 12(1):135–157.

Wainer, G., Jafer, S., Al-Aubidy, B., Dias, A., Bain, R., Dumontier, M., and Cheetham, J. (2007). Advanced DEVS models with application to biomedicine. *Artificial Intelligence, Simulation and Planning, Buenos Aires, Argentina*.

Wolfram, S. (2002). *A New Kind of Science*. Wolfram Media.

Zeigler, B., Kim, T., and Praehofer, H. (2000). *Theory of Modeling and Simulation: Integrating Discrete Event and Continuous Complex Dynamic Systems*. Academic Press.

AUTHOR BIOGRAPHIES

Rhys K. Goldstein received a B.A.Sc. (2003) degree from the Engineering Physics Department at the University of British Columbia (Vancouver, BC, Canada). He then worked in the mining industry, developing data processing software and conducting geophysical surveys in Alaska, Tasmania, Cyprus, and other parts of the world. He is now working towards a M.A.Sc. in Biomedical Engineering from the Department of Systems and Computer Engineering at Carleton University (Ottawa, ON, Canada). His email is `rhys@sce.carleton.ca`.

Gabriel A. Wainer received the M.Sc. (1993) and Ph.D. degrees (1998, with highest honors) of the University of Buenos Aires, Argentina, and Université dAix-Marseille III, France. In July 2000, he joined the Department of Systems and Computer Engineering, Carleton University (Ottawa, ON, Canada), where he is now an Associate Professor. He has held positions at the Computer Science Department of the University of Buenos Aires, and visiting positions in numerous places, including the University of Arizona, LSIS (CNRS), University of Nice and INRIA Sophia-Antipolis (France). He is author of three books and over 160 research articles, and helped organizing over 70 conferences. He is Associate Editor of the Transactions of the SCS, and the International Journal of Simulation and Process Modeling. He was a member of the Board of Directors of the SCS, a chairman of the DEVS standardization study group (SISO). He is Director of the Ottawa Center of The McLeod Institute of Simulation Sciences and chair of the Ottawa M&SNet, and one of the investigators in the Carleton University Centre for Advanced Studies in Visualization and Simulation (V-Sim). His current research interests are related with modelling methodologies and tools, parallel/distributed simulation and real-time systems. His e-mail and web addresses are `gwainer@sce.carleton.ca` and `www.sce.carleton.ca/faculty/wainer`.

Finance and Economics

MICROSIMULATION MODELLING OF WORK-RELATED EXPENSES:
A NEW APPROACH

Gerhard Wagenhals and Jürgen H. Buck
Department of Statistics and Econometrics
University of Hohenheim
D-70593 Stuttgart, Germany
Email: wagenhls@uni-hohenheim.de

KEYWORDS

Tax Microsimulation, Work-related expenses, Commuting Allowance

ABSTRACT

Based on a representative data set of more than 36,000 original tax report extracts, we develop a microsimulation model that calculates the fiscal impact of changes in the area of tax deductible work-related expenses, in particular expenses for travelling from private home to the place of work, in Germany. The new model allows a more accurate estimation of the fiscal impact of changes in the German income tax law than previously possible and thus facilitates an informed evaluation of tax policy options.

INTRODUCTION

Political Background

The tax deductibility of commuting expenses is a hotly debated topic on the German economic policy agenda. In principle, it was abolished on January 1st 2007. Since then, German courts of law have questioned the constitutionality of this abolishment. Political parties and lobby groups came forward with many proposals on how to reestablish or reorganize the tax deductibility of commuting expenses.

Typically, these discussions rely on vague projections of the effects on tax revenues. Estimates of the grossed up revenues due to that tax concession have varied widely. For example, referring to 2005, the European Environment Agency estimates foregone revenues due to the commuting allowance to be less than 4.8 billion € (see European Environment Agency, 2007), whereas the Institute for European Environmental Policy quotes more than 5.8 billion € for the same period (see IEEP et al, 2007). This huge difference is far from satisfactory. Rational public policy must rely on precise figures.

Our model presents a first interpersonally verifiable way to assess the quantitative impact of any reform of the deductibility of work-related expenses, especially the commuting allowance, not only in terms of aggregate revenue effects but also in terms of the number of "winners" and "losers". The German Ministry of Finance uses the model on a regular basis. Its results have been published nation-wide (see e.g. Franz, 2007; Schrinner, 2007a,b).

Legal Background

In Germany, work-related expenses are tax deductible (see § 9 of the German Income Tax Law (EStG)). There is a 920 € lump-sum allowance per gainfully-employed person (§ 9 a EStG). When the taxpayer proves that his or her expenses are higher, they are fully deductible.

As from the assessment period 2001, a distance allowance for commuting between the private home and place of work was introduced in the Income Tax Law (see § 9 Section 2 EStG). From 2001 to 2003, the distance allowance amounted to 0.36 € for the first ten kilometers, and 0.41 € for more than ten kilometers up. From 2004 to 2006, there existed a uniform distance allowance of 0,30 € per kilometer.

With effect of January 1st, 2007, the tax deductibility of commuting expenses was abolished. There is a special hardship compensation, however. Deductions still can be made from the 21st kilometer of travel to the place of work: For any means of transportation and for the shortest distance between private home and place of work a commuter can deduct 0.30 € per kilometer for any actual working day, with a maximum of 4,500 € per fiscal year. The 4,500 € limit does not apply to claims for commuter expenses if the taxpayer uses an automobile which he or she owns or has the right to use or if he or she provides documentary proof that costs of public transport are higher than 4,500 €. There are special written evidence rules for these claims that are explained in § 9 Section 2 EStG and the references quoted there.

Why use the Model?

Up to now, "official" estimates of the effects on total tax revenue of any amendments of § 9 Section 2 EStG have been calculated by the Federal Ministry of Finance according to the Income Tax Law and directives, administrative practice and judicature applicable. Estimation heavily relied on expert knowledge and operating experience. The validity of previous revenue forecasts was reduced by the fact that persons in dependent employment very often do not give full particulars of their commuting expenses if the sum of commuting and other work-related tax-deductible expenses is below the general lump-sum

allowance per gainfully-employed person (i.e. currently 920 €). They often report their work-related commuting expenses to the tax authorities only if the sum of commuting and other work-related tax-deductible expenses is greater than this lump-sum allowance. Our new microsimulation model for work-related expenses resolves this problem using modern statistical techniques.

What he Model does

The aim of this microsimulation model is to facilitate an informed evaluation of policy options regarding the tax-deductibility of work-related expenses with special regard to commuting expenses. For example, this model helps policy makers in calculating to what extent changes of the commuting allowance can be financed through a reduction in the lump-sum allowance per gainfully-employed person.

How the Model works

For speed and simplicity, the model is split into two entirely different programs.

First, a data generating program takes data from the Statistical Office and generates the sub-sample that is needed by the model. The data generating program also performs the matching procedures necessary to model tax deductible expenses such as changes in the commuting allowance. This initial processing of the data is computationally very demanding. Once the base data set has been generated, these tedious calculations have not to be repeated again each time a new policy option is modelled. Furthermore, the resulting model data set to be delivered to policy makers is much smaller that the original data sets.

Second, the simulation program proper includes a routine that calculates aggregate tax revenues (wage tax, income tax and solidarity surcharge) as well as the total number of "winners" and "losers" of the reform. The program allows the user to change the parameters of the rules governing the deductibility of work-expenses interactively at run time. The results are incorporated in an easily manageable and intuitive spreadsheet.

Policy makers using the model will start off by setting basic parameters, such as:

- the levels of the commuting allowance for different distances between private home and place of work,

- the lump-sum allowance per gainfully-employed person, and

- the upper limit for users of public transport.

At any time, the user may set and change any of these basic parameters. Once data have been entered and parameters set, the user can start changing the parameters to examine the effects on total revenue and the number of "winners" and "losers". As the parameters are changed, policy makers instantly see the impact of changes.

What the Model does not do

Tax reforms typically are designed in the light of broader concerns such as creating work-incentives and fostering long-run economic growth. A model that would take into account these concerns would be quite an ambitious project and is well beyond the scope of the present model.

Summing up

This microsimulation model provides policymakers with a simple way of calculating the budgetary and welfare implications of alternative reforms of treating work-related expenses, particularly commuting expenses. It is then up to policymakers to decide on the reform scenario that is the most feasible in the particular context. The model is a flexible, easily manageable tool, not a substitute for good policy-making.

The next section describes the data generation process, then we turn to the setting up of the model, and finally we describe the results of an evaluation of some recent tax reform policy proposals.

DATA

Introduction

The primary data set used in this study was put at out disposal by the German Ministry of Finance. It is based on a special evaluation ("Sonderauswertung") of the income tax statistics in 2002. Sampling was carried out by the German Federal Statistical Office based on records not disclosed to the public.

Unfortunately, official income tax data lack important information necessary to model work-related expenses. Most importantly, official statistics neither provide complete information on work-related expenses submitted by documentary proof nor on the distance between private home and place of work. This information can be derived from our secondary data set, the German Socio-Economic Panel (GSOEP). The GSOEP is an ongoing representative longitudinal study of private households and individuals living in the Federal Republic of Germany (see e.g. Wagner et al., 2007).

So, to overcome the lack of information in German official income tax statistics, we combined information from two different sources. As the two samples do not observe the same set of units, neither merging nor record linkage techniques can be used. Instead, we applied appropriate statistical matching methods described in Buck (2006) or Wagenhals and Buck (2007).

As a result, we obtained a representative sample of more than 36,000 records. Each record represents a precisely defined number of taxation cases. Based on this representative sample we can assess the impact of tax reforms on the population of all taxation cases and thus on the resident population of Germany.

Sampling Design

We start with a stratified ten per cent random sample of all taxation cases in Germany in 2002. This sample

was drawn for our project by the German Statistical Office according to their standard procedures. Stratification variable is the aggregate income ("Gesamtbetrag der Einkünfte"). Stratified sampling ensures that the sample reflects the population of tax units with respect to this variable. The sample is drawn such that high income earners have a higher sampling probability, because high income classes are sparsely populated. Weighting factors make sure that this sample is representative. Based on this sample we draw a simple random sample.

We start with the hypothesis that we can do without a second stratification because of the high number of sample observations. In order to control this hypothesis we drew 50 additional samples and compared the empirical distributions (notably arithmetic mean and quantiles) of important variables like taxable income and marginal tax rates.

The differences between these random samples and the original 10 per cent sample are very small. Detailed results are available upon request.

Imputation of Work-related Expenses

When filing an income tax return, documentary proof for work-related expenses is only required if they exceed the 920 € lump-sum allowance. We assume that filers with work-related expenses well below this limit do not bother to itemize all positions (e.g. specialist literature, work clothes or contributions to professional organizations). To be exact, we assume that taxpayers do without itemized documentary proof if total work-related expenses are below 920 € and if the sum of itemized deductions with documentary proof amounts to less than 200 €.

This means that we include all taxpayers that do not submit documentary material with respect to their work-related expenses. We account for taxpayers who itemize some but not all of their work-related expenses in the tax declaration. They may only list items that are easy to produce such as subscriptions to newspapers or journals without specifying *all* work-related expenses in detail. We chose the limit of 200 € to account for these "lazy" taxpayers in our data fusion.

In our procedure to match supplement records to our primary data set, we supplement itemized work-related expenses of such taxation cases with itemized work-related expenses of taxation cases with work-related expenses exceeding the 920 € lump-sum allowance per gainfully-employed person. The premise behind this approach is that taxpayers with work-related expenses above this overall allowance provide evidence for all expenses eligible for inclusion as work-related expenses because it reduces their tax burden. Our matching approach allows for taxation cases with work-related expenses above the overall allowance with proved evidence of at most 400 €. The reason for that is the fact that on average some 25 to 30 per cent of the total work-related expenses are itemized expenses, the rest is commuting allowance. Therefore it can be assumed that filers with itemized work-related expenses of more than 400 € typically will also claim the commuting allowance because in general they may reckon that the sum of all work-related expenses will be above the overall allowance. In cases of joint filing (marital income splitting) we modify this procedure to allow for husbands or wives without gross income. Spouses not gainfully employed or in marginal employment were removed by considering in our matching procedure only persons with gross incomes of more than 1,000 € per year.

Imputation of Distance between Home and Place of Work

Some taxpayers do not report the distance between private home and place of work. For these persons we complete records using the German Socio-Economic Panel (GSOEP). We only use data of GSOEP respondents who report the distance between private home and place of work and who report a distance of 10 kilometers or less. The reason for this procedure is that in our reference period the standard allowance was exhausted for distances exceeding 13 kilometers. According to official statistics the commuting allowance accounts for some 70 to 75 per cent of the total work-related expenses. Therefore we assume that taxpayers report distances between private home and place of work of 11 kilometers or more, because even with low other itemized work-related expenses their tax base will be smaller when they report the distance compared to the case where they claim the standard lump-sum allowance only.

Imputation of Number of Workdays

In addition to information on the distance between home and place of work, we must also provide data on the number of workdays for the "lazy" taxpayers who do not report their commuting expenses. We use the median of all taxpayers reporting commuting expenses based on official statistics. For single filers, the value is 221 days, for joint filers we obtain 223 days for husbands and 220 days for wives.

Uprating Monetary Values

Our base data set refers to the assessment period 2002. The time lag between the reporting period and the evaluation date is due to (legal) delays in filing tax returns which can take up to several years. This time lag makes it necessary to uprate the monetary values in our model to current price levels. We do this using the consumer price index published by the German Statistical Office.

Marginal Tax Rates

Based on the uprated taxable income we calculate individual marginal tax rates using the first derivative of the income tax function according to § 32a EStG.

APPROACH TO MODELLING

Introduction

The central impetus of our work was to provide an interface that is user-friendly, allows easy simulations and scenario analysis and fast delivery of the results to decision makers. Therefore, we used Microsoft Excel® as a platform.

The computationally intensive data processing steps using complex statistical methods were performed with a high speed computer at the University of Hohenheim using the commercially available software Stata®. The final results are incorporated into the Excel model. As the complex, time-consuming task of data processing has not to be performed repeatedly, the response time to decision makers is almost imperceptible.

Individual Data

Our model is a comparative static microsimulation model. The model engine includes the individual data that are used for the simulations in a spreadsheet. Each of the more than 36,000 rows of this spreadsheet corresponds to a certain number of taxation cases given by inflation factor. Further particulars of the taxation cases include the following

- assessment type (i.e. single or joint filers),

- distance between private home and place of work,

- number of working days,

- audited and itemized "other" work-related expenses,

- taxable income, and

- individual marginal tax rate.

Based on this micro-data set changes of the established law can be simulated.

Revenue Effects

For the simulation of the effects of tax changes we perform a kind of marginal analysis.

First, we calculate work-related expenses in the current legal position, or in any new legal position adjustable by the user. We not only take into regard commuting and other work-related expenses, but also the 4,500 € limit of quantitation of public transport expenses described in the subsection on the legal background of our model.

Second, we simulate changes of the tax base in different scenarios. To calculate the effect of a change of the tax base on individual tax receipts we multiply the difference of the pre- and post-reform tax bases with the individual marginal tax rate. Then we multiply with the inflation factor und sum over all taxation cases in the data set. Implicitly we assume a constant marginal tax rate. This is plausible, because the changes in taxable income to be expected realistically are not expected to lead to dramatic changes of the marginal tax rates but in very few cases.

Finally, the income tax changes are used to calculate the solidarity surcharge, that is a federal tax levied at a rate of 5.5 percent of the personal income tax liability, when this liability exceeds the limit of 972 € (single filers) or 1,944 € (joint filers).

The model output includes the aggregate total revenue effect and its breakdown in wage tax, that is income tax payable on the wages and salaries of employees collected by way of deduction from the source, income tax (other than wage tax), and the solidarity surcharge.

"Winners" and "Losers"

For any tax reform, we check whether an individual tax payer is better off, worse off or whether the financial position does not change. Then we apply inflation factors, gross up to the total population and obtain the total number of "winners" and "losers" of a tax reform.

APPLICATION EXAMPLES

This section contains a series of examples that illustrate uses of our model.

In our introduction we have alluded to an ongoing discussion in Germany on the constitutionality of the virtual abolishment of the distance allowance for commuting between the private home and place of work as of January 1st 2007. Since then, this allowance will be granted only as from a distance of 21 kilometers. The Fiscal Courts of Lower Saxony (decision of February 27th 2007, 8 K 549/06) and the Saarland Fiscal Court (decision of March 22nd, 2007, 2 L 2442/06) consider the commuter tax regulations as unconstitutional. In its decision of August 23rd 2007 (VI B 42/07) the German Federal Fiscal Court, that is the Court of last resort within the German jurisdiction over tax and customs matters, has also announced doubts concerning the constitutionality of the new regulation as well. A final decision of the highest court in Germany, the Federal Constitutional Court, is to be expected in 2008.

In Germany today, many call for a revision of the current legal regulations, but there is no mutual consent on counterfinancing. Our model provides a flexible, easy-to-use tool to assess the impact of various changes in the deductibility of work related expenses.

For example, the speakers of the German Chancellor Angela Merkel and the Minister Of Finance Peer Steinbrück have thought aloud about financing a return to the pre-2007 commuting allowance by a revenue-neutral decrease of the flat-rate allowance, currently amounting to 920 €. With a few mouse clicks a user of our model shows that this proposal results in far more losers than winners. A revenue-neutral 0.25 € per kilometer commuting allowance starting with the first kilometer in distance requires the flat-rate allowance to fall from 920 € to 600 €. Compared to the current legal status, 9.9 million taxpayers would be relieved, but in return 18.6 million taxpayers would have to pay more taxes.

The gap between winners and losers increases for a commuting allowance of 0.25 € accompanied by a revenue-neutral reduction of the flat-rate allowance to 780 €. In this case we obtain 7.9 million winners and 20.6 million losers. The reason for this effect is the fact that many employees have no or very small work-related expenses. For them, the reduction of the 920 € flat-rate allowance acts as a tax increase. For example, a top earner with an individual marginal tax rate of 45 per cent and no work-related expenses would have to pay 144 € more per annum, plus solidarity surcharge, plus church tax if applicable.

If revenue neutrality is to be achieved, if the 920 € flat-rate allowance for work-related expenses remains unchanged and if the commuting allowance starts with the first kilometer in distance, then the commuting allowance must drop to 0.15 €. In this case, the number of winners is greater than the number of losers. Some tax lawyers have announced doubts concerning the legitimacy of such a relatively small commuting allowance. They argue that work-related expenses, even if typified, have to be in line with the actual costs which are considerably higher than 0.15 € per kilometer in distance. Contrary to this opinion, many renowned economists, tax jurists and legal practitioners advocate a complete cancellation of the commuting allowance in the context of a comprehensive *tax cut cum base broadening* tax reform (see e.g. Bareis, 2004; Franz, 2007; Wernsmann, 2007).

A user of our model quickly assesses the effects of other alternatives. Assume that the Minister of Finance is prepared to give up revenue neutrality. What happens if we introduce a 0.20 € per kilometer in distance commuting allowance? Well, almost 8 millon employees profit from this reform, while only some 700,000 long-distance commuters lose. Cost in terms of revenue foregone: 750 millon €.

And what if the tax authorities are ready to give up one billion Euro? The amount of 1.08 billion € could be used to finance a commuting allowance of 0.22 €, the number of losers could be reduced to 400.000. And what if the tax man is ready to take 1.6 billion less? This amount could finance a commuting allowance of 0.25 €. Almost ten million commuters would be better off, the number of losers would be close to zero.

FURTHER RESEARCH

Our model refers to the current assessment period 2008. Future research has to concentrate on maintenance and updates of the model. If the legal framework changes then model adjustments are necessary. Examples for such changes include:

- *Revisions of the income tax schedule (§ 32 a EStG).* Revisions of the tax schedule require a recalculation of marginal tax rates. In this study we use the 2008 tax function. Other tax functions are programmed easily.

- *Abolishment of marital income splitting.* The current system of marital income splitting is a permanent bone of contention in tax policy discussions (see Seel, 2007, for the status quo of this discussion). Its abolishment would require us to consider both spouses as single filers which can be done without difficulty.

- *Introduction of new calculation methods for commuting expenses* requires an adjustment of programs and re-computation of the allowance. Currently, the model allows at most three distance steps.

- *Substantial changes in the tax-deductibility of certain types of income-related expenses* require an adjustment of the data base. This calls for a repetition of the computationally intensive matching process.

In the medium term, the model requires adjustments to changing socio-economic and demographic structures in the population of taxpayers.

CONCLUSION

In this paper, we present a new approach to microsimulation modelling of work-related expenses with special consideration of the commuting allowance. Our aim is to facilitate an informed evaluation of policy options regarding work-related expenses in the German tax system.

The underlying philosophy in developing our model has been to make it as user-friendly as possible. We have developed a "front end" which allows the user to change the parameters of the tax rules governing work-related expenses, activate the run, and view the inputs and outputs from within an interactive environment.

The model is used routinely by the German Ministry of Finance and some of its results have been quoted in the media. Hopefully, the model will help to objectify tax policy discussions.

REFERENCES

Bareis, P. (2004). Ist die Abschaffung der Entfernungspauschale sinnvoll? *ifo-Schnelldienst*, (5):9–11.

Buck, J. (2006). *Datenfusion und Steuersimulation. Theorie und Empirie im Rahmen des Mikrosimulationsmodells GMOD.* Shaker, Aachen.

European Environment Agency (2007). *Size, structure and distribution of transport subsidies in Europe*. Number 3. EEA Technical report.

Franz, W. (2007). Entfernungspauschale. *ZEWnews*, 10:8.

IEEP et al (2007). *Reforming environmentally harmful subsidies*. Final report to the European Commission's DG Environment.

Schrinner, A. (2007a). Eine erneute Reform der Pendlerpauschale kennt auch Verlierer. *Handelsblatt*, 212:3.

Schrinner, A. (2007b). Steinbrück bleibt bei Pendlerpauschale hart. *Handesblatt*, 173:3.

Seel, B., editor (2007). *Ehegattensplitting und Familienpolitik*. Deutscher Universitäts-Verlag, Wiesbaden.

Wagenhals, G. and Buck, J. (2007). Möglichkeiten und Grenzen der Datenfusion für Mikrosimulationsmodelle. In Zwick, M. and Merz, J., editors, *MITAX – Mikroanalysen und Steuerpolitik*, Statistik und Wissenschaft, Band 7, pages 84–105, Wiesbaden. Statistisches Bundesamt.

Wagner, G. G., Frick, J. R., and Schupp, J. (2007). The German socio-economic panel study (SOEP) – Scope, evolution and enhancements. *Schmollers Jahrbuch (Journal of Applied Social Science Studies)*, 127(1):139–169.

Wernsmann, R. (2007). Die Neuregelung der Entfernungspauschale ist verfassungsgemäß. *Deutsches Steuerrecht*, 27:1149–1154.

AUTHOR BIOGRAPHIES

Prof. Dr. rer. pol. habil. GERHARD WAGENHALS is Full Professor of Statistics and Econometrics at the University of Hohenheim in Stuttgart, Germany, and Research Fellow at the Institute for the study of Labor (IZA) in Bonn. He received his diploma in economics from the University of Tübingen, Germany, in 1976; his doctoral and habilitation degrees were received from the University of Heidelberg, in 1980 and 1984. He worked as a postdoc at the Department of Economics, University of Pennsylvania (1980–1982), as an Associate Professor (C2) at the University of Heidelberg (1986–1988), as a Visiting Professor at the Department of Economics, University of Bern (Switzerland) (1989–1990), as a Professor of Computational Economics at the Department of Economics, University of Paderborn (1990–1992), and since then as Full Professor of Statistics and Econometrics at the University of Hohenheim. His key research interests are microsimulation modelling and microeconometrics. His email is `wagenhls@uni-hohenheim.de` and his personal webpage at `http://www.statistik.uni-hohenheim.de`.

Dr. oec. JÜRGEN H. BUCK studied economics at the University of Hohenheim in Stuttgart, Germany. He obtained his diploma in economics in 2002. He has been working as a management consultant in McKinsey's Stuttgart Office serving primarily public sector clients. He is also doing research in econometrics at Hohenheim university together with Gerhard Wagenhals where he obtained his PhD equivalent (Dr. oec.) in econometrics in 2006 working on statistical matching for microsimulation models. His key research interests are practical uses of statistical matching and microsimulation based tax modelling.

SELECTING MUTUAL FUND PERSISTENCE: THE GERMAN CASE DISCOVERING PERSISTENT MANAGERS IN GERMANY

Jorge Sainz, Javier Otamendi, Pilar Grau and Luis Miguel Doncel
Universidad Rey Juan Carlos, Campus Vicálvaro
Facultad de Ciencias Jurídicas y Sociales
Departamento Economía Aplicada I
Paseo Artilleros s/n
28032 Madrid, Spain
E-mail: franciscojavier.otamendi@urjc.es

KEYWORDS

Model, π in the C-Ky, X-Y-Z Analysis, Persistence.

ABSTRACT

In mutual fund industry, managers' ability to generate continuous value in excess compared with the benchmark index is a crucial aspect. Focusing on the German market in this research we apply several methods which let us to avoid statistical problems related to multiple hypothesis testing in traditional financial techniques. By doing so we obtain a threshold value λ that delimits what is considered the true null hypothesis. Our main result is that managers' action is of little significance with only a small part of them adding excess value to mutual funds they run.

INTRODUCTION

The ability of mutual fund managers to beat the market has long been placed in doubt. Literature shows that only a relatively small percentage of the whole universe of mutual funds has been persistently better than the index they use as a benchmark. Nevertheless positive persistence measures may be upwards bias, referring as ability something that is more related to luck. This result may have several effects as it affects managers' retributions which are related to performance and the marketing of the funds, also associated to traditional performance measures.

Traditional measures of persistence are based on Jensen's alpha, where the ability of managers to obtain abnormal results is tested against the behaviour of the market, while including additional factors like size, or book to market or momentum or not. Although these measures have been widely used, they do not take into account the existence of lucky funds, that is, funds that have significant estimated alphas (positive or negative), but zero true alphas. The existence of those values misinterprets the real results on the behaviour of mutual funds and makes it necessary to correct the results by taking in account the statistical shortcomings of the method used to estimate individually the alphas, which strongly rests on the confidence interval chosen.

We estimate the true number of persistent funds whether that persistence by the managers is positive or negative. Previous European wide studies show that persistence as estimated by traditional methods is fairly low both in the positive and negative sides and that the number of funds that shows the ability of management strongly depends on the level of accuracy chosen for the sample. We are able to find the true number of funds that show persistence on the sample, regardless of the confidence level used on the description of the results.

This approach also improves the search of true alphas. Previous papers, as Kosowski, Timmermann, White, and Wermers (2006) for the US and Cuthbertson, Nitzsche and O'Sullivan (2005) for the UK use a more simple bootstrap technique to estimate the ability of managers on positive or negative marginal alphas, while the former is only appropriate for extreme alphas values (either positive or negative). Similar problems are also found on Bayesian analysis as the ones developed by Baks, Metrick, and Wachter (2001) and Pastor and Stambaugh (2002), that yield results that are only relevant on the margin. Additionally, in the Cuthbertson, Nitzsche and O'Sullivan (2006) case, as Nuttall (2007) shows, the results may erroneously identify funds as having skill they do not possess or vice versa, rendering inaccurate results on extreme cases.

In this paper we are going to follow the discoveries of Benjamini and Storey (Benjamini and Hochberg 1995; Benjamini and Yekutieli 2001; Storey 2002) in order to find whether those alphas obtained during the standard estimation process for persistence are actually attributable to the manager's success or to the "false discovery rate". In those cases we will reject to hypothesis that managers create value to investors, that is, they are getting paid for nothing.

PERFORMANCE MEASURES

Managers' ability is commonly measured in different ways widely used in the industry, such as alphas, betas, Sharpe ratios, information ratios, etc. In analyzing performance, traditional measures rely on the capital

market line developed on the 1960s and became more sophisticated through the inclusion of various factors that take into account the evolution of stocks. Their advantage is that they are relatively simple to obtain and the results are easily comparable.

Traditional performance measures are based on the capital asset pricing model derived by Sharpe (1964) and Lintner (1965), who establish a linear relationship between the excess return of an investment and its systematic risk, including an alpha term:

$$R_{it} - R_{ft} = \alpha_i + \beta_{it}(R_{mt} - R_{ft}) + \varepsilon_{it} \quad (1)$$

These measures, similar to the ones developed by Treynor and Mazuy, use the CAPM security line, with the intercept of the regression general expression. The Jensen a is interpreted as a measure of the funds' performance with respect to the market benchmark chosen, where R_{it} is the return in month t of the fund i, R_{ft} the return on a one-month T-bill in the same month, R_{mt} is the return of the benchmark for the period t and ε_{it} is the error term.

To include styles of management Fama and French (1993) developed a model that incorporate two other factors: size (SMB) and book to market (HML) to which Carhart (1997) added a new variable that captures the momentum (tendency) factor (by using the difference between a portfolio that incorporate past winners minus a portfolio of past losers (MOM)):

$$R_{it} - R_{ft} = \alpha_i + \beta_{1i}(R_{mt} - R_{ft}) + \beta_{2i}SMB_t + \beta_{3i}HML_t + \beta_{4i}MOM_t + \varepsilon_{it}$$
$$(2)$$

Any of the models are estimated using Ordinary Least Square (OLS) regression. As it is well known, each estimated α_i comes with its corresponding variance estimator, so that p-values that correspond to the null hypothesis of zero alpha are readily calculated. Individual p-values smaller than a pre-specified significance level γ indicate individual persistence, with the sign of the α_i indicating positive or negative persistence. The number of persistent managers is obtained by repeating the test for each of the i=1…I managers of mutual funds independently.

SIMULATION METHOD TO ESTIMATE TRUE PERSISTENCE

What has been described in the previous section is known as individual, independent hypothesis testing and differs from multiple testing, which covers all the funds jointly.

In individual hypothesis testing, an observed value *x* is compared against a threshold value that result of the

application of a significance level γ, and a decision is taken by deciding to reject or not reject (accept) the null hypothesis. It is well known that when taking this acceptance/rejection decision, two errors might be committed: rejecting the null hypothesis when it should have been accepted (probability=γ) or accepting it when it should have been rejected (probability=ν).

The null hypothesis in this case is that the given mutual fund manager is not persistently different than the market (α_i=0), whereas the alternative is that the managers do really behave differently. Table 1 shows the decision problem in tabular form.

Table 1: Problem in Tabular Form

		DECISION TAKEN	
		ACCEPT NULL	REJECT NULL
NULL HYPOTHESIS	TRUE	CORRECT DECISION	γ
	FALSE	ν	CORRECT DECISION

In multiple hypothesis testing (Table 2), the number of observed values is high (M) and the procedure should detect those null hypothesis that are really true (O) and those that are really false (A). If γ is used for each individual test, the probability of committing errors is greatly increased: the probability of accepting just the true null hypotheses is only $(1-\gamma)^O$, and the probability of rejecting all the ones that are false is only as high as $(1-\nu)^A$. Therefore, in other to take good decisions at the aggregate level, it is necessary to lower γ to the point where $(1-\gamma)^O = \Gamma$, Γ defined as the overall significance level. If O is high and Γ is low, γ will be very close to zero, making it very complicated to reject any individual null hypothesis, thus committing decision errors of not rejecting false null hypothesis.

Table 2: Multiple Problem in Tabular Form

	ACCEPT $H_{0,i}$	REJECT $H_{0,i}$	TOTAL
TRUE NULL	P	F	O
FALSE NULL	N	T	A
TOTAL	W	R	M

Efforts have been made to attack the multiple hypothesis problem from different angles so that the chance of correctly accepting the truly significant alternative hypothesis (A) rises. That includes both the correctly rejected null hypothesis (T) and also those for which the null hypothesis is incorrectly accepted (N).

Two of these main efforts are those of Benjamini and Storey with their fight for controlling the FDR (False Discovery Rate), that is, the number of individual null hypothesis that are rejected that should have been accepted. First, Benjamini developed a sequential algorithm (Benjamini and Hochberg 1995, Benjamini and Yekutieli 2001), which later is improved by Storey (Storey 2002). The latter´s original idea is to include a threshold value λ that delimits what is considered true null hypothesis instead of using the individual γ to reject null hypothesis (Figure 1).
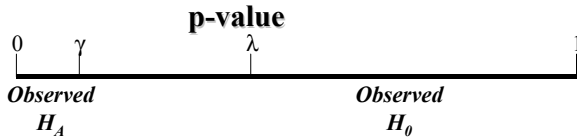
p-value

Figure 1: γ and λ in the p-value axis

The sequential approach, which gives as a result the estimated number k of total true positive rejections of the null, as a function of Γ and λ, is as follows:

1. Let $p_{(1)} \leq \ldots \leq p_{(I)}$ be the ordered, observed p-values for the I hypothesis test
2. Find k such that:

$$\hat{k} = \max\{k: \hat{FDR}(p_{(k)}) \leq \Gamma\}$$

$$= \max\{k: \frac{\hat{\pi}_0 p_{(k)}}{k/M} \leq \Gamma\}$$

$$= \max\{k: \frac{\frac{\#\{p_i > \lambda\}}{(1-\lambda)M} p_{(k)}}{k/M} \leq \Gamma\}$$

$$= \max\{k: \frac{\frac{W(\lambda)}{(1-\lambda)M} p_{(k)}}{k/M} \leq \Gamma\}$$

Storey (2002) later developed the bootstrap procedure to select the best combination of λ and Γ in terms of γ. In this procedure the mean square errors of the estimations of the FDR measure is minimized. The algorithm could be summarized as follows:

1. Set the individual significance level γ

2. Set the feasible range for λ and Γ

3. For each λ:
   a. Estimate the proportion of hypothesis with a true null, π_o

      i. $\hat{W}(\lambda)$ = observed individual tests with a p-value that exceeds the threshold λ

      ii. $$\hat{\pi}_0(\lambda) = \frac{\hat{W}(\lambda)}{(1-\lambda)*M}$$

   b. Calculate MSE(λ, $\hat{\pi}_0(\lambda)$) and the (1-Γ) percentile of $\hat{pFDR}_\lambda^{*b}(\gamma)$ and using a bootstrap procedure of B samples., in which $\hat{\pi}_0^b(\lambda)$ and $\hat{pFDR}_\lambda^{*b}(\gamma)$ are estimated from a sample of I p-values out of the original p-values:

4. Calculate the "optimum λ", λ*, by choosing the one with the smallest MSE(λ)

5. Calculate

$$\hat{pFDR}_\lambda(\gamma) = \frac{\pi_0(\lambda^*)*\gamma}{\hat{Pr}(p-value < \gamma)*\{1-(1-\gamma)^M\}}$$

The joint application of the algorithms therefore allows to obtain both the number of persistante managers and a measure of how good the estimation is, measured by the pFDR.

DATA AND SAMPLE DESCRIPTION

Our dataset comprises 134 mutual funds registered in Germany. The monthly return data for the funds was provided by Morningstar, and the sample period under consideration covers 11 years, from January 1995 to December 2005. All mutual funds are measured gross of taxes, with dividends and capital gains, but net of fees, the Average Annual Return for the sample period was of 9,42% while its Standar deviation was 17,04.

As market factors we use the DAX XETRA index and the one-month interest rate when calculating excess returns. To calculate the Fama and French (1993) factors we follow different strategies for each factor. For the HML we use the data from French's open database estimated on local currency; to evaluate the SMB we use the differences on the returns of small and large capitalized stocks.

Table 1 summarizes the details of the different series used. The returns shown are that of an equally weighted portfolio that includes all the funds for that country. Results from the returns show that although the data for mutual funds and risk-free assets are quite similar, the data on the other benchmarks differ significantly, as we expected, similar to other European analyses (for example, Otten and Bams, 2002).

Table 3: Summary Statistics For Different Factors And Benchmarks Used

| | Return | STD Dev | Cross Correlations | | | | | |
			Fund	Market	Risk free	SMB	HML	Mom
Fund	9.42	17.04	1.00					
Market	13.56	21.47	0.93	1.00				
Risk free	3.19	10.15	-0.19	-0.15	1.00			
SMB	14.85	42.07	-0.17	-0.09	0.29	1.00		
HML	9.50	19.38	-0.08	-0.14	0.03	-0.10	1.00	
MOM	0.28	0.36	0.02	0.07	0.00	0.11	-0.19	1.00

Table 4: Summary Statistics For The Different Models of CAPM-Based Models

	Alfa	Market	SMB	HML	Mom	Adj R^2	dist +/0/-
Jensen	-0.08	0.75*	-	-	-	0.86	1/84/15
Fama-French	-0.07	0.73*	0.03***	-0.07	-	0.86	1/81/18
Carhart	-0.04	0.74*	0.02***	0.03	-0.01	0.82	0/80/20

*** Significant at the 1% level ** Significant at the 5% level * Significant at the 10% level

ESTIMATION OF ALPHAS

Table 2 reports the OLS results of these measures for an equally weighted portfolio that includes the whole sample and, in the last column, the percentage distribution of the sign of the statistically significant alphas of each method plus those alphas that are not different from zero. Germany has a fairly low percentage of abnormal performance over the period which is consistent with other studies.

The Fama and French factors (HML and SMB) are relevant across estimations and countries, especially for the SMB factor. Less relevant is the Carhart momentum, MOM, which shows little relevance across markets and no significance.

PERSISTENT MANAGERS IN GERMANY

The data just presented is therefore liable for an analysis of persistence, in particular, to calculate the significance of the alphas generated by any of the models (performance measures) presented in section 2.

What follows are the analyses of the results that have been obtained after applying the simulation algorithm to the data for different combinations of country, method, Γ, γ, λ, which have been parameterized with the following values:

- Country={Germany}
- Method ∈ {Carhart, FamaFrench, Jensen}
- Γ∈ {0.005, 0.010, 0.025, 0.050, 0.100}
- γ∈ {0.001,0.005,0.010, 0.025, 0.050}

- λ∈ {0.050, 0.100, 0.150, …, 0.400, 0.450,0.500}

The results (Figure 2) indicate that 9 or 10 of the mutual fund managers show persistence (out of 134). The pFDR is reasonably low even for low values of lambda and individual gamma, growing with the individual lambda, giving the indication that the 10 managers are clearly outstanding and are not lucky.

CONCLUSIONS

To obtain positive persistence, that is, beating the market in a continuous way is the goal for mutual fund managers and they get well paid for that. Traditional methods show that in Germany there are some persistence in the returns. Those abnormal returns are the base both manager's retributions and firms profits, but the result may be the consequence of statistical analysis chosen. Our results show that traditional methods have been overestimating the effect in persistence of managers, both positively and negatively, due to do not taking into account some statistical properties like confidence intervals decisions.

By applying a statistical procedure free of these handicaps to a large dataset of funds registered in Germany, we show that persistence is lower than realized on the German Mutual Fund Literature. Just 7.5% of managers add value to investors whereas more that 90% do not create significant benefits or these profits seem to be more related with luck than ability. Our choice of method yields relevance over the whole

distribution and not just in the margin, as the previous methodology used to evaluate this phenomenon.

Still, this research is just a first step. Further research need to be posed on the development on more sophisticated measures that will yield more robust results. Also it would be worthwhile to analyse a wider sample of countries to obtain a cross markets analysis.

Individual Gamma	0.001	0.005	0.010					
Lambda	0.050	0.100	0.150	0.200	0.250	0.300	0.400	0.500
Overall GAMMA	0.005	0.010	0.025	0.050	0.100			

GERMANY
TRULY SIGNIFICANT P

Carhart						**pFDR**	
	Overall GAMMA					Individual Gamma	
	0.005	0.010	0.025	0.050	0.100	0.0010 0.0050 0.0100	
Lambda 0.05	9	9	10	10	10	0.1025 0.1184 0.1565	
0.10	9	9	10	10	10		
0.15	9	9	10	10	10		
0.20	9	9	10	10	10		
0.25	9	9	10	10	10		
0.30	9	9	10	10	10		
0.35	9	9	10	10	10		
0.40	9	9	10	10	10		
0.45	9	9	10	10	11		
0.50	9	10	10	10	11		

FamaFrench						**pFDR**	
	Overall GAMMA					Individual Gamma	
	0.005	0.010	0.025	0.050	0.100	0.0010 0.0050 0.0100	
Lambda 0.05	9	10	10	10	15	0.0898 0.1151 0.1384	
0.10	9	10	10	10	15		
0.15	9	10	10	10	18		
0.20	9	10	10	10	18		
0.25	9	10	10	10	18		
0.30	9	10	10	10	18		
0.35	9	10	10	10	18		
0.40	9	10	10	10	18		
0.45	9	10	10	10	18		
0.50	9	10	10	10	18		

Jensen						**pFDR**	
	Overall GAMMA					Individual Gamma	
	0.005	0.010	0.025	0.050	0.100	0.0010 0.0050 0.0100	
Lambda 0.05	9	9	9	9	9	0.1119 0.1435 0.1897	
0.10	9	9	9	9	9		
0.15	9	9	9	9	9		
0.20	9	9	9	9	9		
0.25	9	9	9	9	9		
0.30	9	9	9	9	9		
0.35	9	9	9	9	9		
0.40	9	9	9	9	9		
0.45	9	9	9	9	9		
0.50	9	9	9	9	9		

Figure 2. Results for Germany

REFERENCES

Benjamini, Y. and Y. Hochberg. 1995. "Controlling the false discovery rate: a practical and powerful approach to multiple testing." *Journal of the Royal Statistical Society B* 57, No. 1, 289-300.

Benjamini, Y. and D. Yekutieli. 2001. "The control of the false discovery rate in multiple testing under dependency." *The Annals of Statistics* 29, No. 4, 1165-1188.

Barras, L.; O. Scaillet; and R. Wermers. 2006. "False discoveries in mutual fund performance: Measuring luck in estimated alphas." *Robert H. Smith School Research Paper No. RHS 06-043*

Carhart, M. 1997. "On persistence in Mutual Fund Performance." *The Journal of Finance* 52, No. 1, 57-82.

Cuthbertson, K.; D. Nitzsche; and N. O'Sullivan. 2005. "Mutual Fund Performance: Skill or Luck?" *Cass Business School Research Paper*.

Fama, E. and K. French. 1993. "Common Risk Factors in the Returns on Bonds and Stocks", *Journal of Financial Economics* 33, 3-53.

Kosowski, R.; A. Timmermann; R. Wermers; and H. White. 2006. "Can Mutual Fund 'Stars' Really Pick Stocks? New Evidence from a Bootstrap Analysis." *Journal of Finance,* 61, 2551-2595.

Lintner, J. 1965. "The Valuation of Risk Assets on the Selection of Risky Investments in Stock Portfolios and Capital Budgets." *Review of Economics and Statistics* 47, 13-37.

Otten, R., and M. Schweitzer. 2002. "A Comparison Between the European and the U.S. Mutual Fund Industry." *Managerial Finance* 28, 14-35.

Pastor, L and R. Stambaugh. 2002. "Mutual Fund performance and seemingly unrelated assets." *Journal of Financial Economics,* 63, 315-349.

Sharpe, W. 1964. "Capital Asset Prices: A Theory of Market Equilibrium Under Conditions of Risk." *Journal of Finance*, 19, 425-442.

Sharpe, W. 1991. "The Arithmetic of Active Management." *The Financial Analysts' Journal* 47, 7-9.

Storey, J. D. 2002. "A Direct Approach to False Discovery Rates." *Journal of the Royal Statistical Society B* 64, 479-498.

AUTHOR BIOGRAPHIES

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 at the Rey Juan Carlos University in Madrid.

JORGE SAINZ has a BS in Economics, from University Complutense of Madrid. Latter he attended University of Rochester in New York obtaining a Master in Business Administration and Finance. Back in Spain he got a Ph.D. in Economics from the Rey Juan Carlos University and a LL.B. from UNED. He has worked as Chief Economist for CIMD and Product Manager at Yahoo!, were he headed Yahoo! Finance in Spain for several years. Currently he is Associate Professor at URJC in Madrid and invited Professor at UOC, Barcelona. His research focuses on financial

markets and the effect of ICT on product innovation and pricing. His e-mail address is : jorge.sainz@urjc.es and his Web-page can be found at http://www.economiaaplicada.com/.

LUIS MIGUEL DONCEL obtained a BS in Economics, from University Complutense of Madrid. to obtain Master in Economics and Finance from the University of York. He got a Ph.D. in Economics from the Rey Juan Carlos University with a research about foreign exchange rates and simulation. Currently he is a lecturer at Rey Juan Carlos University. His research focuses on

financial markets and Simulation. His e-mail address is: luismiguel.doncel@urjc.es

PILAR GRAU recived a BS in Economics at University Complutense of Madrid. Later on she got a Ph.D. in Economics from the same university with a research about nonlinearites and chaos in financial markets. Currently she is a lecturer at Rey Juan Carlos University. Her research focuses on computational finance and stochastic modeling of financial time series. Her e-mail address is: pilar.grau@urjc.es

A BEHAVIORAL MODEL OF SIMULTANEOUS EXTREME RETURNS

Zsolt Tulassay
Department of Finance
Corvinus University of Budapest
8 Fővám tér, Budapest 1093, Hungary
E-mail: zsolt.tulassay@uni-corvinus.hu

KEYWORDS

simultaneous extreme returns, tail dependence, heterogeneous-agent models

ABSTRACT

A multivariate stylized fact of financial markets is the frequent occurrence of simultaneous extreme returns. A simple agent-based behavioral model is presented that accounts for this phenomenon. Joint extremes are generated by heterogeneous fundamentalist traders whose perceptions of the effect of a common news factor on asset values become aligned during market stress. Simulation results are compared to an empirical investigation of two major Hungarian stocks.

INTRODUCTION

A fundamental insight of finance is the idea that diversification is beneficial. It is based on the observation that asset prices move together imperfectly. For a long time, ever since Markowitz founded portfolio theory, these comovements were described by a linear correlation structure, where the benefits of diversification result from imperfect correlations between returns. During the same period, however, several financial market crashes occurred, in the course of which asset prices fell simultaneously, displaying comovements that were much stronger than usual. Market wisdom summarizes this as "correlations rise to one in times of market stress". While correlations certainly do not become one, joint occurrence of such extreme returns, whether negative or positive, might still seriously affect portfolio risk and diminish the benefits of diversification.

The occurrence of such simultaneous extreme returns is necessarily a multivariate phenomenon that requires multivariate analysis. The statistical properties of *univariate* return data have been extensively studied in the financial literature. These properties, commonly referenced to as the stylized facts of financial returns, include heavy tails, change and clustering of volatility, aggregational Gaussianity and a particular scaling behavior, among others (see e.g. Dacorogna et al., 2001; Cont, 2001). Much less attention has however been devoted to the systematic study of the corresponding *multivariate* properties of returns. Notable exceptions include (Breymann et al., 2003); a brief overview of multivariate stylized facts is given in (McNeil et al., 2005). Still, in recent years several statistical tools have been developed to handle such phenomena. There are now time series models like multivariate GARCH models that allow for a time-varying and volatility-dependent correlation structure. Stationary models of simultaneous extremes include various classes of copulas and multivariate EVT models.

There is a lack, however, in models that try to explain these multivariate phenomena in behavioral terms. Non-equilibrium, heterogeneous-agent models have been remarkably successful in replicating univariate stylized facts, while also providing insights into the price formation process (see e.g. Arthur et al., 1997; Brock and Hommes, 1998; Lux and Marchesi, 2000; Farmer and Joshi, 2002). However, very few papers have examined the interrelations and dependence of several asset markets in a heterogeneous-agent framework.

This paper tries to fill this gap by studying asset price dynamics and dependence in a multi-market, heterogeneous-agent framework. Based on the multi-market model of (Westerhoff, 2004), I present a simple behavioral model that can account for extremal dependence in financial returns. The mechanism behind the model is based on the idea that in times of market turmoil (i.e. after a large news shock) the reactions of fundamentalist agents, being heterogeneous themselves, become aligned and highly sensitive to later news.

EMPIRICAL INVESTIGATION

In order to be able to discuss, let alone measure (and test for) extremal dependence, it is necessary to give a sufficiently precise definition first. This is formulated via the *tail dependence coefficient* – a notion that usually arises in the copula literature. In this section I present a non-parametric estimation of tail dependence, based on the empirical cumulative distribution function (cdf) of the data.

The data investigated cover intradaily (2-hour) and daily bivariate log-returns of Hungarian stocks MOL and OTP, from January 2002 to April 2007. These two stocks were selected because they are among the most frequently traded in the Hungarian stock market. Besides that, the two companies are from different industries (energy and finance, respectively), therefore simultaneous extreme returns are not likely to be caused by industry-
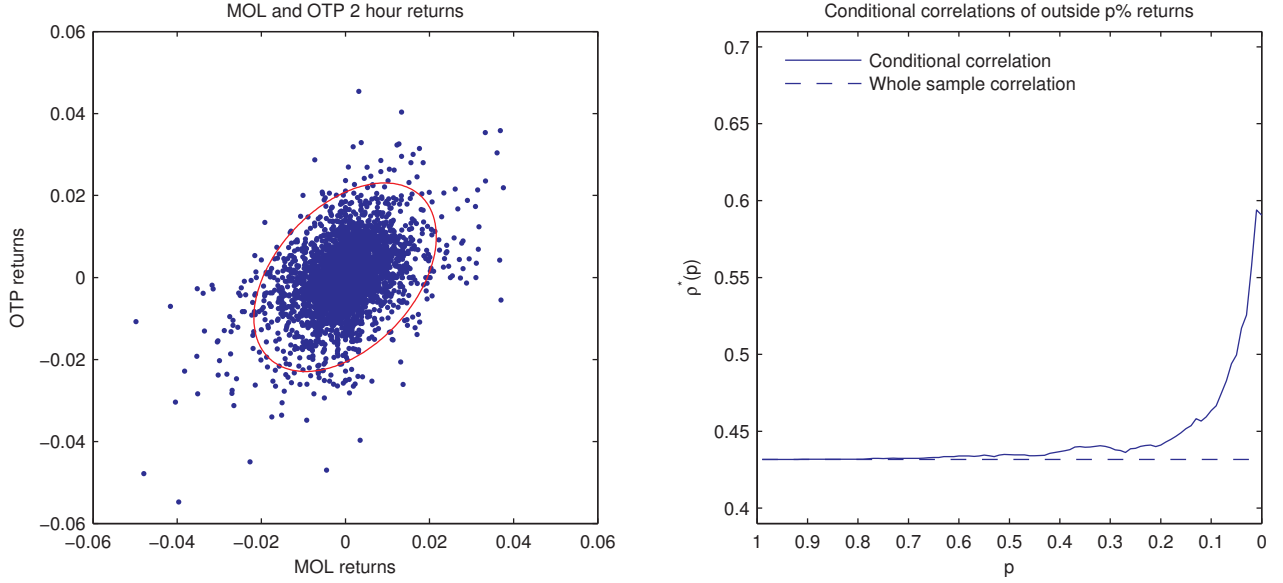
Figure 1: MOL and OTP 2 hour returns scatterplot with the ellipse containing 95% of the data (left); conditional correlations of return data outside ellipses of progressively larger size (right)

related shocks. Additionally, an extended sample of daily returns (ranging from January 1998 to October 2007) was also investigated. The purpose of this was to include data from the Russian financial crisis of August 1998, during which several simultaneous extreme returns occurred. The inclusion of this period alters results for the daily data considerably.

Tests of elliptical symmetry

Though not directly related to extremal dependence, it is quite instructive to test the data for elliptical symmetry – in other words, to test whether a single linear correlation coefficient (or, in higher dimensions, correlation matrix) is sufficient to describe dependence between returns. Elliptical distributions have the property that their densities' contours are ellipses. It is also true that conditional distributions of elliptically distributed random vectors lying outside such quantile ellipses retain the same correlation matrix as the original random vector. For a more detailed discussion of elliptical distributions, see (McNeil et al., 2005).

Thus, for an elliptically distributed random vector X, the conditional correlation $\rho^*(p)$, defined as

$$\rho^*(p) = \rho\left(X \mid h(X) \geq c\left(p\right)\right)$$

is expected to remain approximately stable with p approaching 0 (where $h(x) = (x - \widehat{\mu})' \widehat{\Sigma}^{-1} (x - \widehat{\mu})$, $c(p)$ is a multivariate analogue of the univariate quantile function and $p \in [0, 1]$). In contrast, estimates for the MOL and OTP 2 hour return data show that the conditional correlation is increasing in the tails (see Figure 1). The plot suggests that extreme events might be more strongly correlated than normal ones. However, this method is only exploratory and does not allow one to come to any for-

mal conclusion. Elliptical symmetry can be tested instead by standardizing the data and testing for spherical symmetry. In the polar representation, angles should be uniformly distributed; for the data investigated here, this is rejected at 95% in both the intradaily and the extended daily sample.

The tail dependence coefficient

A shortcoming of the standard linear correlation coefficient discussed above is that it is intimately linked to the marginal distributions. A better dependence measure would be one that only depends on the dependence structure, that is, on the *copula* of a multivariate distribution. The *tail dependence coefficient*, discussed in this section, is such a measure. Moreover, using the tail dependence coefficient it is possible to define extremal dependence precisely.

The tail dependence coefficient λ is essentially the limiting conditional probability of quantile exceedances. One can distinguish between upper and lower tail dependence. Upper tail dependence is the (limiting) probability that X_2 exceeds its q-quantile, given that X_1 exceeded its q-quantile. Similarly, lower tail dependence is the (limiting) probability that X_2 falls below its q-quantile, given that X_1 fell below its q-quantile. In financial applications we are more concerned with downside risk, therefore here I am mainly interested in the lower tail dependence coefficient:

$$\lambda(q) = \Pr\left(X_2 \leq F_2^{\leftarrow}(q) \mid X_1 \leq F_1^{\leftarrow}(q)\right) =$$
$$= \frac{\Pr\left(F_1(X_1) \leq q, F_2(X_2) \leq q\right)}{\Pr\left(F_1(X_1) \leq q\right)} = \frac{C(q, q)}{q}$$
$$\lambda = \lim_{q \to 0^+} \lambda(q)$$
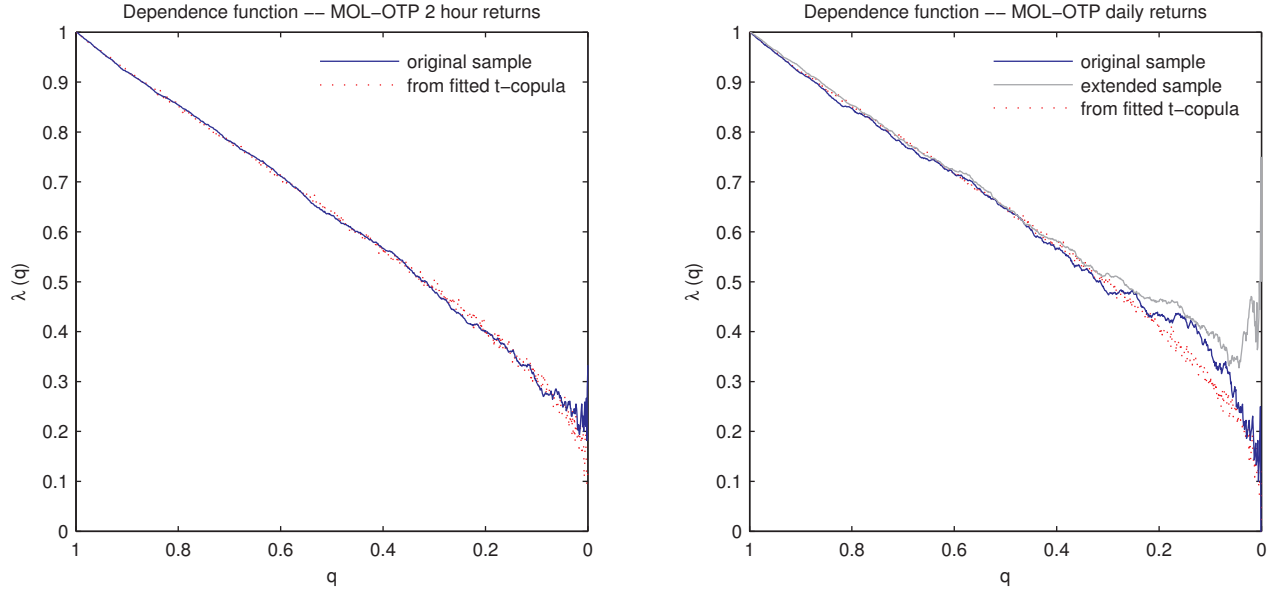
Figure 2: Dependence functions $\lambda(q)$ of MOL and OTP 2 hour (left) and daily (right) returns. Dotted lines are dependence functions from fitted t-copulas.

I will call $\lambda(q)$ the quantile-based dependence function; C is the (unique) copula of the multivariate distribution. If $\lambda = 0$, then X_1 and X_2 are *asymptotically independent* in the lower tail; if $\lambda \in (0, 1]$ then they are said to show *(asymptotic) lower tail dependence*. It can be shown that the Gaussian copula (and thus, in particular, the multivariate normal distribution) is asymptotically independent for $\rho < 1$, while the t copula (and therefore the multivariate t-distribution) is asymptotically dependent for $\rho > -1$ (see e.g. McNeil et al., 2005).

Estimating $\lambda(q)$ from the data is straightforward; it is even more appealing since this can be done in a non-parametric way, using the empirical cumulative distribution (in the copula parlance this step is usually referred to as constructing a pseudo-sample of observations). Figure 2 displays the estimated dependence functions for the 2 hour and the daily return datasets. Table 1 presents the estimates of the tail dependence coefficient (averages for the smallest 2% of q's). Intradaily returns seem to be asymptotically dependent. For the daily returns, the picture is less clear; including data on the Russian financial crisis however results in a much higher λ.

	$\widehat{\lambda}$
2 hour returns	0.2278
daily returns (extended sample)	0.4576
daily returns	0.1575

Table 1: Estimated tail dependence coefficients (average of the last 2% of $\lambda(q)$ as $q \to 1$).

Based on the estimated dependence functions, extremal dependence is clearly present in intradaily returns. It is also present in daily returns – although somewhat less pronouncedly if the crisis data is omitted. Thus, its

effects should not be neglected and it is worthwhile to investigate possible kinds of behavior that could result in such joint extreme outcomes.

MODELING SIMULTANEOUS EXTREMES

In the remainder of this paper I present a model that provides a behavioral explanation for the occurrence of these simultaneous extreme returns. The model follows the heterogeneous-agent, non-equilibrium modeling strand of literature, which in recent years has been successful in replicating univariate stylized facts of financial markets. The current model is a modified, stochastic version of the model presented in (Westerhoff, 2004).

Assets

There are two risky assets ($k \in \{1, 2\}$), traded in two separate, symmetric asset markets of equal size. The fundamental values of assets are governed by two different kinds of shocks. One is a common news process N_t, which affects all assets simultaneously, but to a different, asset-specific extent (determined by a coefficient $\alpha^{(k)}$). The other kinds of shocks, denoted by $\varepsilon_t^{(k)}$, are idiosyncratic, asset-specific, and independent across assets (as well as across time). Their effect on the fundamental value is controlled by the coefficient $\beta^{(k)}$. The (log) fundamental values therefore evolve as follows:

$$F_{t+1}^{(k)} = F_t^{(k)} + \alpha^{(k)} N_t + \beta^{(k)} \varepsilon_t^{(k)},$$

where $N_t \sim$ iid $N(0, 1)$ and $\varepsilon_t^{(k)} \sim$ iid $N(0, 1)$, and the squares of α and β sum to one, thus resulting in a random walk for F. One can interpret N_t as incorporating common, macroeconomic or political news that affect the entire economy, and $\varepsilon_t^{(k)}$ as company-specific news (e.g.

quarterly reports) or shocks which affect only a particular company.

Price dynamics

Prices of the two assets are determined on order-driven markets, in the spirit of (Kyle, 1985) and (Farmer, 2002). There are three types of agents operating in the markets: market makers, fundamentalists and chartists. Orders are initiated by the latter two types against market makers; these fulfill excess demand or absorb excess supply, and adjust prices according to a loglinear market impact function. This results in the following price dynamics:

$$S_{t+1}^{(k)} = S_t^{(k)} + a^{(M,k)} \left(D_t^{(F,k)} + W_t^{(k)} D_t^{(C,k)} \right)$$

where $S^{(k)}$ is the log price of asset k, $D^{(F,k)}$ and $D^{(C,k)}$ are the orders of fundamentalists and chartists in market k, and $W^{(k)}$ is the fraction of chartists currently active in market k (see below for details). The (positive) coefficient $a^{(M,k)}$ determines the sensitivity of market makers' price adjustment reactions to orders; thus it is essentially a measure of liquidity, or depth, of market k.

Agents' trading strategies

Agents base their trading decisions on their price expectations: they buy if they expect an increase in the price and vice versa. Their demand (orders) is as follows:

$$D_t^{(F,k)} = a^{(F,k)} \left(E_t^F \left[S_{t+1}^{(k)} \right] - S_t^{(k)} \right)$$
$$D_t^{(C,k)} = a^{(C,k)} \left(E_t^C \left[S_{t+1}^{(k)} \right] - S_t^{(k)} \right)$$

where $a^{(F,k)}$ and $a^{(C,k)}$ are positive reaction coefficients.

The basic difference between the two types of traders (fundamentalists and chartists) is in how they form their expectations. *Chartists* are simple trend-followers:

$$E_t^C \left[S_{t+1}^{(k)} \right] = S_t^{(k)} + b^{(C,k)} \left(S_t^{(k)} - S_{t-1}^{(k)} \right) \qquad (1)$$

Chartists' presence in a market is reflected by $W_t^{(k)}$, the fraction of them active in market k at time t. They enter a market if they find it attractive enough, where attractiveness is measured as follows:

$$A_t^{(k)} = \log \frac{1}{1 + f^{(k)} \left(S_t^{(k)} - S_{t-1}^{(k)} \right)^2}.$$

The relative percentage of chartists trading in market k at time t is given by the discrete choice model of (Manski and McFadden, 1981), a common vehicle of heterogeneous-agent models:

$$W_t^{(k)} = \frac{\exp \left(\gamma A_t^{(k)} \right)}{\sum_k \exp \left(\gamma A_t^{(k)} \right)},$$

where the parameter γ reflects the intensity of choice.

In contrast, *fundamentalists* are themselves heterogeneous; they differ in their perceptions of the fundamental value, but they all expect prices to revert back to this (perceived) fundamental value:

$$E_t^{F,i} \left[S_{t+1}^{(k)} \right] = S_t^{(k)} + b^{(F,k)} \left(\widehat{F}_t^{(k,i)} - S_t^{(k)} \right) \qquad (2)$$

where i refers to the specific fundamentalist trader.

Perceptions of fundamental values

The essential part of the model is how fundamentalists form and update their $\widehat{F}_t^{(k,i)}$ perceptions of fundamental values. It is assumed that the true values of both types of news (N_t and $\varepsilon_t^{(k)}$) are known to fundamentalists at time t. However, they cannot assess correctly the specific *extent* to which each of this news affects the true fundamental values – that is, they do not know the true $\alpha^{(k)}$ and $\beta^{(k)}$ coefficients. Instead, they have different, subjective, time-varying *perceptions* about the effects of news on the fundamental values. These will be denoted by the coefficients $\widehat{\alpha}_t^{(k,i)}$ and $\widehat{\beta}_t^{(k,i)}$.

Each fundamentalist trader thus has a subjective perception of the fundamental value of asset k:

$$\widehat{F}_{t+1}^{(k,i)} = \widehat{F}_t^{(k,i)} + \widehat{\alpha}_t^{(k,i)} N_t + \widehat{\beta}_t^{(k,i)} \varepsilon_t^{(k)}$$

These are then used to form subjective price expectations according to Equation (2).

The fundamental behavioral assumption concerns how fundamentalist traders' perceptions of the effect of news vary over time. The basic idea is as follows: fundamentalists differ in their sensitivities to news; however during a market turmoil traders start to ascribe much higher importance to news than usually, as they lose their anchors of value. In highly volatile times traders cannot distinguish between important and unimportant information, so they regard every piece of information as possibly highly important and react accordingly (that is, overreact).

In terms of the model there is a population of $\widehat{\alpha}$'s and $\widehat{\beta}$'s, the members of which all rise together and become much less dispersed during a market turmoil. This shows up as follows: with N_t large enough in absolute terms, a "panic term" ξ_t kicks in, and by raising all perceived α's simultaneously this makes all fundamentalists much more sensitive to news:

$$\widehat{\alpha}_t^{(k,i)} = \widehat{\alpha}_{t-1}^{(k,i)} + \xi_t + c^{(k)} \left(\alpha^{(k)} - \widehat{\alpha}_{t-1}^{(k,i)} \right) + d^{(k)} \eta_t^{(k,i)}$$

$$\xi_t = \begin{cases} 0.5 & \text{if } |N_t| > 2 \\ -0.01 & \text{otherwise} \end{cases}$$

$$\eta_t^{(k,i)} \sim \text{iid } U(-1,1)$$

An additional random term η ensures that fundamentalist traders remain heterogeneous.

Different sensitivities thus become aligned and high during market turmoils. This is the mechanism that "generates" simultaneous extreme returns. Over time these
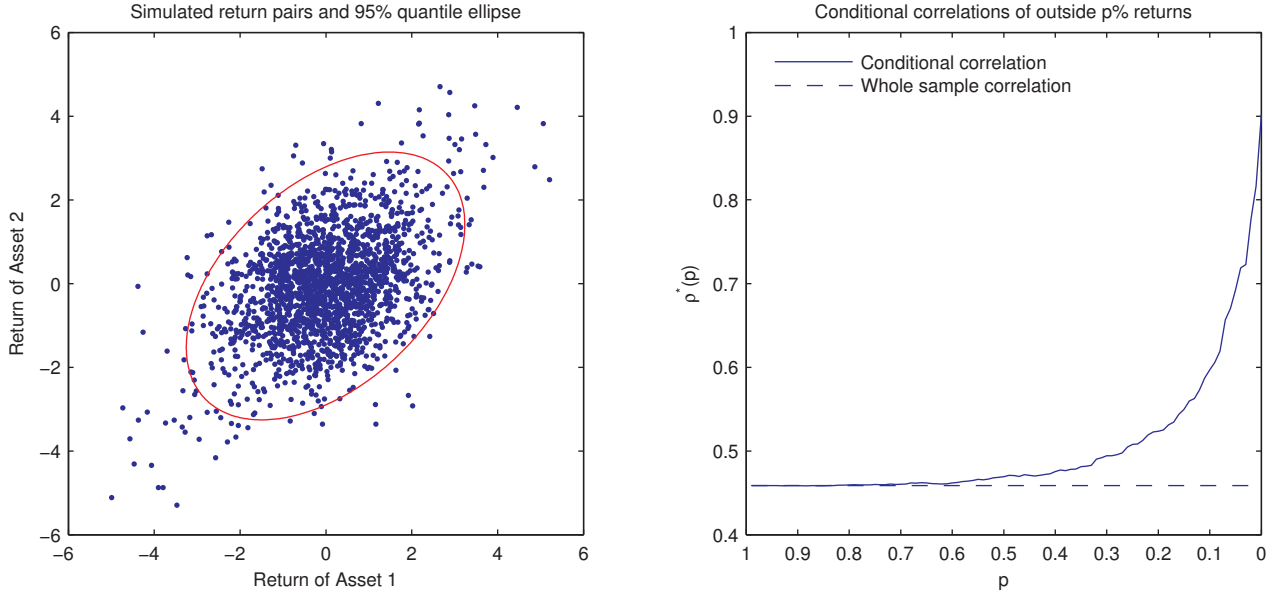
Figure 3: Scatterplot (left) and conditional correlation plot (right) of simulated returns

sensitivities eventually return to more moderate levels and will be dispersed around the true value, or even go below it. This allows the price of assets to break away from their fundamental values in the short run. However, the market on average "gets the price right" in the long run, even if individual perceptions about the fundamental values show considerable variation over time and across traders.

SIMULATIONS

I performed simulations for 2000 consecutive trading periods to see whether the model is able to produce joint extreme returns. Table 2 contains the parameter values used for simulations; N and M denote the number of fundamentalists and chartists, respectively. As there is little empirical guidance on how to pick the parameter values of such models, I based them mostly on parameter choices used by (Westerhoff, 2004).

T	2000	f	1
N	100	γ	2
M	100	α	$\sqrt{0.2}$
$a^{(M)}$	1	β	$\sqrt{0.8}$
$a^{(F)}b^{(F)}$	1	c	0.05
$a^{(C)}b^{(C)}$	0.5	d	0.1

Table 2: Parameter values of the simulations

In the simulations, the $\widehat{\alpha}$'s of fundamentalists were initially uniformly distributed around the true α's, while the true β's were known to the agents.

Figure 3 displays the scatterplot of returns of a typical simulation and the conditional correlation plot calculated from the same simulated return series. The plots display features that are roughly similar to those of real

world data displayed in Figure 1: extreme returns cluster at the lower-left and upper-right corners of the scatterplot, signaling a relatively high occurrence of joint extremes, and conditional correlation is rising towards the tails. Likewise, elliptical symmetry is rejected at 95% confidence level.
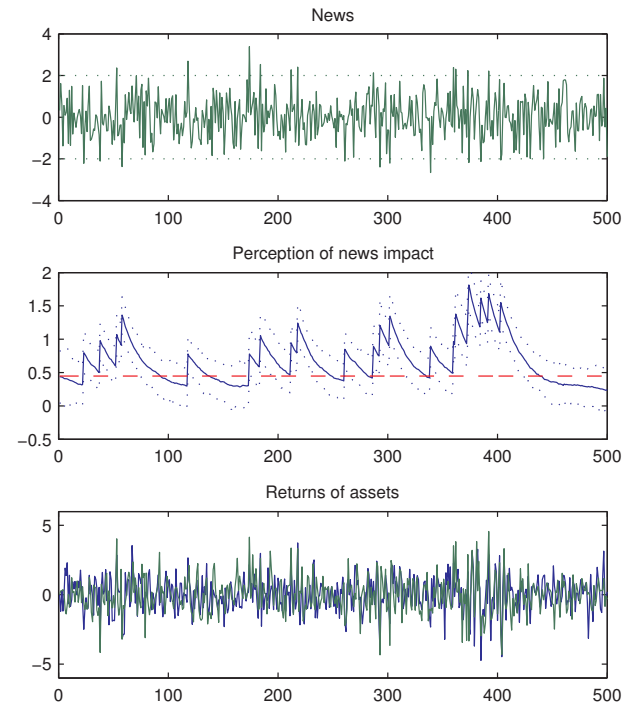
Figure 4: Common news N_t (top); variation of average α's over time (middle, solid line) and the true α (dashed line); returns of assets (bottom).

Figure 4 reveals the inner workings of the model. The
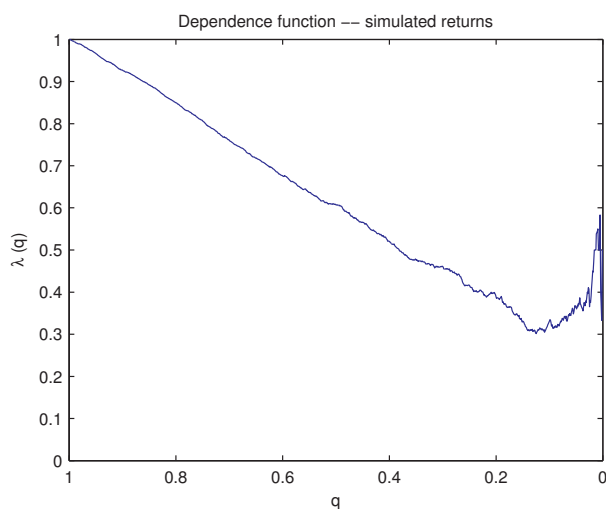
Figure 5: Dependence function of the simulated returns.

top panel displays a realization of the N_t common news over 500 trading periods; the middle panel plots the variation of fundamentalists' average α's over time (along with the 5% and 95% quantiles); the lower panel shows the asset returns. It is clear that periods with higher volatility correspond to periods with higher perceptions of news impact (triggered by large news).

Figure 5 displays the dependence function calculated from a randomly chosen simulated return series. It clearly shows lower tail dependence. While limiting behavior is different from the intradaily sample, it is roughly similar to its counterpart calculated from the extended daily sample.

Simulated returns also reproduce a number of univariate stylized facts. First, they display excess volatility: prices deviate from fundamental values in the short run, but revert back in the long run. In fact, as there is no market consensus among fundamentalists on the true fundamental value, deviations from fundamental values ("bubbles") can last for quite a long time. There is no significant autocorrelation in returns, which is in line with hypotheses of market efficiency. There is however autocorrelation in the second moments, which signals clustered volatility. In addition, returns are fat-tailed: tails follow a power law with a tail index slightly above 3.

CONCLUSION

The model presented above demonstrates that it is possible to replicate the multivariate stylized fact of simultaneous extreme returns using non-equilibrium, heterogeneous-agent models. Additionally, the model also reproduces a number of univariate stylized facts. These results are promising.

The main behavioral assumption in the model is the aligned reaction of heterogeneous fundamentalists. While this mechanism is plausible enough, it is nevertheless important that any behavioral assumption one entertains in an agent-based model be well-grounded. Therefore, stronger empirical behavioral underpinnings are probably needed to support the model's assumptions.

REFERENCES

Arthur, W. B., Holland, J. H., LeBaron, B., Palmer, R., and Tayler, P. (1997). Asset Pricing under Endogenous Expectations in an Artificial Stock Market. In Arthur, W. B., Durlauf, S. N., and Lane, D. A., editors, *The Economy as an Evolving Complex System*, volume II, pages 15–44. Addison-Wesley.

Breymann, W., Dias, A., and Embrechts, P. (2003). Dependence Structures for Multivariate High-Frequency Data in Finance. *Quantitative Finance*, 3(1):1–16.

Brock, W. A. and Hommes, C. H. (1998). Heterogeneous Beliefs and Routes to Chaos in a Simple Asset Pricing Model. *Journal of Economic Dynamics and Control*, 22(8-9):1235–1274.

Cont, R. (2001). Empirical Properties of Asset Returns: Stylized Facts and Statistical Issues. *Quantitative Finance*, 1:223–236.

Dacorogna, M., Gençay, R., Müller, U., Olsen, R. B., and Pictet, O. V. (2001). *An Introduction to High-Frequency Finance*. Academic Press.

Farmer, J. D. (2002). Market force, ecology and evolution. *Industrial and Corporate Change*, 11(5):895–953.

Farmer, J. D. and Joshi, S. (2002). The price dynamics of common trading strategies. *Journal of Economic Behavior & Organization*, 49:149–171.

Kyle, A. S. (1985). Continuous Auctions and Insider Trading. *Econometrica*, 53(6):1315–1335.

Lux, T. and Marchesi, M. (2000). Volatility clustering in financial markets: A micro-simulation of interacting agents. *International Journal of Theoretical and Applied Finance*, 3:675–702.

Manski, C. F. and McFadden, D. L., editors (1981). *Structural Analysis of Discrete Data with Econometric Applications*. MIT Press.

McNeil, A. J., Frey, R., and Embrechts, P. (2005). *Quantitative Risk Management*. Princeton University Press.

Westerhoff, F. H. (2004). Multiasset Market Dynamics. *Macroeconomic Dynamics*, 8:596–616.

AUTHOR BIOGRAPHY

ZSOLT TULASSAY is a Ph.D. student at the Department of Finance at Corvinus University of Budapest. His main topics of interest include quantitative risk management, the microstructure of financial markets and agent-based modeling. His e-mail address is `zsolt.tulassay@uni-corvinus.hu`.

A MODEL FOR THE MEASUREMENT OF ECONOMIC FREEDOM

Ángel Díaz and José Luis Montes
Universidad Rey Juan Carlos
Departamento de Economía Aplicada I
Paseo de los Artilleros sn., 28032, Madrid., Spain
E-mail: angel.diaz@urjc.es; joseluis.montes@urjc.es

KEYWORDS

Model, Structural Equations Modelling, SEM, Index construction, Economic freedom.

ABSTRACT

The paper presents, by means of a structural equations model, an account of the economic freedom factors and their importance. Considering that economic freedom manifests through: business freedom, trade freedom, fiscal freedom, government size, monetary freedom, investment freedom, financial freedom, property rights, freedom from corruption and labour freedom. Results show that all the considered *freedoms* have statistical significance, being the most influential the ones related to the property rights and the freedom from corruption. Besides, an objective economic freedom index is developed through a new, structural equations based, methodology.

INTRODUCTION

Although the current world liberalization process is one of the most important economic trends in the current years, there are a lack of econometrics models helping to measure and understand its effects and the structure and importance of its underlying economic and social factors. Such a model will enhance the comprehension of the phenomena and assist to their management.

This paper aims to contribute to attenuate these lacks, by developing -applying structural equations modelling- an index to measure the economic freedom of a country, or of a significant economical area. Finding out what variables, and to what extent, influence the liberalization process; in order to establish the most efficient economics measures, necessary to reach the required liberalization degree. The definition of economic freedom, considered in this paper, is an *absence of government coercion or constraint*. The obtained statistical results imply not desirability at all, and are not meant to have any ideological implications.

The paper proceeds as follows, in the first part we outline the method, in the second we propose and estimate the model, in the third we show the results, concluding, in the fourth, with some conclusions and further research suggestions.

METHOD

There are in the economic literature some high-quality indexes trying to measure the economic freedom. The ones we know are built as a weighted mean of a set of variables. Two main problems arise at this point: The first one is what variables should form part of the index; an economics problem that we are not going to address here. The second is to assign weights to the index components, so that the weight reflects the real importance of the variable. In this paper we propose a solution for this second problem: regard the economic freedom as a latent variable and *let* the data assign weights to the variables.

In order to construct the index, a system of structural equations, whose outline is presented next, with latent (not measurable directly) variables has been developed. Besides, and as a distinct desirable characteristic in the model, the measurement errors are explicitly included for each variable in the estimation process. The system parameters were estimated by the Full Information Maximum Likelihood (FIML) method and checked by Bayes' estimation, obtaining similar values. To estimate the index weights we considered the database of the Heritage Foundation (heritage.org), covering 162 countries, but the proposed method is equally applicable for any other set of variables.

The Structural Equation System General Model

The Structural System General model is a mathematical model, a set of linear equations, which encompass a large class of models (Jöreskorg, K.G., 1984). Variables in the equation's system may be either directly observed or unmeasured latent (theoretical), representing concepts not directly observed, that can only be approached and measured trough observed and measurable variables. Latent variables must be continuous, nevertheless observed dependent variables can be continuous, censored, binary, ordered, categorical (ordinal), counts or combinations of these variable types.

Structural equation models are composed of two sub models: the structural model linking only latent variables, and the measurement model, that links each

latent variable with its correspondent measurement variables (indicators). A *causal* structure among latent variables is assumed.

Structural equation modelling has certain desirable features: a) explicit inclusion of error's measurement terms for each variable, b) simultaneous estimation of parameters of a series of dependence relationships, where a variable can act as dependent in some equations and independent in other ones, c) can take account of reciprocal causation and recursive and non recursive models, d) can be considered as confirmatory technique and, also, as exploratory.

The Model

The Structural System General Model is defined (Jöreskorg, K.G., *op. cit.*) through a system of linear structural relations, whose matrix representation is:

$$\eta = \alpha + B\eta + \Gamma\xi + \zeta \qquad (1)$$

where η (m x 1) and ξ (n x 1) are random vectors of, respectively, latent dependent and independent variables, α is a (m x 1) intercept's terms vector, B (m x m) is the coefficient's matrix for the latent endogenous variables, representing the effects of η-variables on other η-variables, Γ (m x n) is the coefficient matrix for latent exogenous variables representing the direct effects of ξ-variables on η-variables and ζ is a (m x 1) vector indicative of the random disturbance in the equation. It is assumed that $E(\eta) = 0, E(\xi) = 0, E(\zeta) = 0$.

Observed (measured) variables are represented by the vectors y (p x 1), where p is the number of indicators of ξ, and x (q x 1), where q is the number of indicators of η, which are related to de latent variables trough the equations:

$$y = \tau_y + \Lambda_y \eta + \varepsilon \qquad (2)$$

$$x = \tau_x + \Lambda_x \xi + \delta, \qquad (3)$$

where ε (p x 1) and δ (q x 1) are error terms vectors. It is assumed that ε is uncorrelated with η, ξ and δ; and that δ is uncorrelated with η, ξ and ε. Λ_y (p x m) and Λ_x (q x n) are matrices containing the structural coefficients λ_{ij} linking the latent and measured (observed) variables, and τ_y (p x 1) and τ_x (q x 1) are vectors of constant intercept terms.

The structural equation model fundamental hypothesis is: $\Sigma = \Sigma(\theta)$, where Σ is the population covariance matrix and $\Sigma(\theta)$ is the model's covariance matrix, written as a function of a vector of the model parameters θ. Parameter estimates are obtained by minimizing some fit function $F(\theta) = F\left(S, \hat{\Sigma}(\theta)\right)$.

Once estimated the parameters, the model covariance matrix is compared with the empirical data covariance matrix, and if the difference between both matrixes is statistically acceptable the proposed structural equation model is acknowledged as a plausible explanation of the reality.

The measurable variables, below, that have been considered are defined following Holmes *et al.* (2008), and their original measurements scales has been transformed to the range 0-100, to ease the comparisons, meaning 0 minimum degree and 100 maximum. A score of 100 signifies an economic environment or policies most favourable for economic freedom.

Business freedom: ability to create, operate, and close an enterprise quickly and easily.

Trade freedom: is a composite measure of the absence of tariff and non tariff barriers that affect imports and exports of goods and services.

Fiscal freedom: is a measure of the burden of government from the revenue side. It includes both the tax burden in terms of the top tax rate on income (individual and corporate separately) and the overall amount of tax revenue as a portion of gross domestic product (GDP).

Government size: is defined to include all government expenditures, including consumption and transfers. Ideally, the state will provide only true publics goods, with an absolute minimum of expenditure.

Monetary freedom: combines a measure of price stability with an assessment of price of controls. Both inflation and price controls distort market activity. Price stability without microeconomic intervention is the ideal state for the free market.

Investment freedom is an assessment of the free flow of capital, especially foreign capital

Financial freedom: is a measure of banking security as well as independence from government control. State ownership of banks and other financial institutions such as insurer and capital markets is an inefficient burden and political favouritism has no place in a free capital market

Property rights: is an assessment of the ability of individuals to accumulate private property, secured by clear laws that are fully enforced by the state.

Freedom from corruption: is based on quantitative data that asses the perception of corruption in the business environment, including levels of governmental legal, judicial, and administrative corruption.

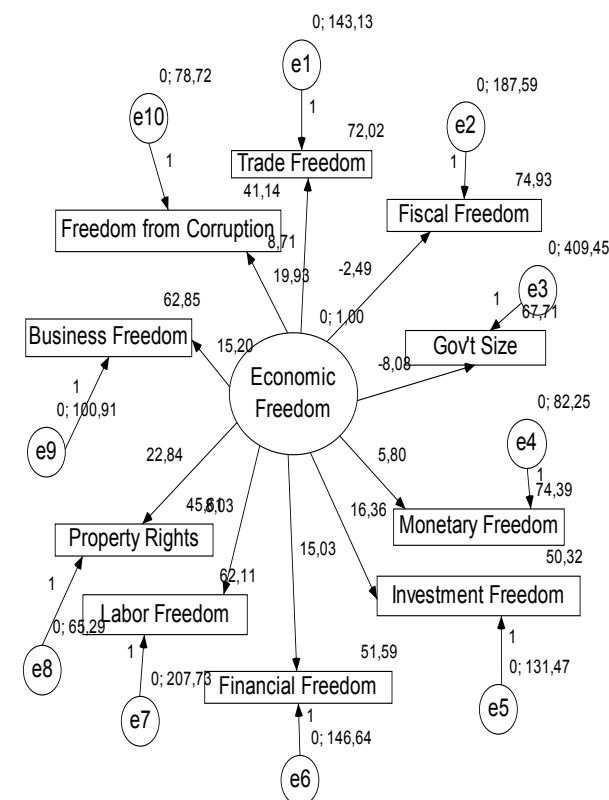Labor freedom: is a composite measure of the ability of workers and businesses to interact without restriction by the state.

Figure 1: Model Outline

RESULTS

Results are shown in figure 1 and in table 1. Figure 1 shows the model estimates' outline. We can observe three types of variables: non-observed latent variables (Economic Freedom), indicators (Table 1), and measurement errors (e). The structural equation 3 relates the matrix of exogenous latent variables ξ with their indicators (x); in this case with the economic freedom and the matrix of the measurement errors. The equation 2 shows the relationships among indicators of the endogenous latent variable (η), being "y" the matrix of these indicators, η the latent endogenous variables matrix, and ε the estimation error. Finally, the equation 1 shows the relationships among non-observed

latent variables that do not exist in this case, but it is also included for the sake of model completeness.

All variables (table 1) are statistically significant (p-value < 0.001), except the Fiscal freedom (p= 0.028). Most of the parameters values are positive with the unexpected exception of Fiscal Freedom and Government size.

The variables with higher importance (with higher weights) are Property Rights and Freedom from Corruption. Statistically, these two variables are the ones that better explain the Index of Economic Freedom variability. From an economic point of view, those countries making more efforts trying to reduce corruption and protect property rights will achieve more economic freedom. This fact is especially important in less developed countries where, as expected, these factors are the first to improve when trying to assure a minimum of economic freedom.

Next to the previous ones more influent variables are those related with finance. Once assured the property rights and the freedom from corruption, the most important variables in the index are Investment Freedom, Financial Freedom and Business Freedom. Countries with fewer barriers of this type will have more economic freedom.

The third group of variables, although with less importance, than the preceding are Freedom of Trade, Freedom of Work and Monetary Freedom.

To conclude, two variables have negative signs: Government Size and Fiscal Freedom. This result is quite surprising and requires further research.

Table 1: Maximum Likelihood Estimates. Regression Weights

Indicators		Estimate	S.E.	P
FiscalFreedom	<--- Ec.Fr.	-2,491	1,132	,028
InvestmentFreedom	<--- Ec.Fr.	16,364	1,323	***
FinancialFreedom	<--- Ec.Fr.	15,033	1,311	***
BusinessFreedom	<--- Ec.Fr.	15,204	1,195	***
TradeFreedom	<--- Ec.Fr.	8,710	1,099	***
LaborFreedom	<--- Ec.Fr.	8,030	1,267	***
PropertyRights	<--- Ec.Fr.	22,842	1,463	***
GovtSize	<--- Ec.Fr.	-8,082	1,722	***
FreedomfromCorrup.	<--- Ec.Fr.	19,931	1,349	***
MonetaryFreedom	<--- Ec.Fr.	5,802	,814	***

Table 2: Standardized Regression Weights

Indicators			Estimate
FiscalFreedom	<---	Ec.Fr.	-,179
InvestmentFreedom	<---	Ec.Fr.	,819
FinancialFreedom	<---	Ec.Fr.	,779
BusinessFreedom	<---	Ec.Fr.	,834
TradeFreedom	<---	Ec.Fr.	,589
LaborFreedom	<---	Ec.Fr.	,487
PropertyRights	<---	Ec.Fr.	,943
GovtSize	<---	Ec.Fr.	-,371
FreedomfromCorruption	<---	Ec.Fr.	,914
MonetaryFreedom	<---	Ec.Fr.	,539

Table 3: Intercepts

Indicators	Estimate	S.E.	P
BusinessFreedom	62,847	1,459	***
GovtSize	67,715	1,745	***
TradeFreedom	72,016	1,185	***
FreedomfromCorruption	41,140	1,747	***
FiscalFreedom	74,928	1,115	***
FinancialFreedom	51,592	1,546	***
LaborFreedom	62,113	1,321	***
InvestmentFreedom	50,318	1,600	***
PropertyRights	45,605	1,940	***
MonetaryFreedom	74,390	,862	***

CONCLUSIONS

In this paper we have proposed, through a structural equations model, an explanation of the economic freedom factors and their importance.

All the considered *freedoms* have statistical significance, with p-values lower than 0.001 (except fiscal freedom p=0.028).

The most influential freedoms are the ones related to the property rights and the freedom from corruption, showing that a sense of safety is of the utmost importance for the economic freedom.

As expected, there is a negative correlation between the economic freedom and the fiscal freedom and between the economic freedom and the government size.

Another contribution of this paper is the development of an economic freedom index. As said above, in spite of the number of economic freedom' indexes in economic literature, the variables weight assignments is somehow subjective and in some cases even lacks of a scientific basis what hinders a rigorous analysis.

REFERENCES

Bentler, P.M. and D.G. Bonett. 1980. "Significance tests and goodness of fit in the analysis of covariance structures". *Psychological Bulletin*, 88, 588–606.

Bentler, P.M. 1990. "Comparative fit indexes in structural models". *PsychologicalBulletin*, 107, 238–246.

Bollen, K.A. 1986. "Sample size and Bentler and Bonett's nonnormed fit index". *Psychometrika*, 51, 375–377.

Bollen, K.A. 1989. "A new incremental fit index for general structural equation models". *Sociological Methods and Research*, 17, 303–316.

Bollen, K. A, and P. J. Curran. 2006. *Latent Curve Models. A Structural Equation Perspective*. Wiley.

Díaz, A. and J. L. Montes. 2005. "La competitividad de las regiones europeas: Una propuesta metodológica para evaluar el efecto de las tecnologías digitales". XXXI Reunión de Estudios Regionales: "El estado autonómico a debate: eficiencia, eficacia y solidaridad". Alcalá de Henares. (Nov).

Díaz, A. and J.L. Montes. 2006. "Digital Technologies and Competitiveness. Effects on Manufacturing Companies: The Spanish Case". In *From Small Firms to Multinationals: Industrial, Entrepreneurial, Managerial, Financial, Fiscal, Transaction Cost and Consumer Perspectives in era of Globalisation*. 231-244. Ioannis-Dionysios Salavrakos (Eds.). Athens.

Díaz, A., J. M. Menéndez, and J.L. Montes. 2006. "Los recursos humanos en la empresa industrial: un análisis empírico sobre la eficiencia del capital humano". *Revista de Trabajo y Seguridad Social*, 274. Centro de Estudios Financieros.

Holmes, King R., Edwin J. Feulner and Mary Anastasia 0'Grady. 2008. *The 2008 Economic Freedom Index*. The Heritage Foundation and the Wall Street Journal.

Jöreskog, K.G. and D.Sörbom. 1984. *LISREL-VI user's guide*. Mooresville, IN. USA. Scientific Software.

ANGEL DÍAZ, Master in Applied Economics Analysis from the Universidad Autónoma de Barcelona (UAB) and B.A. in Economics and in Business Administration from Carlos III University in Madrid. Associate professor in the Economics and Economic History department of the UAB and researcher in the Project Internet Catalonia. He is currently head of the research department of the SEPI Foundation and a member of the Department of Applied Economics I on the Rey Juan Carlos University in Madrid, where he teaches statistics.

JOSÉ-LUIS MONTES. Ingeniero Industrial (Universidad Politécnica de Madrid) and Ph.D. in Economics (Universidad Pontificia Comillas). His research interests are: economic modelling of competitiveness, efficiency and quality. At this time is Professor of Statistics and Mathematics at the Universidad Rey Juan Carlos in Madrid.

Simulation, Experimental Science and Engineering

SIMULINK LIBRARY FOR SIMULATION OF THE PERMANENT MAGNETS SYNCHRONOUS MACHINE DRIVE

Sergiu Ivanov

University of Craiova
Faculty for Electromechanical Engineering
107 Decebal Blv, 200440 Craiova, Romania
E-mail: sivanov@em.ucv.ro

Francis Labrique
Vincent Defosse
Paul Sente
Catholic University of Louvain
Laboratory for Electrotechnics and Instrumentation
3, Place du Levant
B-1348, Louvain la Neuve, Belgium

KEYWORDS
PMSM, vector control, fault operation.

ABSTRACT

Reliability of electromechanical actuation systems as those used in aerospace applications may be improved by a proper choice of the motor, of its control strategy and of the architecture of the power electronics feeding the motor. This paper presents a Simulink library for the simulation of the motor and of its control electronics allowing to choose between three types of motor and between two types of vector control. The models allow the simulation of different fault conditions in order to identify the weak and strong points of each architecture. The research reported in this paper was performed within the European Union FP6 project DRESS (Distributed and Redundant Electromechanical nose gear Steering System) for more electrical aircraft (http://www.dress-project.eu).

INTRODUCTION

The permanent magnets synchronous machine is a well known compact motor, suited for high power density applications. In most of electrical drives using this type of machine, the motor is Y connected, fed by a three phase inverter and equipped with a classic vector control allowing to regulate the torque. A way to improve the reliability of such type of drives consists of choosing a power electronic architecture allowing a two phase remedial operation (Heb et al. 1994; Bolognani et al. 2000; Welchko et al. 2004). In this paper we present a Simulink library allowing to compare under normal and fault operation the use of a classic PM synchronous machine fed by a three phase inverter and equipped with a classic vector control with the use of two variants of a more special type of motor, the HDD servomotor[®] which in addition to a higher torque to mass ratio offers the very interesting feature of having magnetically and mechanically decoupled phases and the possibility to split each phase into two half phases also mechanically and magnetically decoupled. Furthermore in parallel with the classic vector control, a phase by phase pseudo vector control is also considered which can be used with classic PM synchronous machine only for normal operation and for HDD servomotors both for normal and fault operation.

The paper will present the approach for building the different models, their inter-connection and some results of the simulation of the different systems.

As the work described in this paper presents only a part of the models developed within the DRESS project, in the very first stages of the work, a model assembly philosophy was stated. In accordance with this, the models of the motors must have as inputs the phase voltages delivered by the inverter, the mechanical speed and position of the rotor. The outputs must be the phase currents and the electromagnetic torque. The mechanical parameters (rotor inertia, friction) are included in the model of the mechanical load (not presented in the paper) which is then driven by the electromagnetic torque. The controllers which deliver the reference phase voltages must have as input the electromagnetic torque reference, delivered by an upper level controller. Finally, it should be noted that the PWM inverters are considered to act as pure gains blocks in the proposed library.

MOTORS MODELS

The philosophy in building the models was that they must be able to deal with several types of fault:
- a broken phase (i.e a phase in open circuit);
- a phase totally or partially short circuited;
- the loss of a current sensor.

For this reason we have chosen to model the machines in terms of instantaneous phase variables.

Three phase classic PMSM with separate feeding of the phases

The first architecture which is considered is corresponding to a classic PM synchronous motor with separated feeding of the phases, each phase being fed by an H bridge (Figure 1).
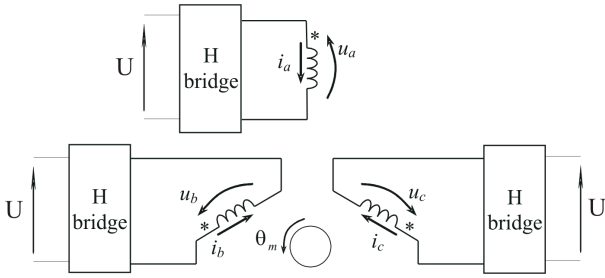
Figure 1: Three Phase Machine with Separated Phases

In this case the electrical equations are (all the notations are specified in the Appendix):

$$\begin{bmatrix} u_a \\ u_b \\ u_c \end{bmatrix} = \begin{bmatrix} R_s & 0 & 0 \\ 0 & R_s & 0 \\ 0 & 0 & R_s \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} L_s & M_s & M_s \\ M_s & L_s & M_s \\ M_s & M_s & L_s \end{bmatrix} \frac{d}{dt} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} +$$

$$+ K_T \dot{\theta}_m \begin{bmatrix} -\sin P\theta_m \\ -\sin\left(P\theta_m - \dfrac{2\pi}{3} \right) \\ -\sin\left(P\theta_m - \dfrac{4\pi}{3} \right) \end{bmatrix}$$

and the electromagnetic torque expression is:

$$T_{em} = -K_T \left(i_a \sin P\theta_m + i_b \sin\left(P\theta_m - \frac{2\pi}{3} \right) + \right.$$
$$\left. + i_c \sin\left(P\theta_m - \frac{4\pi}{3} \right) \right)$$

In these equations, rotor zero position is considered to correspond to the maximum coupling between phase "a" and the rotor, so that the flux induced in the phase "a" by the magnets, Φ_{af} is maximum for $\theta_m = 0$ and, assuming a sinusoidal coupling, can be expressed as follows:

$$\Phi_{af} = \Phi_M \cos P\theta_m .$$

Consequently, the back emf induced by the magnets in phase "a" is equal to:

$$e_a = -P\Phi_M \dot{\theta}_m \sin P\theta_m = -K_T \dot{\theta}_m \sin P\theta_m ,$$

with $K_T = P\Phi_M$.

With this model, the following faults can be simulated:
• one phase in open circuit is simulated by imposing simultaneously its current and di/dt to zero;
• one phase in short circuit is simulated by imposing to zero the corresponding feeding voltage;
• fault of a current sensor is simulated by imposing to zero the feedback value of the current on one phase (phase "a").

Classic PMSM with star connected windings and insulated neutral

If the machine phases are star connected with insulated neutral, it must be fed by a three phase inverter (Figure 2).
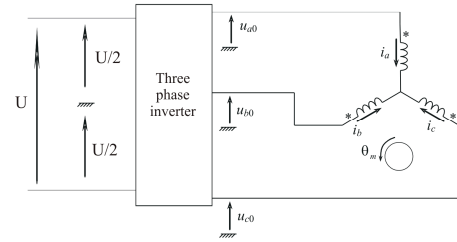
Figure 2: Three Phase Machine Star Connected

Since in this case, $i_a + i_b + i_c = 0$ the matrix

$$\begin{bmatrix} L_s & M_s & M_s \\ M_s & L_s & M_s \\ M_s & M_s & L_s \end{bmatrix} \text{ can be replaced by } \begin{bmatrix} L_0 & 0 & 0 \\ 0 & L_0 & 0 \\ 0 & 0 & L_0 \end{bmatrix},$$

with $L_0 = L_s - M_s$ $\left(M_s < 0 \right)$, the cyclical inductance.

The input variables controlled by the inverter are the line voltages u_a-u_b, u_b-u_c and u_c-u_a and as their sum is equal to zero, only two of them are independent, so the motor equations can be written as

$$\begin{bmatrix} u_a - u_b \\ u_a - u_c \end{bmatrix} = \begin{bmatrix} R_s & -R_s \\ 2R_s & R_s \end{bmatrix} \begin{bmatrix} i_a \\ i_b \end{bmatrix} + \begin{bmatrix} L_0 & -L_0 \\ 2L_0 & L_0 \end{bmatrix} \frac{d}{dt} \begin{bmatrix} i_a \\ i_b \end{bmatrix} +$$

$$+ K_T \dot{\theta}_m \begin{bmatrix} -\sin\left(P\theta_m \right) + \sin\left(P\theta_m - \dfrac{2\pi}{3} \right) \\ -\sin\left(P\theta_m \right) + \sin\left(P\theta_m - \dfrac{4\pi}{3} \right) \end{bmatrix} .$$

With $i_c = -i_a - i_b$, the electromagnetic torque is obtained with the same expression as the previous model.

With this model, the fault that can be simulated is the current sensor failure as for the previous model. It should be noted that writing the equations in terms of line voltage allow to consider without problem a phase partial short circuit even if not done at this day in the library.

HDD servomotor with separated feeding of its six half phases

The HDD servomotors (Figure 3) has six physically and magnetically separated half phases which can be fed by six H bridges.

The model consists in the concatenation of two three phase machines with no mutual inductances between the phases, shifted between them by 180°.

In addition to the faults that can be simulated for the three phase classic PMSM with separated phases (Figure 1), considering the case of partial short circuit of several turns of one winding was of interest for the research. In this case, the equivalent circuit is the one depicted in Figure 4.

It is important to note that in this case a seventh circuit appears corresponding to the section in short circuit. The corresponding equations are
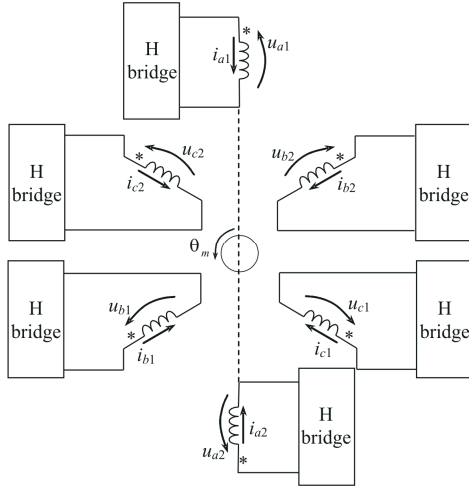
Figure 3: HDD Servomotor
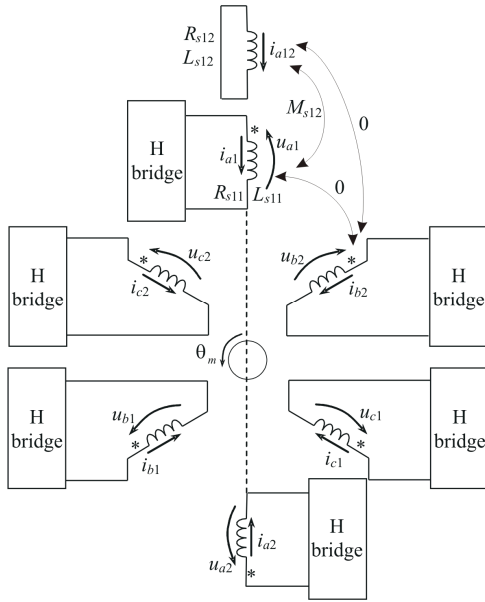
Figure 4: HDD Servomotor with Partial Short Circuit

$$\begin{bmatrix} u_{a1} \\ u_{c2} \\ u_{b1} \\ u_{a2} \\ u_{c1} \\ u_{b2} \\ 0 \end{bmatrix} = \begin{bmatrix} R \end{bmatrix} \begin{bmatrix} i_{a1} \\ i_{c2} \\ i_{b1} \\ i_{a2} \\ i_{c1} \\ i_{b2} \\ i_{a12} \end{bmatrix} + \begin{bmatrix} L \end{bmatrix} \frac{d}{dt} \begin{bmatrix} i_{a1} \\ i_{c2} \\ i_{b1} \\ i_{a2} \\ i_{c1} \\ i_{b2} \\ i_{a12} \end{bmatrix} +$$

$$+ K_T \dot{\theta}_m \begin{bmatrix} 0 \\ -\sin\left(P\theta_m - \dfrac{\pi}{3}\right) \\ -\sin\left(P\theta_m - \dfrac{2\pi}{3}\right) \\ -\sin\left(P\theta_m - \pi\right) \\ -\sin\left(P\theta_m - \dfrac{4\pi}{3}\right) \\ -\sin\left(P\theta_m - \dfrac{5\pi}{3}\right) \\ 0 \end{bmatrix} + \dot{\theta}_m \begin{bmatrix} K_{T11}\left(-\sin P\theta_m\right) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ K_{T12}\left(-\sin P\theta_m\right) \end{bmatrix}$$

with

$$\begin{bmatrix} R \end{bmatrix} = \begin{bmatrix} R_{s11} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & R_s & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & R_s & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & R_s & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & R_s & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & R_s & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & R_{s12} \end{bmatrix},$$

$$\begin{bmatrix} L \end{bmatrix} = \begin{bmatrix} L_{s11} & 0 & 0 & 0 & 0 & 0 & M_{s12} \\ 0 & L_s & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & L_s & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & L_s & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & L_s & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & L_s & 0 \\ M_{s12} & 0 & 0 & 0 & 0 & 0 & L_{s12} \end{bmatrix}.$$

The electromagnetic torque is in this case given by

$$T_{em} = -K_T \left(i_{c2} \sin\left(P\theta_m - \frac{\pi}{3}\right) + i_{b1} \sin\left(P\theta_m - \frac{2\pi}{3}\right) + \right.$$

$$+ i_{a2} \sin\left(P\theta_m - \pi\right) + i_{c1} \sin\left(P\theta_m - \frac{4\pi}{3}\right) + i_{b2} \sin\left(P\theta_m - \frac{5\pi}{3}\right) \right) +$$

$$+ K_{T11} \left(-i_{a1} \sin P\theta_m\right) + K_{T12} \left(-i_{a12} \sin P\theta_m\right)$$

The parameters of the two sections of the partial short circuited phase are computed in accordance with the number of turns in short circuit.

It must be noted that before a partial short circuit of an half phase occurs (i.e. during normal operation) the model has only 6 state equations instead of 7 after the occurrence of the fault. As Simulink does not allow such a change during simulation, some techniques were used for bypassing the problem.

It must be noted that it is possible to use the HDD servomotors as three phase machines with or without a Y connection of the phases and to feed the phases separately via H bridges or via a three phase inverter.

CONTROLLERS MODELS

Two types of controllers were considered for the vector control of the drive.

The first one is the classic vector control in the Park reference frame (Figure 5).
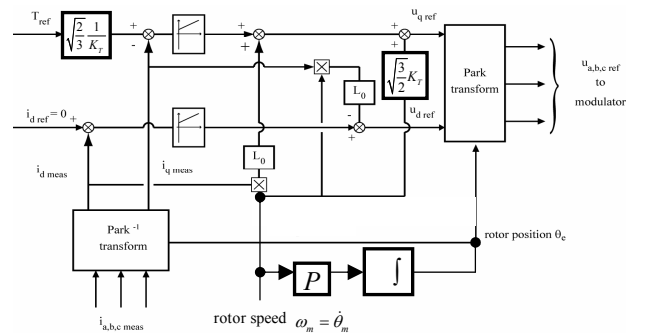
Figure 5: The Classic dq Vector Control of the PMSM

This type of controller can be used with all types of the motors described above provided that the machine emf's are purely sinusoidal. In fact, this is generally not the case as the phase emf's contain generally a significant amount of third harmonic which are homopolar components. For a motor fed by a three phase inverter, these third hamonics are not seen from the terminals due to the Y connection so that the presence or not of the third harmonics in the emf's is without any importance and emf's can be considered as purely sinusoidal as far as the control is concerned. For a motor which phases are fed separately, it is necessary to add to the voltages generated by the controller a third harmonic component equal to that present in the emf's in order to avoid the appearance of homopolar currents. As the third harmonics in the feeding voltages and in the emf's cancel each other, both have been discarded in building the models.

The second type of vector control is a phase-by-phase control with a simulated β phase.

Indeed, with a phase-by-phase current control it is possible to use a pseudo dq control in order to control currents which are constant at constant torque. For that purpose, the following scheme could be used (Figure 6), following denoted dq-αβ control.

For a classical three phase machine, the above model ceases to be valid if one phase is in fault (in open or short circuit) since in this case the sum of the three phase currents is no more equal to zero so that we may no more discard the coupling between the phases by taking in the model the cyclical inductance ($L_0 = L_s - M_s$). For the HDD servomotors, as the phases are magnetically separated, the mutual inductances between the phases are almost equal to zero and the above model is valid for normal or fault conditions.

SIMULINK LIBRARY

By using the models described above, a Simulink library was developed.

Basically, it contains five types of blocks, three corresponding to the three variants of PMSM and two for the controllers.

Concerning the motors blocks, all were developed in a quite compact manner, the heart being an S-function block corresponding to each variant of PMSM. As the models are quite general, the parameters of some predefined types of motors were included in the masks of the blocks being associated to the type of motor chose in the dialog box (Figure 7).

Figure 6: The dq-αβ Vector Control of the PMSM

This control consists in building for each phase of the machine (a,b,c), assimilated to be the α phase of a fictitious αβ machine, the model of the corresponding β phase, since the rotor position $\theta_{el} = P \cdot \theta_m$ is known. For instance, if we assume that phase j is phase α_j we get for this phase

$$u_{\alpha j} = R_s i_{\alpha j} + L_0 \frac{di_{\alpha j}}{dt} - K_T \dot{\theta}_m \sin\left[P\theta_m - 2(j-1)\frac{\pi}{3}\right],$$

the associated fictitious phase β_j being ruled by the equation

$$u_{\beta j} = R_s i_{\beta j} + L_0 \frac{di_{\beta j}}{dt} + K_T \dot{\theta}_m \cos\left[P\theta_m - 2(j-1)\frac{\pi}{3}\right].$$

Figure 7: The Choice of the Type of Motor

The masks allow to use the model for different, by the pre-defined ones, types of motors, by choosing the "Other" type of motor. In this case, the dialog box expands with the necessaries parameters to be specified (Figure 8).
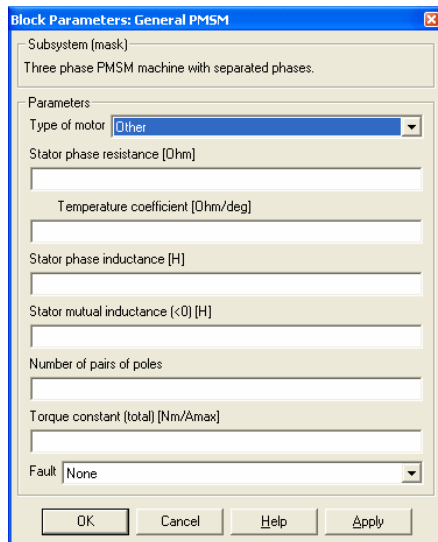
Figure 8: The Dialog Box for "Other" type of Motor

The dialog box of the motors' models allow the choice, during simulation, of one type of fault, as can be seen in Figure 9.
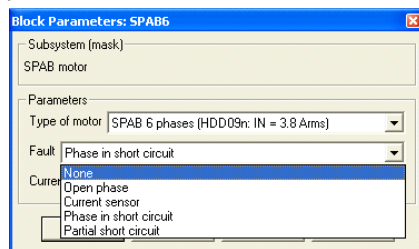
Figure 9: The Choice of the Type of Fault

The dialog box adapts itself to the chosen type of fault, being possible to specify further information, as the location of the short circuit in the case of the entire phase short circuit (Figure 10), or the number of turns in short circuit in the case of partial short circuit of one phase (Figure 11).
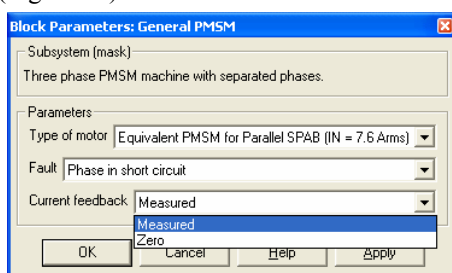
Figure 10: The Location of the Short Circuit of an Entire Phase
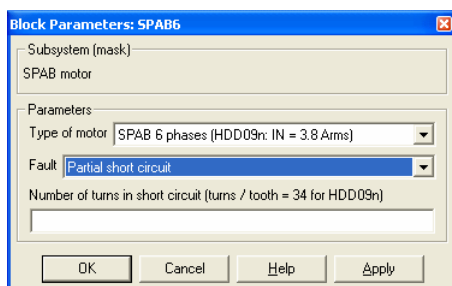
Figure 11: The Number of Turns in Short Circuit

Of course, if a different type of motor is chosen, more parameters must be specified in the dialog box if partial short circuit fault is selected (Figure 12).
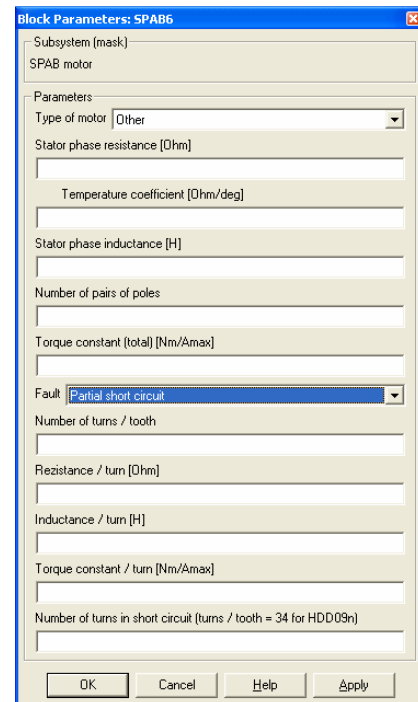
Figure 12: Additional Parameters to be specified for Partial Short Circuit Fault

The two blocks for the vector control can be used for any type of motor and use a special mechanism for automatic identification of the type of motor used in a certain model, necessary for using the proper parameters. This special mechanism was developed by the authors within the masks of controllers blocks, knowing that Simulink does not allow the transfer of the parameters from one dialog box to another. The mechanism is based on identification of the parameter specified in the "Type of motor" field of the motors dialog boxes and automatically updated in its proper field with the same name which is not active (Figure 13).
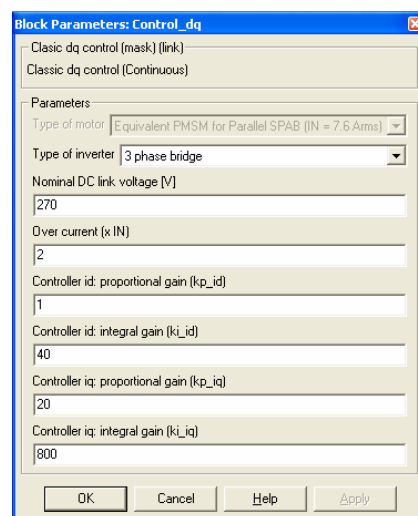
Figure 13: Automatic Update of the Type of Motor in the Controller Dialog Box

If "Other" type of motor is chosen within the dialog box of the motor used in the model, the dialog box of the controllers expands to specify the additional necessary parameters.

SIMULATION RESULTS

As indicated in the introduction, the developed Simulink models are quite versatile. It results a large number of possible combinations motor-controller. For each of the combinations it is possible to simulate different faults which could occur at different operating conditions.

Hereafter will be given only an example obtained with the HDD servomotor and dq-αβ control when one phase is suddenly broken (open circuit). In Figure 14 is plotted a detail of the phase currents and the electromagnetic torque after this fault occurs. As it can be expected , the torque oscillations are quite important, being of the order of the rated torque (imposed earlier in the simulation).
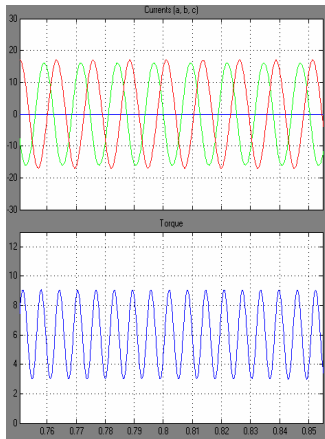
Figure 14: Details of the Phase Currents and Electromagnetic Torque when One Phase is in Open Circuit

Figure 15 allows to follow the phase currents before the fault, when the fault occurs (about 0.3 sec) and after a remedial is applied (about 1 sec).
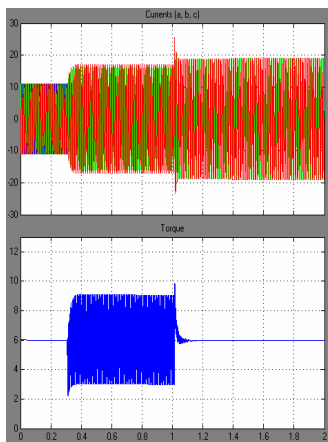
Figure 15: Fault Occurrence and its remedial

The remedial consists in supplying the two remaining phases with currents having adequate amplitude and phase shift with respect to the corresponding emf' (Heb

et al 1994). It is important to note that, the torque oscillations totally disappear provided that, as assumed in the models, the machine emf's are purely sinusoidal, but at the price of higher currents in the remaining phases. This is not quite dangerous, as only two phases contribute now to the heating of the motor. Further thermal simulations will investigate more in detail these aspects. If the phase emf's contain a certain amount of third harmonic with the implemented control strategies, a given amount of torque ripple remains which is directly proportional to the amplitude of the third harmonic.

CONCLUSIONS

The paper describes a Simulink library for the simulation of different structures of PMSM and vector control, in normal operation, as well as when different faults occur.

The developed blocks are quite versatile, allowing the simulation of a large number of possible types and sizes of motors. The vector control blocks use a special mechanism developed by the authors to identify the type of motor used in model and to adjust its own parameters consequently.

The results of the simulations are very useful when the decision on the most suited system would have to be done.

ACKNOWLEDGMENTS

REFERENCES

Bolognani, S.; Zordan, M.; and Zigliotto, M. 2000. "Experimental Fault-Tolerant Control of a PMSM Drive". In *IEEE Transactions on Industrial Electronics*, Vol. 47, No. 5, October 2000, 1134-1141.

Heb, E. et al 1994, "Reliability improvement of field oriented controlled three-phase AC drives by means of two phase remedial operation", In *Proceedings of International Conference on Electric Machines* (ICEM) Vol. 2, 194-198.

Kaswierkowski and al. 1991, "Novel space vector based current controller for PWM inverters". In *IEEE Transactions on Power Electronics*, Vol. 6, No. 1, January 1991.

Séguier, G.; Bausiere, R.; Labrique, F. 2005, *Electronique de puissance*, Editions Dunod.

Welchko, B.A.; Lipo, T.A.; Jahns, T.M.; and Schulz, S.E. 2004. "Fault Tolerant Three-Phase AC Motor Drive Topologies: A Comparison of Features, Cost, and Limitations". In *IEEE Transactions on Power Electronics*, Vol. 19, No. 4, July 2004, 1108-1116.

APPENDIX

In the equations, the following notations were used:
u_a, u_b, u_c – phase voltages;
i_a, i_b, i_c – phase currents;
R_s – phase resistance;
L_s – phase inductance;
M_s – mutual inductance between two phases;
K_T – torque constant;
P – number of pairs of poles;
θ_m – angular position of the rotor.

AUTHOR BIOGRAPHIES

SERGIU IVANOV was born in Hunedoara, Romania and went to the University of Craiova, where he studied electrical engineering and obtained his degree in 1986. He worked for the Institute for Research in Motors, Transformers and Electrical Equipment Craiova before moving in 1991 to the University of Craiova. He obtained his PhD in 1998 with a topic in the field of the control of the electric drives systems. He is involved in modelling of the electromechanical systems.

FRANCIS LABRIQUE was born in Maurage (Belgium) in 1946. He received the degrees "Ingénieur civil électricien" (1970) and "Docteur en Sciences appliquées" (1983) from the Université Catholique de Louvain (UCL).
He is the head of the Laboratory for Electrotechnics and Instrumentation. He has also been invited professor at the Technical University of Lisbon from 1985 to 1990. His research activities are in the pole of power electronics and digital control of electrical drives.
Francis Labrique is co-author of several books on power electronics (in english, in french, and in portuguese) and author or co-author of more than 150 journal or conference papers.

VINCENT DEFOSSE Vincent Defosse was born in Liège in 1979. He received the "Ingénieur civil électromécanicien" Degree from the Université de Liège in 2002. Since then he has worked as a research assistant, first at the Department of Electrical Engineering of the Université de Liège, then at the Department of Electrical Engineering of the Université Catholique de Louvain. His fields of interest are electrical motor modelling using FEM methods and their control.

PAUL SENTE was born in Brussels, Belgium, in 1955. He received the "Ingénieur civil électricien" Degree from the Université Catholique de Louvain (UCL), in 1978.
From 1978 to 1985, he was Assistant with the Laboratory of Electrical Measurements at the same University where he contributed to the development of microprocessor based instrumentation. He then has been a research engineer with the Laboratory of Electrotechnics and Instrumentation, working in the field of the digital control of DC and AC machines.
He was involved in several research projects having as aim the development of special drives.

369

INFORMATION ACQUISITION FOR CITY TRAFFIC MODELS BASED ON IMAGE ANALYSIS

Michał Łyczek
University of Warsaw
ul. Banacha 2
02-097 Warszawa, Poland
michal.lyczek@students.mimuw.edu.pl

Piotr Butryn
University of Warsaw
ul. Banacha 2
02-097 Warsaw, Poland
piotr.butryn@students.mimuw.edu.pl

Agata Chrobak
University of Warsaw
ul. Banacha 2
02-097 Warszawa, Poland
agata.chrobak@students.mimuw.edu.pl

Anna Kulka
University of Warsaw
ul. Banacha 2
02-097 Warsaw, Poland
anna.kulka@students.mimuw.edu.pl

KEYWORDS

Traffic modelling, image processing, vehicle detection, image analysis

ABSTRACT

Fast development of cities and increasing number of cars cause that metropolises need to improve their monitoring system. This paper presents the system which gathers information about traffic and help to monitor and model it. It is not as expensive as current solutions, because it is built on existing infrastructure and low-cost devices. The basic part of this system is a smart phone, on which application is implemented. It contains capturing device and processing unit, as a result it returns information about traffic. Information among other things includes time of the appearance, type of a car and its speed. Image analysis (processing) consist of pre-processing (geometry reconstruction and denoising), classification (geo-metrical methods), object detection and tracking algorithm, which based mostly on calculating gradient. Thanks to this solution, monitoring system can be easily implemented on the places, where it was not possible because of high cost and problem with creating infrastructure.

INTRODUCTION

Metropolises around the world are continuously rebuilding and expanding their boundaries so each year the number of highways and roads is increasing. The continuous growth of the number of vehicles driving on the road, makes a need for a traffic modelling obvious.

All these changes require appropriate decisions in planning and maintaining right functioning of city traffic. All tools, which support making decisions need reliable and actual data. As a result, many metropolises decide to improve their traffic monitoring systems. Current solutions for gathering data are very expensive and require special devices and infrastructure. It is often an impassable barrier for cities in developing countries where large and dynamical changes take place. Wrong decisions can bound potential of cities in the future. Two main disadvantages of existing solutions are:

- using of special devices

- outlay increase

- problems with spare parts

- special transmission infrastructure

- outlay increases

- importability

- time-consuming installation

Above problems were the basis for creating new traffic data acquisition system built on popular low-cost devices and existing transmission infrastructure.

In solution presented in this paper a distributed network of sensors on large junctions will collect information about traffic volume and its distribution. A popular smart phone have been chosen as a single integrated sensor for capturing and processing visual data. Nowadays, those mid-end mobile phones are powerful small computers equipped with camera, fast processor, memory, Internet connection and operating system. Thanks to that smart phones can fulfil a role of independent image processing unit. GSM network which is an inseparable transmission channel for cell phones, small size and low power consumption can be a inexpensive substitute for existing commercial products. These essential features cause that system is not only economical but also more extensible and portable in implementation.
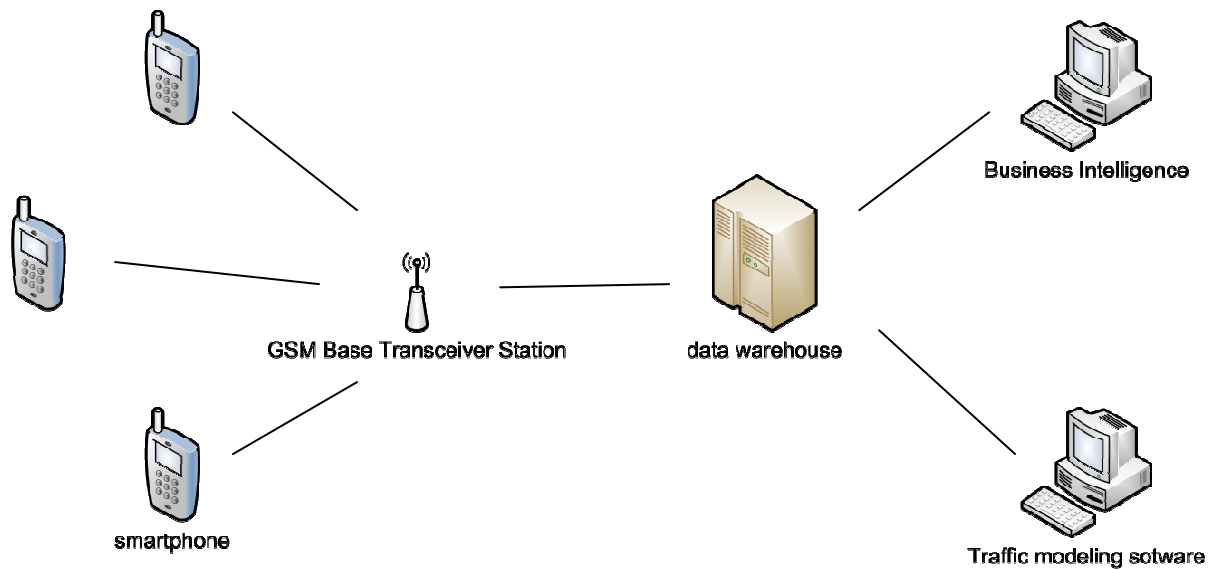
Figure 1. Architecture of the system.

STAGES OF WORK

The first stage of project was to develop a software tools for creating and testing different image processing algorithms. Created software enabled testing various methods on personal computer before implementing final method on smart phone. It also gives a future opportunity to modify algorithm for a special purposes. This software was developed in Java enabling users to easily extend its features by writing user libraries. The processing schema is defined as an XML file which contains processing order and all required parameters. Program has many built-in I/O modules (for reading movie files, image sequence files and real-time camera data) and processing modules (for filtering, segmentation and classification).

The second stage was to choose an algorithm for detecting vehicles. After testing several approaches, final algorithm was implemented on a smart phone. Because of differences between Java distribution for personal computers (Java SE) and mobile devices (Java ME) algorithm implementation needed some changes.

The last stage was to tweak an algorithm for a smart phone architecture. It was rewritten and recompiled for native phone system. This operation improved program efficiency dramatically.

CONCEPT OF THE SYSTEM

Figure 1 shows an architecture of a complete system.
The basic element of the system is a smart phone. Table 1. shows the technical specification of an example testing phone. This device realize several crucial function for the system. Firstly, it is used as a capturing device which grabs visual data from the camera. Secondly, it is used as a processing unit which runs an image processing algorithm. As a result, it returns data about vehicles. After that, the smart phone transmits calculated results, using GSM infrastructure, to a data warehouse. Then info stored in the data warehouse can feed Business Intelligence (BI) software or the Traffic Modelling software.

CPU	332 MHz
Memory	160 MB + 2 GB
Operation system	Symbian OS
Size (width x height x depth)	99 x 53 x 21 mm
Weight	120 g

Table 1. Specification of the smart phone

IMAGE PROCESSING

The key element of the system is image analysis computed on smart phone level. Algorithm cannot be too complicated computationally, due to memory and computing power constraints.
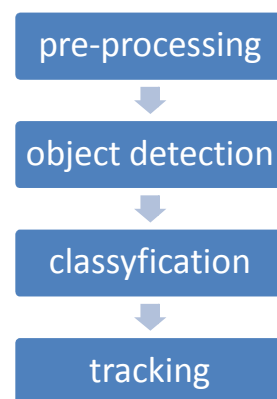
Figure 2. Processing scheme.

d)   original image

c)   geometrical reconstruction

f)   denosing

b)   motion detection

a)   object detection

e)   object classification
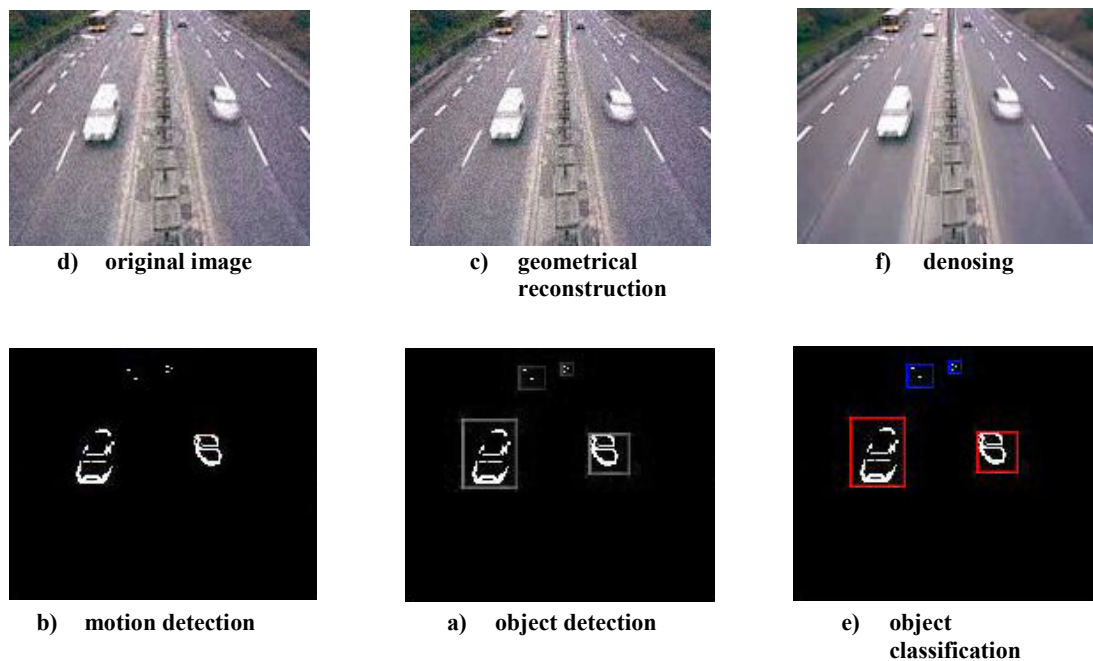
Figure 3. Phases of the algorithm.

Figure 2 presents general processing schema. There have been made some important assumptions about location of smart phone and environment, in order to save complex computations. Significant simplification is a relevant position of camera. As a result, it can be assumed that the vehicles move in specifically way and their shape change in specifically manner.

Pre-processing

Due to small size of optical set-up built-in smart phone the captured image is not perfect. It shows up as a noise and geometrical distortion.
That is why, before proper analysis, image data is pre-processed. In this case it means a geometry reconstruction and denoising. These operations are computationally inexpensive, but considerably improve quality of the further process.

Object detection

In order to find moving objects gradient between adjacent image frames is calculated. To speed up processing it is done only in a few pixels height stripe. More advanced algorithms have been also tested, but due to higher computational complexity and a small quality improvement gradient method has been chosen. Each detected object is added to the global list of vehicles.

Classification

Due to established assumptions about position and orientation of the camera it was possible to base

classification on simple geometrical methods. These methods gave satisfying results in reasonable computational time.

Tracking

Parallel for each car from the list its position is being actualized by analyzing gradient between frames in a rectangular area where the front of a vehicle is expected. If no motion is detected, the object is deleted from the list. This method prevents from taking as cars these objects which appear only in one frame because of noise or light effects.
When a car crosses the last line of measurement, it is taken from the vehicles' list and its velocity is calculated. In order to make the results of our algorithm as close to the reality as it's only possible, the distance between pixels is converted into accurate distance using some trigonometric functions.

Figure 3 shows sequential phases of an algorithm.

CONCLUSIONS AND FUTURE WORKS

Presented system can be used as an inexpensive substitute for complex system. Due to its features it can be easy implemented in places where it was not possible because of high cost or problems with creating infrastructure. This system itself is also a prove that current complex systems solving sophisticated problems can be constructed on popular devices, which were not design for those purpose.

Future works should focus on improving quality of gathering data in weak light condition and at night. At the moment system cannot work properly when there is not enough light (for example unlighted street at night).

Besides, increasing computational possibilities will allow to implement more complex algorithms, which help to gather additional information about traffic and cars.

One of the possible ways of using data from presented system is to use flow network to model traffic. Using algorithms which find maximum flow in the graph, in which the crossroads are nodes and the streets are edges, we can match the capacity with our need.

Based on results of this experience and on the data from our system we can find out where should we extend or build roads to cut down large traffic jam. It can be also used to predict what happen during renovating when some roads or lanes will be closed or what happen when we apply some regulation such as speed limit.

We can also use Dijkstra algorithm in graph, where like in previous case, streets are edges and crossroads are nodes and costs depends on data from our system. Finding the shortest path, we can plan our journey as good as it possible and lost as short time as it possible staying in the jam.

Another type of models which can use data gathered with presented system are CFD based models. Street network is modelled as a set of connected pipes (Figure 4 shows example mesh for the center of Warsaw). Cross-section area of the each pipe reflects capacity of the street. The traffic flow is treated as a continuous fluid flow in fluid dynamics. Data from described system would be used as initial conditions. In the simplest model fluid would be uniform and input data would be density and mean velocity field. In more

advanced case it would be possible to analyse fluid with microstructure, which would reflect diversification of the vehicles on the streets (for example large molecules would represent buses and lorries and small - cars).

ACKNOWLEDGEMENT

REFERENCES

Ali, M. and Clausi, D. 2001. "Using the Canny edge detector for feature extraction and enhancement of remote sensing images." in *Geoscience and Remote Sensing Symposium, 2001.* Vol. 5, pp. 2298-2300.

Jones, B.J.T. 1995. "Low-cost outdoor video motion and non-motion detection" in *Proceedings of the 29th Annual 1995 International Carnahan Conference on Security Technology.* pp. 376-381.

J. Lu and M. L. Liou 1997. "A simple and efficient search algorithm for blockmatching motion estimation" in *IEEE Circuits and Systems for Video Technology*, vol. 7, pp. 429–433, (Apr.)

Hsun-Jung Cho; Hsien-Hung Shih; Ming-Chorng Hwang 2003. "Numerical simulation of a multilane drift-diffusion traffic model" in *Proceedings of IEEE International Conference on Systems, Man and Cybernetics,* vol. 1, pp. 90-95 (Oct.)

AUTHORS
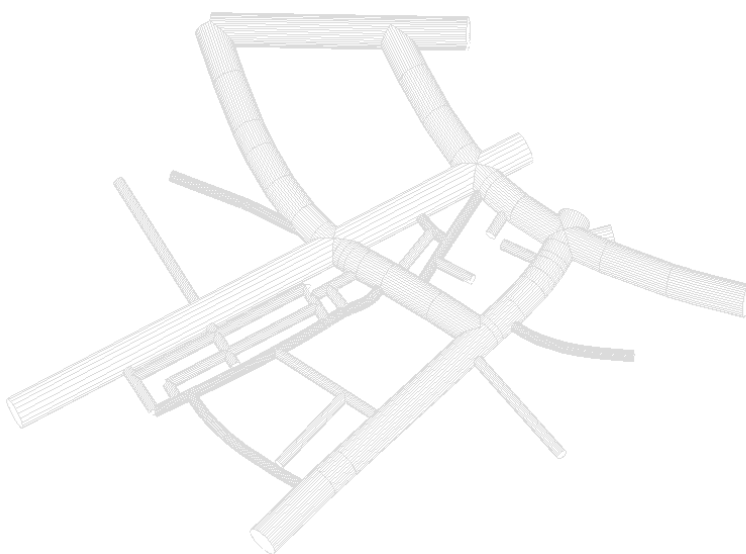
Figure 4. Map of center of Warsaw with corresponding mesh.

MICHAŁ ŁYCZEK born in Warsaw 1986 finished high school with special mathematical profile in 2005. Since then he studies in the Center for Interfaculty Individual Studies in Mathematical and Natural Sciences at the University of Warsaw (Poland). In 2006 he developed a data visualization software for Institute of Geophysics at University of Warsaw. Since 2007 he works on image processing and visualization software in Interdisciplinary Center for Mathematical and Computational Modeling at the University of Warsaw. His email is:
`michal.lyczek@students.mimuw.edu.pl`

PIOTR BUTRYN born in 1986 Warsaw. He finished high school with special mathematical profile in 2005. Since then he studies mathematics and computer studies in the Faculty of Mathematics, Informatics and Mechanics at the University of Warsaw (Poland).
His email is:
`piotr.butryn@students.mimuw.edu.pl`

AGATA CHROBAK born 1987 in Lipno. She finished high school with special mathematical profile in 2005 since then she studies mathematics and computer studies in the Faculty of Mathematics, Informatics and Mechanics at the University of Warsaw (Poland).
Her email is:
`agata.chrobak@students.mimuw.edu.pl`

ANNA KULKA born in Warsaw 1986 finished high school with special mathematical profile in 2005. Since then she studies mathematics and computer studies in the Faculty of Mathematics, Informatics and Mechanics at the University of Warsaw (Poland).
Her email is:
`anna.kulka@students.mimuw.edu.pl`

FLOREON - SYSTEM FOR FLOOD PREDICTION

Ivo Vondrák, Jan Martinovič, Jan Kožusznik, Jan Unucka, Svatopluk Štolfa
Department of Computer Science, VSB-Technical University of Ostrava
17. listopadu 15, 708 33 Ostrava-Poruba
Czech Republic
{ivo.vondrak, jan.martinovic, jan.kozusznik, jan.unucka, svatopluk.stolfa}@vsb.cz

KEYWORDS

Flood Prediction, Flood Simulation, Disaster Management.

ABSTRACT

The main goal of our system is to provide the end user with information about an approaching disaster. The concept is to ensure information access to adequate data for all potential users, including citizens, local mayors, governments, and specialists, within one system. It is obvious that there is a knowledge gap between the lay user and specialist. Therefore, the system must be able to provide this information in a simple format for the less informed user while providing more complete information with computation adjustment and parameterization options to more qualified users.

One system feature in high demand is the ability to display reliable and understandable graphical and textual information. Information for various types of users must be adapted to a desired format which is understandable to a particular group of people. For example, a specialist can ask for all available results from different simulation models in text format. This type of information may be useless, however, to the user who only wants to find out whether or not his house will be flooded.

Another important feature is the open structure and modular architecture that enables the usage of different modules. Modules can contain different functions, alternative simulations or additional features. Since the architectural structure is open, modules can be combined in any way to achieve any desired function in the system.

INTRODUCTION

There are many types of natural disasters in the world. Many of which depend specifically upon geography. The flood is one of the worst and most recurrent types of natural disasters in our region [3, 4, 18]. Local governments require reliable models for flood simulations and predictions to save on ample funding that must be otherwise invested in post-flood repairs for impacted regions [4]. Therefore, the issue of flood prediction and simulation has been selected as a case choice for experimental development. This sophisticated system should combine all knowledge from existing models and systems while providing results to a wide range of people that are not experts in this particular field. The name of the system that is being developed is FLOREON (**FLO**ods **RE**cognition on the **N**et).

A wide range of professionals in various fields are already familiar with this system concept. Systems that are able to model floods and present them to qualified system users already exist. The concept of using single, preferred and utilized software (or the combination of several types) plays a major role regarding the limited experience and viewpoints of some specialists [3, 18]. Furthermore, strict deadlines, resulting from the demand for promptly obtained modeling results to support timely decision making, present another significant issue [3]. In addition, the individual, qualified user is not able to operate more models and assemble predictions at the same time. For these reasons, the goal of this system is to combine the results of all prediction and simulation systems and provide simulation results to people who are less informed concerning the complexity of this issue. These results should be accessible to a wide range of users including professionals from various fields. Inhabitants of flood plains should be able to view the results directly in a simple way while more complex information should be accessible to local representatives and rescue teams. Legal bodies responsible for entire regions (i.e. local governments) should be able to view as much information as possible. Thus, the front end of the entire system must be able to satisfy all types of users. A logical solution, and genuinely efficient method of achieving this goal, would be to combine several visualization tools and equipment in the system. All users would be able to access the system from one website. Local representatives and rescue teams can also use other types of thin equipment such as local visualization applications. Governments will be able to use more sophisticated thin or fat equipment that will enable access and visualization of all types of information, based on their needs.

The main concern of our project is to prove that a combination of the latest information technology, along with results from fields that are in some way connected to creating flood simulations and predictions, would be very useful. Fields of interest considered detrimental to supplying theses results include hydrology, meteorology, geoinformatics, remote sensing, integration of prediction models, development of new models, and others [3, 18]. The information system itself is focused on the development of a reliable architecture and a graphic user interface (2D, 3D) that should satisfy the needs of all types of users expected to eventually interact with the system.

The technological integration of the necessary system module is provided by means of internet technology which enables dynamic configuration and the construction of an open architecture [4]. We base the communication of modules within the system and communication with the other systems on Web Services [19].

Although, the primary goal of this system is flood prediction, simulation and presentation, it is possible to expand this system to include modules that can represent a wide range of other disasters. Examples of some modules that may be added to the system include modules that simulate and predict air and water pollution, advective-dispersion processes, forest fire spreading, and etc…

The addition and removal of system modules (including alternative models for simulations and predictions) is based on the plug and play principle. All of these system futures should establish the base for a complex and unique system that enables users to tap into its potential for observing and controlling various types of disasters. Every type of model (rainfall-runoff, hydrodynamic etc…) can be replaced by other models. Reasons for considering such substations include the dropout of hydrometeorological data input for a particular model method (i.e. snowmelt module) and the added issue of method limitations during specific rainfall-runoff episodes – i.e SCS CN method performance differs from SMA method performance during rainfall-runoff events of varying duration and rainfall intensity.
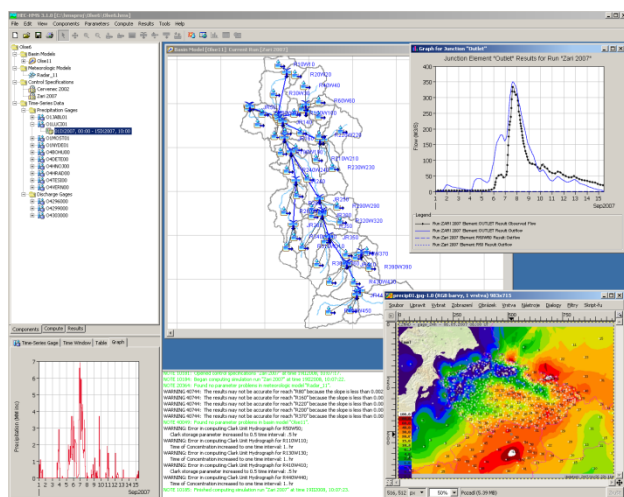
Fig. 1: Rainfall-runoff calibration episode 9/2007 and the result of the rainfall-runoff HEC-HMS simulation model

MATHEMATICAL MODELING

The input data for such a system is subdivided into two groups. The first type of data is the spatial data required for model development (catchment schematization, river channels schematization, etc...). This type of input data is mainly obtained via GIS preprocessing. Time

series (Hydrometeorological data) is the second type of data. Both types have their own spatial and temporal variations. The basic character of the modeled domain (catchment size, geometry and topology) is constant during the modeled rainfall-runoff episode. Another type of spatial data with a higher level of variability is the antecedent precipitation index (API). Time series and hydrometeorological data vary both spatially and temporarily (i.e. measured and predicted precipitation and temperatures). The general computation scheme of the rainfall-runoff and hydrodynamic modeling task is as follows [3, 18]:

1. input data pre-processing and storage
2. rainfall-runoff modeling (discharge hydrograph forecast)
3. hydrodynamic modeling (water level forecast)
4. post-processing of the results (hydrographs, floodplain delineation, etc...).

Precipitation, both measured and predicted, represents the most significant time series data input for such basic computation schemes and cases. There is also a radar data utilization option in the FLOREON system.

Motivation Scenarios

At the beginning of the development of all information systems, interaction scenarios between the system and user must be defined. In our case, two main scenarios were selected at the beginning of our project. These two scenarios are:

- Simple communication between the system and user – users can see the actual situation, simulated prediction, or historical data about floods.
- Coordination of the result generation – for example, if the data about rainfall intensity and prediction represent the system input, the result of the flood simulation must be generated by a computation chain.

These two scenarios, as well as other derived or future scenarios, must be supported by the system's architecture.

Two basic scenarios have been implemented. The first first scenario presents a computation of hydrographs in the rainfall-runoff model and flood areas (floodlake) in the hydrodynamic model (Fig. 2). A computation chain is started as soon as the hydrometeorology data is available – precipitations, temperatures and measured discharge volumes in our case. Data frequency is dependant upon the provider and is available every 6 hours in an FTP service provider. Our module downloads it into our data store and starts the computation chain. This data is then processed by a software module that performs the computation – the hydrologic module. After the module finishes the computation, obtained results - discharge volumes and floodlakes - are uploaded back to the data store. An

hydrologic module provides a simulation for up to 2 days.

The second scenario involves user interaction with a Floreon graphic user interface (GUI) accessible with a web browser (Fig. 3). The user can define areas, layers – ortophotomaps, growth, buildings, motorways, rivers, foodlakes, etc... – and the necessary timeframe for the construction of a result picture. These attributes can also be changed, interactively, by the user.
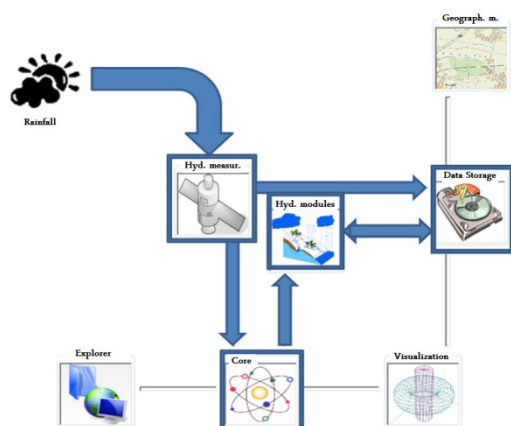

Fig. 2: First scenario – hydrograph computation in the rainfall-runoff model and flood areas in a hydrodynamic model.
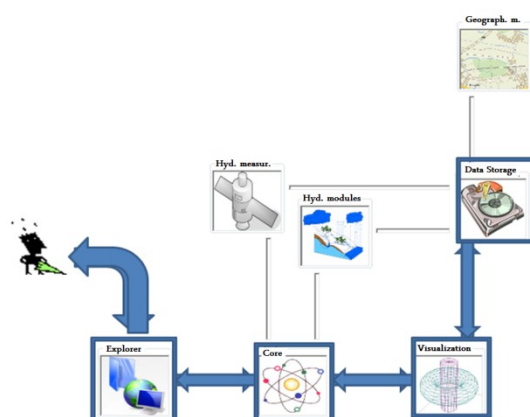
Fig. 3: Second scenario – user interaction with graphic user interface.

ARCHITECTURE

The system architecture that has been developed for this project fulfills all desired features and enables any extension that might be added in the future. Moreover, our architecture includes an option for changing the technology that implements the architecture because the structure of the architecture itself is completely independent and does not rely on implementation.

The main modules of the present system are the core, data warehouse, meteorological data module, hydrology module, geographic module, alternate numeric method module, visualization module, and web server module. The system is mainly based on a Microsoft .NET framework and written in C#. Other languages, such as C++ for special functions, are used when necessary.

All these modules must be connected and communicate in some way. Web Services seem to be the best technology currently available. Web Service technology ensures implementation independence, simple maintenance of connections, standardized communication, and simple module exchange. This technology enables the construction of a flexible and independent system.

Coordinator

A core module is responsible for the coordination of system scenarios and controls all modules that are plugged into the system.

Database Storage

A data warehouse stores all necessary data provided by a wide range of sources (geological data, meteorological data, etc…) or serves as a proxy to remote resources. The following technology is being used:

- PostgreSql database and its spatial extension PostGis
- Microsoft SQL Server
- Raster Provider - proprietary solution that will be soon published as an open source project [8].

Hydrometeorology

A meteorological module is a provider of meteorological data and weather predictions. We use a variety of sources:

- CHMU – Czech Hydrometeorological Institute provides measurement of hydrometeorological data (precipitation, temperature, water level stages, discharges, snow cover) obtained both from the station net and remote sensing methods (e.g. radar estimations of precipitation rates) [3]
- ALADIN – numerical weather forecasting system engaged by CHMU [1]
- Povodi Odry - company established by the government to provide measured discharge volumes in rivers and information about reservoir water levels and operations [15].

The hydrology module is mainly responsible for the computation of floods. More independent applications are used in our computation chain. These programs are connected to provide fully automated options. They involve the following applications:

- HEC-HMS (USACE) –Rainfall-runoff model
- MIKE 11 (DHI) – 1D hydrodynamic model
- MIKE 11 GIS (DHI) – GIS post-processing software, which inspects and provides measures of the flood occurrence and extent in the terrain

- ArcGIS (ESRI) – another platform for the post-processing of results

Additional information about the involved models is available on the internet or in [3].

An alternative, numeric method module is an experimental module that encapsulates new flood prediction methods developed by the group of mathematicians that we are cooperating on this project with. There is a 1D numerical approach as well as a more sophisticated 2D approach. The 2D numerical approach is based on the Finite volume method [2].

Geography and Visualization

The geographical module is responsible for the provision of all actual terrain data, maps, etc... It is a group of standalone tools used for data pre and post-processing:

- SharpMap for shapefile parsing [11]
- Quantum GIS [9] for data import and management

This module also combines all the information stored and created by the system and prepares this information for user presentation. This module uses 2D and 3D techniques to display results.

Web User Interface

A web server module communicates with users and transfers the visualization data to the user. The user can be equipped with any type of device. This module adapts the information to the user's device type.

RESULTS

We have already implemented the two scenarios described above. The system fully computes and predicts floods automatically and periodically. Results are accessible on the Internet with any modern browser. The user only needs to select a specific area and input the date of a given situation. The date can be set from the past or for the future. The system returns pre-computed and stored results.

A GUI demonstration, as well as a computed floodlake, is displayed in Fig. 5. Other layers such as buildings, roads, river channels or orthophoto maps, are displayed behind the floodlake. The additional layers are important for fast and precise orientation during a flood situation (i.e. damaged buildings visible in the picture).

This view represents a rather small area with high zoom. The application also provides a less detailed overview of an inspected area. The view also contains an expressively smaller amount of information, because it could be quite messy to draw a larger scale of information with a lower grade zoom.

Created Applications and Modules

- Web client based on Web 2.0 technology. Average size of transferred data between server and client is 50 KB.
- 3D client also for stereo projection
- Mobile client which identifies location with GPS and actual situations downloaded from FLOREON
- Desktop client based on vector graphics
- Application of import model schematization to FLOREON
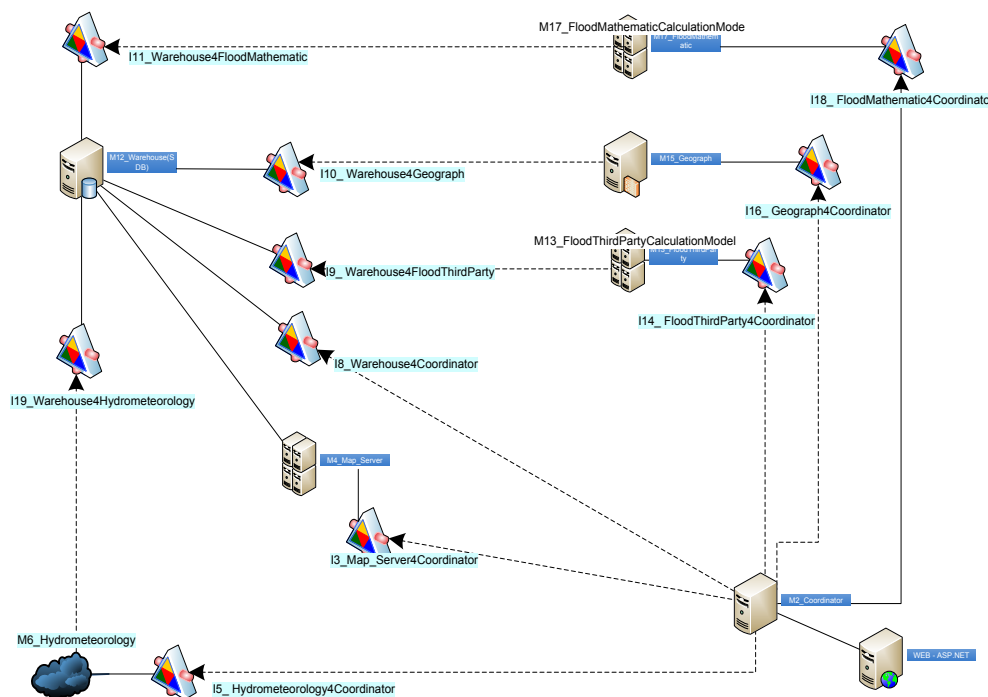
Fig. 4: System architecture

- 3D library for generation 3D maps as images
- RasterProvider for work with raster data and for storing terrain data
- Interface to FLOREON based on Web Services
- MATLAB libraries for model computation
- others

Modeled Regions

The western Carpathian catchment of the Olse/Olza river (2-03-03) was selected as the pilot area. The upper part of this catchment has a mountainous character and the lower parts have been significantly affected by human activity (e.g. coal mining). The Olse catchment has been damaged by significant flooding twice within the past two years. Both floods were identified as the flash flood type. The confluence of Olse, and its tributary Stonavka, is the pilot area for hydrodynamic modeling. Thalwegs, a cross-section and detailed LIDAR digital terrain model was accessible and was further developed and reconstructed for FLOREON modeling cascade requirements.

TEAM PARTICIPANTS

Our development team is composed of professionals from several fields and divided into various departments based on individual modules. The Web, visualization, coordination, data storage and metrological modules are made up of a group of employees and Ph.D. students from our "Department of Computer Science" [12]. The group also realizes the integration of applications in the hydrology module.
Data gathering and pre-processing in the geographical module is carried out by the "Institute of Geoinformatics" [13].

Employees of "The Institute of Geological Engineering" [14] are responsible for the hydrological model construction.

The mathematical models are developed by the "Department of Applied Mathematics" [15] in cooperation with University of West Bohemia in Pilsen [16].
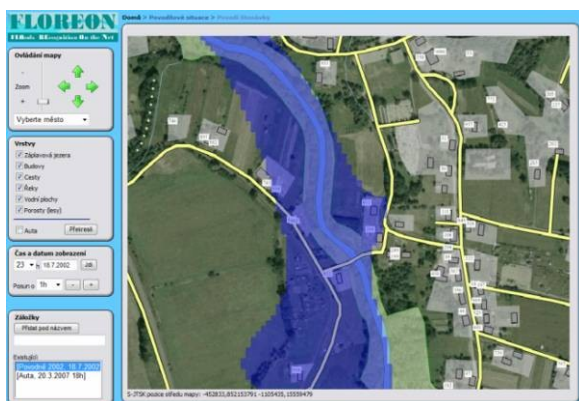
Fig. 5: Flood demonstration

FUTURE WORK

The main goal of the informatics group of this project is the systematic development of the system's architecture. The continual enhancement of 2D and 3D visualization to provide optimal visualized information to the user, based on an open source and commercial technology, is another significant goal for this group.
Of further relevance is the issue of system coordination and communication with external subjects. The main concern is the usage of a standardized interface and security.
An information system is practically useless without current data. Therefore, the data for the simulations and predictions must be systematically and periodically updated. Some of this data is stored in a data warehouse – schematizations of river basins, terrains etc… Data source coordination, data selection and updating are long term tasks that must be carefully planned.
Ensemble simulations and predictions based mainly on the assumption of the input meteorological data uncertainty and fuzziness is another advance planned in our system. Information about the "optimistic" and more "catastrophic" variant of predictions for qualified users can be provided as a sequence of hydrographs or buffered zones with a statistical assessment. The offer of ensembles based on various methods of rainfall-runoff and hydrodynamic solutions implemented in various types of models is yet the another important point of view.
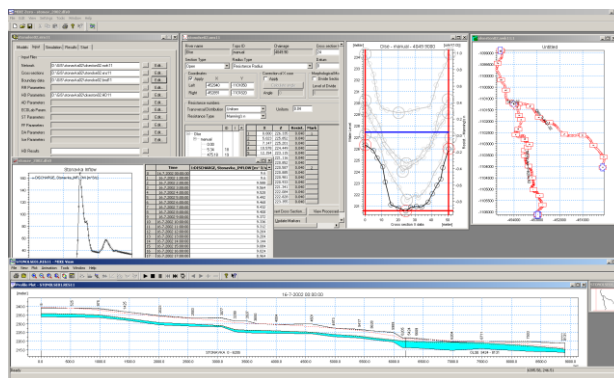
Fig. 6: Hydrodynamic model MIKE 11 GUI and the result of the model simulation (water levels in thalweg)

CONCLUSION

In our opinion, and based on the response of external subjects specializing in the field of flood prediction or the coordination of emergency units, this project is thoroughly useful. This system can be easily extended or connected to other systems and establish a portal for disaster control, management, and presentation. This project seems to be innovative in its unique connection and coordination of many results from many different disciplines.
The architecture of this system is completely open and general. Therefore, all future features can be easily added to this system without negatively affecting its present functionality. Even the implementation of the

architecture itself can be rebuilt without significant malfunctions in the entire system.

ACKNOWLEDGEMENT

REFERENCES

[1] ALADIN - Aire Limitée, Adaptation Dynamique, Development International, http://www.chmi.cz/meteo/ov/aladin/index.php

[2] Barth T, Ohlberger M, *Finite volume methods: Foundation and analysis.* Volume 1, chapter 15 in Encyclopedia of computational Mechanics, editors: Stein, E.; Borst, R., Hughes, T. J. R. West sussex, England: John Wiley & Sons http://citeseer.ist.psu.edu/barth04finite.html

[3] BEDIENT, P.B., HUBER, W.C. et VIEUX, B.C. (2007): *Hydrology and floodplain analysis*. 4th edition. Prentice Hall, London, 795 s., ISBN: 978-0131745896

[4] BRÁZDIL, R. et al. (2005): *Historické a současné povodně v České republice.* Brno-Praha. MU Brno a ČHMÚ, 369 p. ISBN 80-210-3864-0

[5] Czech hydrometeorological institute, http://www.chmi.cz/indexe.html

[6] Günter Oliver, *Data Management in Enviromental Information Systems*, Handbook of massive data sets, Dortrecht, The Netherlands : Kluwer academic publishers, 2007, 981 – 1091, ISBN 1-4020-0489-3

[7] Institute of Computer Science, Academy of Sciences of the Czech Republic, http://www.cs.cas.cz/

[8] MEDARD, http://www.medard-online.cz/

[9] QGIS – Quantum GIS, http://www.qgis.org/

[10] Raster provider project, http://www.codeplex.com/rasterprovider

[11] SharpMap project – Open source mapping engine for .NET 2.0, http://www.codeplex.com/SharpMap

[12] :URL: http://www.cs.vsb.cz/en/index.php (2007-06-30)

[13] URL: http://gis.vsb.cz/gisen2006/ (2007-06-30)

[14] URL: http://www.hgf.vsb.cz/hgf/eng/kat541_eng .html (2007-06-30)

[15] URL: http://www.am.vsb.cz/index.php?lang=en (2007-06-30)

[16] URL: http://www.zcu.cz/index-en.html (2007-06-30)

[17] URL: http://www.pod.cz/ (2007-06-30)

[18] WOHL, E. W. ED. (2000): *Inland Flood Hazards*. Cambridge,Cambridge University Press. 498 p. ISBN 0-521-62419-3,.

[19] Damien Foggon, Dan Maharry, Chris Ullman: Programming Microsoft .NET XML Web Services (Pro-Developer) (Hardcover), Microsoft Press

Reliability Factors

Jan Amborski

Institute of Aviation
Warsaw, Poland
j_amborski@yahoo.com

ABSTRACT

There are two ways of exploitation of an aircraft. The first one is called Hard Time exploitation, the second one is called On Condition exploitation. The only difference is what we have to do after in specified advance time. In first case you have to replace old part by a new one. It doesn't matter if the old one is still working, if it looks nice, it also doesn't matter if you can see any wear. You just have to replace it. Reliability analysis is analysing observed factors to predict maintenance before aircraft defect appear in the air, but depending on statistics which are continuously updated by every operator. The analysis of this statistics use some factors which answer questions important for operator, maintenance department, crew department and so on. The answers comes from statistics based simulation. After one simulates when some component will failed, one can replace the component by new one before failure.

INTRODUCTION

Reliability engineering and maintenance are the source of safety. They are one of the most important aims in modern aviation. The modern reliability engineering is also the way of seeing safety as exploitation costs reduction. Following good written reliability programs airplanes earning money as long as possible, as often as possible. They are not spending time waiting for parts, which were unexpectedly broken, they are reaching destinations on time. The customers are glad because of accuracy and reliability. How to reach this aim?

PROBLEM FORMULATION

There are two ways of exploitation of an aircraft. The first one is called Hard Time exploitation, the second one is called On Condition exploitation. The only difference is what we have to do after in specified advance time. In first case you have to replace old part by new one. It doesn't matter if the old one is still working, if it looks nice, it also doesn't matter if you can see any wear. You just have to replace it. Reliability analysis is analysing observed factors to predict maintenance before aircraft defect appear in the air, but depending on statistics which are continuously updated by every operator. The analysis of this statistics use some factors which answer questions important for operator, maintenance department, crew department and so on. The answers comes from statistics based simulation. After one simulates when some component will failed, one can replace the component by new one before failure.

Those questions are not only important for economic reasons, but also - or first - for safety reason, that is why it results from reliability analysis. The Reliability Report is expected by aviation authorities from every operator. For European countries authority is the EASA (European Aviation Safety Agency) and for USA it is the FAA (Federal Aviation Agency). Reports in most cases reach authorities every month, for every aircraft, from every operator. Reports differ between each other depending on type of aircraft, number of aircrafts operated in company, number

of hours flown, tape of missions and so on. What form and kind of data will appear in report is written in Reliability Programs accepted by aviation authorities and created by operators, but in every report there are such data as hours flown by each aircraft, number of landings, number of defects and accidents.

THE ATA 100

All of reported defects have to be divided using ATA code. The Ata code is the Component Code Table DEVELOPED by The Air Transport Association of America (ATA), and called Specification 100 code. Now it is used By FAA and accepted by EASA. Such fragmentation is useful when you want to know what part of the airplane, what system or what component most common makes problems, where you have to look more often. The main system is as follows:

21 Air conditioning
22 Auto Flight
23 Communications
24 Electrical Power
25 Equipment Furnishing
26 Fire Protection
27 Flight Controls
28 Fuel
29 Hydraulic Power
30 Ice & Rain
31 Indicating & Record
32 Landing Gear
33 Lights
34 Navigation
35 Oxygen
36 Pneumatic
37 Vacuum
38 Water / Waste
45 Central Maintenance
49 Auxiliary Power
51 Structures
52 Doors
53 Fuselage
54 Nacelles / Pylons
55 Stabilizers
56 Windows
57 Wings
71 Power plant
73 Engine Fuel Control
74 Ignition
76 Engine Controls
78 Exhaust
79 Oil
80 Starting

As it can be seen there are only some chosen chapters, and in every chapter there are also subchapters. The system is independent on aircraft type so if some operator frequently have to change parts in its different type of aircraft's, so with different Part Numbers it is simple to check if, for instance there is permanent problem with landing gear system. Then one can ask if it is a maintenance problem or exploitation or conditions problem. But this question is to answer using system other then ATA 100.

RELIABILITY FACTORS

The reliability factors can be divided in 3 groups. The first group contains factors describing whole aircraft and people who are somehow connected with the aircraft, pilots, management, and maintenance. In this group one can find such factors as:

- PRR (*Pilot's Reports Rate*) Factor

This factor shows how many defects or problems were reported/found by pilots. Some of defects appeared during the flight but there is also group of cases, where the problems were found by pilots because they were not found by mechanics. The parameter is given as cases for 1000 flights.

- PRMR (*Pilot's Reports/Maintenance Report*) Factor

This factor show how many defects or problems were reported/found by pilots in compare to defects found by maintenance departments.

- DR (*Dispatch Reliability*) Factor

This factor analyses technical reason influence on dispatch delays/cancelations compare to all delays/cancelations.

- SR (*Shop Rate*) Factor

Factor shows amount of time when aircraft spend on unplanned maintenance every 100 flight hours.

- TF (*Technical Functionality*) Factor

The TF Factor is number of hours flown with accepted defects and flight limitations (Flight with Minimum Equipment List Regulation, and Hold Item List Document usage) for every 100 flight hours.

- TI (*Technical Incidents*) Factor

This factor is calculated for 1000 flights. The deference is because of calculated case. It is a

serious defect with influence on flight safety, which should be report in separate report to authorities.

The second group contains factors, which more precisely describe aircraft and its components. Every of factor below is calculated separately for each ATA 100 chapter or, in case of number of defects/removals/shop time even subchapters. In this group one can find following factors:

- RR (*Removals Rate*) Factor

The RR factor show number of removals of exact component from an aircraft, independently of reason for this removal for every 1000 cycles or working hours.

- FR (*Failure Rate*) Factor

The FR factor shows number of removals of exact component from an aircraft, because of failure for every 1000 cycles or working hours

- MTBR (*Mean Time Between Removals*) Factor

The MTBR factor, as the name says show average time between removals of exact component from an aircraft for every 1000 cycles or working hours.

- MTBF (*Mean Time Between Failure*) Factor

The MTBR factor, as the name says show average time between removals of exact component from an aircraft, because of failure for every 1000 cycles or working hours

- MTBUR (*Mean Time Between Unscheduled Removals*) Factor

The MTBR factor, as the name says show average time between unplanned removals of exact component from an aircraft for every 1000 cycles or working hours.

The third group of factors is the group of factors combined with power plants. Those very important aircrafts components have separate factors.

- IFSD (*In-Flight Shut Down Rate*) Factor

Factor describing number of In flight shut down of engines or other serious problems with engines during flight. Because the case is not very common the factor is calculated for 1000 flight hours.

- URR (*Unscheduled Removals Rate*) Factor

This factor is unscheduled engines removals from an aircraft rate. The factor differs from previous one, because removal can be made also after defect find on ground. The factor is calculated for 1000 flight hours.

- SVR (*Shop Visit Rate*) Factor

This is a parameter showing number of engine repairs in a shop for 1000 flight hours.

CONCLUSIONS

Lot of the presented factors is similar and the choice which is more useful for an operator or airlines depends on operator its self and authorities which have to accept the Reliability Program. Factors can also be created especially for operators needs or problems they think are more common. But the most important thing is to analyze the changes of calculated factors during time, after each probe and compare it to reliability levels predicted by previous years of flying on analyzed aircraft and results for rest aircraft the same type which are flying all over the world, whole manufactured fleet. The simulation is as proper as large is the number of cases the statistics data used for this simulation is based on.

REFERENCES

1. Jerzy Lewitowicz *Podstawy Eksploatacji Statków Powietrznych – Statek Powietrzny i Elementy Teorii*, Wydawnictwo Instytutu Technicznego Wojsk Lotniczych, Warszawa 2001

2. Jerzy Lewitowicz, Kamila Kustroń *Podstawy Eksploatacji Statków Powietrznych – Własności i Właściwości Eksploatacyjne Statku Powietrznego,* Wydawnictwo Instytutu Technicznego Wojsk Lotniczych, Warszawa 2003

3. Jerzy Lewitowicz *Podstawy Eksploatacji Statków Powietrznych – Systemy Eksploatacji Statków Powietrznych*, Wydawnictwo Instytutu Technicznego Wojsk Lotniczych, Warszawa 2006

4. Reliability Programs of deferent Airlines and Aircraft Operators.

5. Appendix 1 (part M) to implementing Rules EC 2042/2003 dated November 20/2003

BIOGRAPHY

Jan Amborski born in 1972 received his M.Sc. in Engineering in 1999 from Warsaw University of Technology, Faculty of Power and Aeronautical Engineering and 2006 PhD from Warsaw University of Technology, Faculty of Automotive and Construction Machinery Engineering. Now he is working at Warsaw Institute of Aviation.

Discrete Event Modelling and Simulation in Logistics, Transport and Supply Chains

SIMULATION-BASED MEASUREMENT OF SUPPLY CHAIN RISKS

Ruslan Klimov and Yuri Merkuryev
Department of Modelling and Simulation
Riga Technical University
1 Kalku Street, Riga LV – 1658, Latvia
E-mail: ruslan@itl.rtu.lv

ABSTRACT

For today's supply chains, risk management becomes an important instrument, which can protect a system from sudden collapses. Still, the meaning of the supply chain risk and techniques for its evaluation and mitigation still continue to be extensively researched. In this paper, simulation is used for risk management in supply chains. Within the discussed research, the supply chain risk is defined as possible disruptions that can affect the supply chain ability to function normally. Due to that, system reliability parameters are taken as a basis for risk evaluation. The difficulties of making the corresponding assessments mathematically are argued because of supply systems complexity. Then, simulation is applied as an effective tool for risk evaluation and management of the predefined supply chain. At the same time, the reliability theory mathematics is used for the created simulation model validation.

INTRODUCTION

The supply chains of today represent complex systems with wide profit increase opportunities and potentially high risks level. At the same time, the meaning and the concept of supply chain risk are not strongly defined. Most researches on this subject observe risks connected with separate disruptions within discrete supply chain members. Then, supply chain management influence on particular risks values is considered. For instance, such risks can be prescribed as deficiency in inventory, forecast failures, manufacturing quality failures, etc. Those risks are well known to managers in the sphere of logistics. A more detailed discussion of particular risks is presented in (Klimov and Merkuryev 2007). The obvious disadvantage of such supply chain risk management is difficulties in recognition, how particular risks affect other supply chain members. Thus, with such perception managers can at best evaluate failure rates for a single supply chain member, but not for the whole system. On the other hand, the conception of supply chain organization assumes management based on the whole system coordination. Thus, risk management should be directed to increase the reliability of the whole system. For example, within the management process, logistic specialists might be interested to know: what real reliability of supply system is and how many critical failures occur during a chosen period of time; which suppliers are vital and should be provided by particular detailed management; how the chosen supply chain can be reconstructed in order to reduce critical failure risks, etc.

So, appropriate answers can provide supply chain managers with necessary decision support information for further supply system development, reorganization and supply chain risk monitoring. Within the current research the meaning of supply chain risk is accepted as failures, which lead supply systems to the so-called critical or emergency states that imply unavailability to fulfil main function of the supply chain system - satisfying end customer need and profit for supply chain members. Thus, within our research, it is advised to use reliability theory aspects with the aim to create a simulation model, which will be used for further supply chain risk management.

RELIABILITY EVALUATION OF "NON-REPAIRABLE" SUPPLY CHAIN

Although pure mathematical approaches are hardly adoptable for supply chain reliability estimating, it is difficult to overstate the importance of the mathematical model development. Here, reliability theory basic aspects are considered with the aim to evaluate reliability parameters for a supply chain with various simplifications. An analogical simulation model is created and validated by mathematically estimated values. Then, a more complicated reliability simulation is performed on the basis of validation results.

Supply System

Within the current research, the supply chain is defined as a network of organizations involved in managing material, information and financial flows, that is necessary for realization of basic logistics operations (manufacturing, storing, transportation etc.), aimed to furnish the end customer with needed products or service produced by those organizations. In the paper the simplified supply chain is considered, whose network structure is shown in Figure 1a. It consists of six main and two substitution supply chain members. The first two suppliers provide the system with raw materials of two different types. Then, the remaining relevant supply chain members assemble a final product and deliver it to customers. The substitution supplier (Supplier 3) can deliver both types of material directly to the assembly if necessary. The usage of substitution

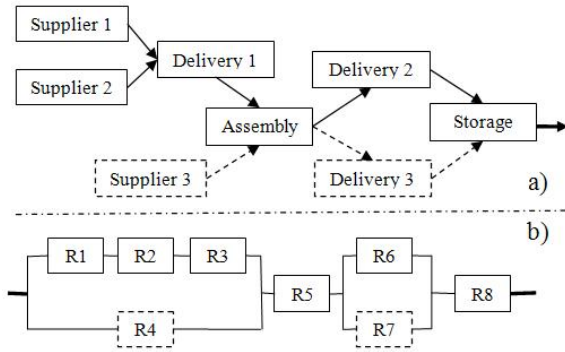supplier and substitution transportation intermediary (Delivery 3) increases supply chain total costs.

Figure 1: The Network of Simple Supply Chain

In this section, the supply chain risk is treated as particular (or several) supply chain member failures that leads the whole system to an unworkable state. Accordingly, system reliability parameters are studied as a basis for such risks evaluation.

System Reliability and Assumptions

Over time the system under study may have changed from one conditional state to another. Still, within this section, the supply chain is defined as a "non-repairable" system.
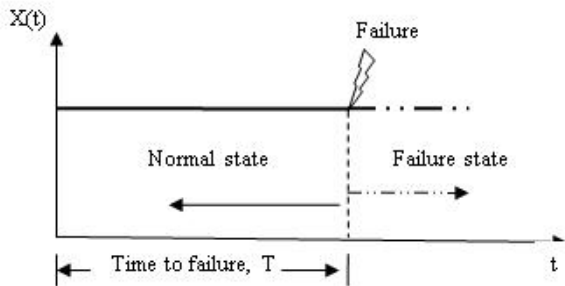
Figure 2: Reliability of "Non-Repairable" System

Thus, there are only two conditional states: failure and functioning (or normal) states.

$$X(t) = \begin{cases} 1, & system\ is\ functioning\ at\ time\ t \\ 0, & system\ is\ not\ functioning\ at\ time\ t \end{cases} \quad (1)$$

Two global assumptions are defined for this paper. Thirst, it is argued that studying a supply chain and its reliability, a "burn-in" and a "wear-out" period can be not taken into account. It is also suggested that supply chain managers can predict failure and recovery rates (in cases of failures) for a particular supply chain member and at the same time those values are fully independent. The last assumption is that time failure (and recovery) functions can be described by exponential distribution, as it is the most commonly used life distribution for technical systems reliability

analysis (Andrews and Moss 2002; Rausand and Hoyland 2004).
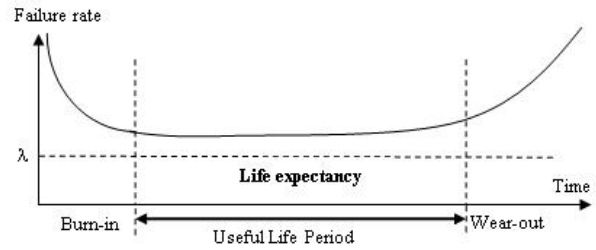
Figure 3: The "Bathtub" Curve

In general, the reliability of a supply chain can be defined as a probability that system will not fail before the predicted time moment T:

$$R(t) = Pr(T \geq t) = \int_t^\infty f(u)du = e^{-\lambda t} \quad (2)$$

In the same way, from a known probability distribution, it is possible to define the mean time to failure (MTTF):

$$MTTF = \int_0^\infty R(t)dt = \int_0^\infty e^{-\lambda t}dt = \frac{1}{\lambda} \quad (3)$$

The above mathematics is usually used for describing a single component's (in most cases, a mechanical or electronic equipment) reliability (Jastrebeneckij and Ivanova 1989). However, supply chain system consists of several separate organizations; hence, a structural analysis should be used for further mathematical evaluations of the whole system reliability parameters.

Structural Analysis of a Supply Chain

As stated, it is assumed that the data about failure rates in particular supply chain organizations are known. Then, reliability or functionality for whole supply chain system can be evaluated using structural analysis methods. There are several techniques for reliability estimation within complex systems. Here "Combinations of simple series and parallel structures" and "Minimal path sets" methods are used for reorganizing the supply network into one complex node, whose reliability parameters can be calculated with similar equations (2, 3).

Combination of Simple Series and Parallel Structures
The redrawn network of the defined supply system is presented above (see Figure 1b). When a network consists of only combinations of series and parallel structures, its analysis can be carried out in stages. Each stage simplifies the network by combining series and parallel sections. By using equations for series and parallel combinations, network sections can be obtained at each stage. The reduction process continues until one "super-component" remains, which links the start and end component (Rausand and Hoyland 2004). The performance of the studied system is identical to the developed "super-component" (see Figure 4).
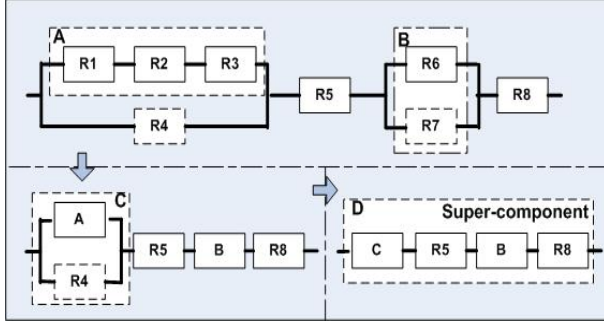
Figure 4: System Reliability Network

Equations for calculating reliabilities of simple series and parallel networks of n components with reliabilities R_i are the following:

$$R_{series} = \prod_{i=1}^{n} R_i \qquad (4)$$

$$R_{parallel} = 1 - \prod_{i=1}^{n}(1 - R_i) \qquad (5)$$

The accumulative formula for the whole supply system reliability calculation is shown below.

$$R_{sys} = \left(1 - (1 - R_1 R_2 R_3)(1 - R_4)\right)R_5\left(1 - (1 - R_6)(1 - R_7)\right)R_8 \qquad (6)$$

$$R_{Sys}(t) = \left(1 - \left(1 - e^{-\lambda_1 t}e^{-\lambda_2 t}e^{-\lambda_3 t}\right)\left(1 - e^{-\lambda_4 t}\right)\right) * e^{-\lambda_5 t} * \left(1 - \left(1 - e^{-\lambda_6 t}\right)\left(1 - e^{-\lambda_7 t}\right)\right) * e^{-\lambda_8 t} \qquad (7)$$

The corresponding evaluated parameters are provided in Table 3.

Minimal Path Sets
Unfortunately, the above method is hardly adoptable for real systems. Thus, in practice, more universal methods are used. The "minimal path" method has a wider range of applications and it is adoptable for systems, which do not consist only of combinations of series and parallel structures. Still, it has a disadvantage; it only provides an approximation of reliability values and this method becomes useless in cases when systems components have low reliability rates (Andrews and Moss 2002). The "minimal path" set (which is part of total component set), ensures that the system is functioning. In Table 1 the "minimal path" set for the predefined supply chain is provided.

Table 1: "Minimal Path" Set

Path name	Path structure
Total path set	{R1;R2;R3;R4;R5;R6;R7;R8}, {R1;R2;R3;R4;R5;R6;R8}, {R1;R2;R3;R5;R6;R7;R8}, {R1;R2;R3;R5;R6;R8}, ..., {R4;R5;R7;R8}.
"Minimal path" set	{R1;R2;R3;R5;R6;R8}, {R1,R2,R3,R5,R7,R8}, {R4;R5;R6;R8}, {R4;R5;R7;R8}

According to the "minimal path" method, the following equations should be used for the network structure function evaluation:

$$R_{sys}(X) = \coprod_{j=1}^{p} p_j(X) = 1 - \prod_{j=1}^{p}\left(1 - p_j(X)\right) \quad (8)$$

$$p_j(X) = \prod_{i \in P_j} R_i \; for \; i = 1,2,\dots,s \qquad (9)$$

p_i is the minimal path set

Using the above equations, it is possible to define structural function (11), whose approximated value can be used for considering system reliability:

$$Q_{sys} \leq \phi(X) = 1 - (1 - X_1 X_2 X_3 X_5 X_6 X_8) * (1 - X_1 X_2 X_3 X_5 X_7 X_8) * (1 - X_4 X_5 X_6 X_8) * (1 - X_4 X_5 X_7 X_8) \qquad (10)$$

$$Q_{Sys}(t) \leq 1 - \left(1 - e^{-\lambda_1 t}e^{-\lambda_2 t}e^{-\lambda_3 t}e^{-\lambda_5 t}e^{-\lambda_6 t}e^{-\lambda_8 t}\right) * \left(1 - e^{-\lambda_1 t}e^{-\lambda_2 t}e^{-\lambda_3 t}e^{-\lambda_5 t}e^{-\lambda_7 t}e^{-\lambda_8 t}\right) * \left(1 - e^{-\lambda_4 t}e^{-\lambda_5 t}e^{-\lambda_6 t}e^{-\lambda_8 t}\right) * \left(1 - e^{-\lambda_4 t}e^{-\lambda_5 t}e^{-\lambda_7 t}e^{-\lambda_8 t}\right) \qquad (11)$$

Q_{sys:} approximated reliability value

The evaluated reliability characteristics are given in Table 2.

Reliability Simulation

The complexity of mathematical methods application is obvious. Thus, it is suggested that any network, which consists of basic supply chain members (no detailed information within separate nodes is taken into account), can be easily represented by a simulation model. Moreover, it is easy to develop a program for such simulation model automated generation. It gives to potential supply chain managers the opportunity to make simulation-based experiments using different input data: supply chain network structures and particular supply chain nodes failure rates (see Figure 5). Then, executed experiments will provide specialist with statistical data, which can be used for further reliability analysis.
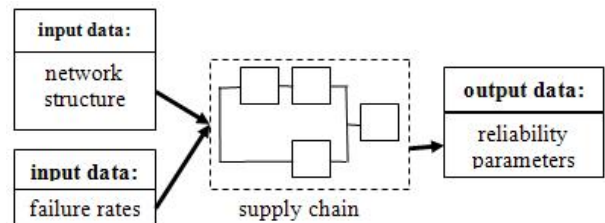

Figure 5: System Reliability Simulation

For our goals ARENA 8.0 simulation language is used. The simulation process of the "non-repairable" system includes three experiments; each of them consists of five hundred replications. The basic simulation time period is one day. The necessary reliability parameters

values are evaluated using statistics gained from those five hundred replications.

The aim of the first experiment is to find the probability that the system will survive in the time period of 730 days. In the second experiment the necessary survival time period is decreased by 200 days. Then, failure rates for separate network components are increased in the third experiment with the goal to make 11th formula more suitable (t=730 days) for system reliability evaluation. The calculated experimental data are shown in Table 3. For all three experiments mean time to failure is calculated, where $MTTF$ is gained from simulation results, but $MTTF_R$ and $MTTF_Q$ are calculated due to R_{sys} and Q_{sys} accordingly.

Table 2: Failure Rates

Node	Failure rates, λ days^{-1}	
	Experiments 1-2	Experiment 3
R1	$3{,}87*10^{-4}$	$1{,}35*10^{-4}$
R2	$3{,}80*10^{-4}$	$1{,}15*10^{-4}$
R3	$3{,}75*10^{-4}$	$0{,}90*10^{-4}$
R4	$2{,}55*10^{-4}$	$0{,}45*10^{-4}$
R5	$2{,}75*10^{-4}$	$0{,}35*10^{-4}$
R6	$2{,}90*10^{-4}$	$1{,}80*10^{-4}$
R7	$2{,}70*10^{-4}$	$1{,}55*10^{-4}$
R8	$1{,}85*10^{-4}$	$0{,}25*10^{-4}$

Table 3: Reliabilities of "Non-Repairable" System

Exp. No.	Mathematical evaluation			
	$R_{sys,}$	Q_{sys}	$MTTF_R$	$MTTF_Q$
	percentages		days	
1	62,4	≤85,	1304	1693
2	72,	≤93,	1304	1693
3	93,	≤99,	5263	5841

Exp. No.	Simulation	
	$R_{sys,,\%}$	$MTTF$, days
1	63,2%	1303
2	71,0%	1303
3	94,4%	5458

By observing the results of simulation-based and mathematics-based evaluations some obvious conclusions can be made. First of all, the method of "minimal path" set, which is more usable for the network structural analysis, can't be used for components with large failure rates; thus, it is useless in most supply chains reliability analysis tasks. On the other hand, the method of simple structures combinations is limited by analysed system network complexity. Besides, mathematical methods are labour intensive enough even for simplified tasks. On the other hand, the simulation results are very close to mathematical evaluation; this gives the opportunity to conclude about further simulation model application to the reliability estimation of more complicated systems.

SUPPLY CHAIN RELIABILITY SIMULATION

In the previous section the simplified system was considered, which gives the opportunity to realize a validation for the created simulation model. Still, the results gained are hardly adoptable in a real life supply chain management. Then, the next step is to extend the system characteristics in order to get a more adequate reliability management tool. Thus, the created simulation model can be restructured with the aim to analyse various supply systems with no limit on a excessive complicity. In addition, simulation can be used for "repairable" or "multiconditional states" systems.

Repairable Supply Chains

Supply chain functioning doesn't end at the moment when the first failure has occurred. In practice, a supply chain can go through multiple disruptions, which stop system functioning only for a certain period of time, within which a necessary repair process is initialized (see Figure 6). Thus, repairable systems more precisely characterize supply chain functioning.
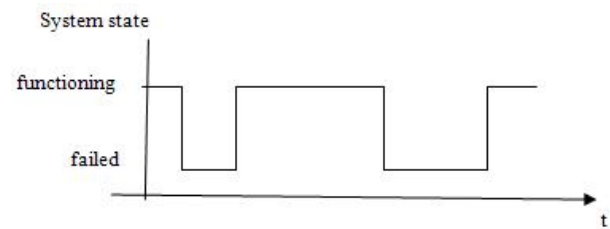
Figure 6: Repairable System Life History

As it is mentioned above, within the current paper repair distributions are described by exponential functions. The repair rates are assumed to be the same for all supply chain members. The corresponding repair rates are shown in Table 4.

Table 4: Repair Rates

Experiment No.	Repair rates, υ days^{-1}
4 and 7	$2{,}5*10^{-2}$
5 and 6	$10{,}5*10^{-1}$

The common parameters, which characterize a repairable system performance and which should be studied within a reliability analysis, are: system availability, system unavailability, failure intensity, expected number of failures, repair intensity, expected number of repairs, etc. Since our research is entirely theoretical and non-realistic input data are used, only a few parameters are presented in the output data Table 5.

Table 5: Results of Experiments 4 and 5

Experiment No.	4	5
Number of failures	520	481
Average time for repairs	40,1 days	8,44 days
Approximate system failure rate	$5,2*10^{-4}$	$4,8*10^{-4}$
Staying in repair state	2,11%	0,42%

So, to perform two additional experiments, the simulation of the repairable supply system is considered. Each experiment consists of one replication. Since the goal of those simulation experiments is to find the frequency of system failures, but not the moment until the first failure had occurred, the time length of replications is extended to a million.

The output data of our simple experiments show self-evident correlation between components repair rates and system functioning characteristics. In practice, simulation can help to determine, which supply chain member repair rate is most important for the whole system normal functioning.

Emergency and Crisis

Previously, the simulation of the system, with only two possible conditional states, was considered. Thus, the supply chain risk has been defined as the likelihood that the system will have changed its state from functioning to unworkable. In practice, however, it is more effective to analyze system conversions within various conditional states, which gives additional management flexibility.

For instance, in the sphere of risk managements, some researchers pose conditional states for a studied system: normal state, emergency and crisis (Borodzicz 2005). Similar various conditional states can be considered in supply chains:

$$X(t) = \begin{cases} N, & normal\ state\ at\ time\ t \\ E, & emergency\ at\ time\ t \\ C, & crisis\ at\ time\ t \end{cases} \quad (11)$$

Normal condition defines supply chain normal functioning that is material, financial and information streams flow through the system within the predefined rules. At the same time some failures of suppliers are possible; for such cases, reserve suppliers or substitution materials are used. Accordingly, the final profit for supply system decreases, thus such functioning is called emergency state. Cases, in which reserve or substitution are not possible, are called crises in the sense that they affect the ability to satisfy end customer. It is self-evident that any state is possible for real supply chain. At the same time, system will always try to reach the normal state (see Figure 7).
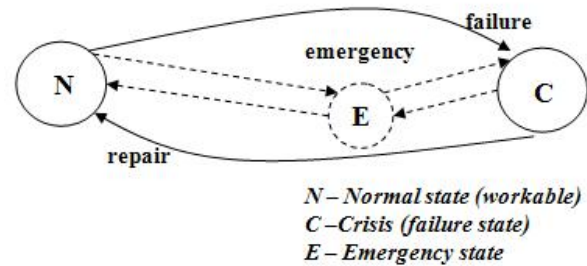
Figure 7: System Processes

Usually the parameters, which characterize changes of a system from one state to another, can be described by Markov processes (Andrews and Moss 2002). Accordingly, an assessment of transition probabilities between different conditional states (so called transition matrix) is needed for further system reliability analysis. Theoretically these probabilistic values can be obtained with the help of mathematical equations in cases when failure and repair rates of each network node are known; but in practice, a task of that kind is almost undoable. Still, simulation uses failure and repair rates as input data, and then the necessary statistical data about transitions can be modelled.

Another goal of the last two experiments is to estimate system functioning characteristics, which provide information about:

- how long a system stays in normal, emergency and crisis states?
- for how long a system stays in each of the states before next transition to another conditional state?

The final results of the last two experiments are presented in Table 6:

Table 6: Results of Experiments 6 and 7

Experiment 6		N	E	C
Time in each state		92,46%	2,11%	5,42%
Average time of staying in state		511 days	41 days	39 days
Rate of transition to state:	N	99,81%	0,05%	0,14%
	E	2,19%	97,64%	0,17%
	C	1,19%	0,03%	98,78%

Experiment 7		N	E	C
Time in each state		98,13%	0,43%	1,43%
Average time of staying in state		548 days	9 days	11 days
Rate of transition to state:	N	99,82%	0,05%	0,13%
	E	10,49%	89,31%	0,20%
	C	4,82%	0,04%	95,14%

The simulation conditions of the last two experiments are similar to the simulation conditions used in Experiments 4 and 5. The only difference exists in the changed output data assembling module, which provides the necessary statistical data about system transition events.

It should be noted that the length of the replication must be large enough to get reliable data. Moreover, the simulation length directly affects the output values of the transition matrix; with a sufficiently long simulation, the transition matrix becomes limiting matrix, which characterizes probabilities that the system will be in a certain conditional state at a certain moment of time. (Andronov et al. 2004). As an alternative, these values can be obtained directly from the simulation experiment output data or can be evaluated mathematically using the primary transition matrix.

CONCLUSIONS AND FUTURE WORK

The goal of this paper is to advise using system reliability parameters in order to analyse supply chain risks and possibilities of its future management. The supply chain risk is defined as possible disruptions that can affect the supply chain ability to function normally. Thus, the system reliability parameters are taken as an approach to risks assessment. As the discussed research has a theoretical nature, the simplified supply chain is considered. At the same time, the assumptions made provide the opportunity to make necessary mathematical evaluations. This mathematics is too time-consuming to be completed without the predefined limitations (for more realistic systems); although performed calculations are valuable, as they can be used for simulation model validation.

The simulation tool is assumed to be a very effective way to analyse system reliability; it can seriously abate the necessary mathematical instrument. At the same time, simulation models are very flexible and can be easily transformed synchronously to changes in a studied supply chain. Experiments demonstrate the efficiency of simulation application within system reliability analysis. Still, it only helps in cases, when necessary input data are available.

The target of further research is simulation-based decision support tool development, which can be used by supply chain managers as an additional aid for risk analysis. It also can be used for studying risk mitigation methods. At the same time, some complementary parallels with mathematical justifications should be made with the purpose to increase the objectiveness of simulation models output results.

ACKNOWLEDGEMENTS

REFERENCES

Andrews J. and Moss T. 2002. *Reliability and Risk Assessment.* Professional Engineering Publishing Limited. 119-122, 165-199.

Andronov, A., Kopitov, E. and Gringlaz, L. 2004. *Probability Theory and Mathematical Statistics.* Piter,. 218-222. (in Russian)

Borodzicz, E. 2005. *Risk, Crisis & Security Management.* John Willey & Sons, Ltd. 6-7.

Jastrebeneckij M. and Ivanova G. 1989. *Automated systems reliability for technical processes management.* Energoatomizdat. 63-79. (in Russian)

Klimov R. and Merkuryev J. 2007. "Supply Chain Risk Model Recognition". In *Proceedings the 4th International Mediterranean Modelling Multiconference* (Claudio Hotel & Congress Center, Bergeggi, Oct. 4-6). Italy, 311-317.

Rausand M. and Hoyland A. 2004. *System Reliability Theory: Models, Statistical Methods, and Applications.* John Willey & Sons, Ltd. 28, 164-169.

AUTHOR BIOGRAPHIES

RUSLAN KLIMOV earned his Master degree in information technology from Riga Technical University in 2005. Since that time he has been taking a researcher position at the Department of Modelling and Simulation of Riga Technical University. Being a member of Latvian Simulation Society, he focuses his professional interests on the application of simulation to risk handling in supply chains.

YURI MERKURYEV is Full Professor and Head of the Department of Modelling and Simulation at the Institute of Information Technology of Riga Technical University (Latvia). His professional interests include the methodology of discrete-event simulation, optimisation of simulation models, supply chain modelling and management, as well as education in the areas of modelling & simulation, and logistics management. He is Programme Director for the Master-level curriculum "Industrial Logistics Management" at Riga Technical University. Prof. Merkuryev regularly participates in organising international conferences in the area of modelling and simulation. He has served as General Chair of the International Conference "European Conference on Modelling and Simulation", ECMS'2005 (Riga, Latvia, June 1-4, 2005). Prof. Merkuryev is a Corresponding member of the Latvian Academy of Sciences, President of Latvian Simulation Society, Board Member of the Federation of European Simulation Societies (EUROSIM), Senior Member of the Society for Modelling and Simulation International (SCS), and Chartered Fellow of the British Computer Society

INVESTIGATION OF ACCIDENT BLACK SPOTS ON LATVIAN ROADS USING SCAN STATISTICS METHOD

Vitalijs Jurenoks, Vladimirs Jansons, Konstantins Didenko
Faculty of Engineering Economics
Riga Technical University
Kalku iela 1, Riga, LV1658, Latvia
vitalijs.jurenoks@rtu.lv
vladjans@latnet.lv
ief@rtu.lv.

KEYWORDS

Logistics system, accident black spots, scans statistics, modelling, Monte-Carlo method, system optimization.

ABSTRACT

Scan statistics is an instrument of research of statistical dynamics of the development of the object under investigation in space and time [1] – [5]. Research is made of statistics of road accidents distributed in space and time in the territory of Latvia. Alongside with the Monte-Carlo method when calculating the p-value, it is possible to use the direct method of scan modelling, taking into consideration the empirical rules of distribution of road accidents on the highways of Latvia. In this case both parametric and non-parametric methods may be used to describe the distribution of road accidents in Latvia. The methodology of research of road accidents in space and time enables to detect the most dangerous places on the highways of Latvia, check the efficiency of the decisions taken to improve the quality of roads and allocate the resources of the socio-economic system in a more rational way.

BACKGROUND

In most cases the possibility of traditional statistical conceptions and methods for investigation of real socio-economical objects is limited (the grey zone in Figure 1). Traditional statistical methods are more appropriate for investigation of the influence of internal factors – for the investigation of localized (grey zone) objects. Scan statistics allows investigating the socio-economical problems having extremely complex, i.e. synergic structure of interrelations (structure of the open systems). The analytical description of such systems in a simplified way enables to consider the likelihood scenarios of development of the object under investigation, but rarely of an object as a whole.
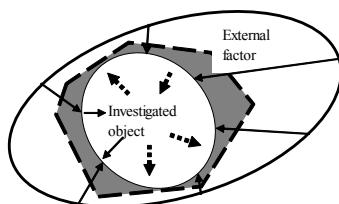
Figure 1: Illustration of area (grey zone) using traditional statistical methods
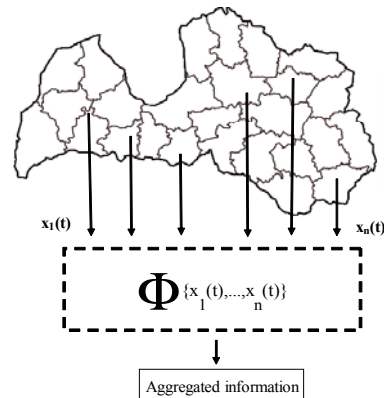
Figure 2: Data aggregation process

Traditional methods of research of industrial and economic objects is the analysis of the received statistical supervision of investigated objects $x(t_1)$, $x(t_2)$, ... ,$x(t_n)$ in time. As a result of processing the received statistical information, as a rule, is generalized information of the object as a whole, where $\Phi\{x_1(t), x_2(t),\ldots, x_n(t)\}$ is the generalized statistical information after processing in the aggregation block (Figure 2). Thus, use of traditional methods of research does not allow receiving necessary information about the behavior of separate parts of the object investigated. The illustration of accidents on Latvian roads, divided by administrative regions of Latvia is presented in Figure 3.
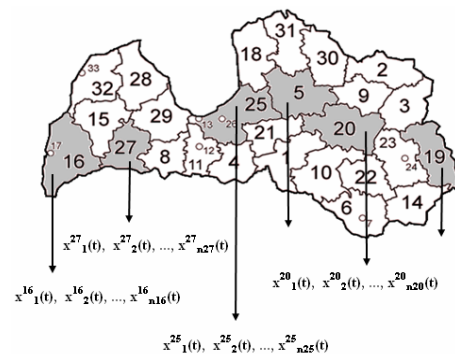
Figure 3: Illustration of accidents on roads of Latvia (divided by administrative regions)

The statistics of accidents on roads of Latvia is presented in Table 1.

Table 1: Traffic accident statistics on roads of Latvia

Time period, years	Traffic accidents	Persons injured	Persons killed
1993	3389	3721	670
1994	3814	4380	717
1995	4056	4903	611
1996	3709	4326	540
1997	3925	4674	525
1998	4540	5414	627
1999	4442	5244	604
2000	4482	5449	588
2001	4766	5852	517
2002	5083	6300	518
2003	5368	6634	483
2004	5081	6416	516
2005	4466	5600	442
2006	4302	5404	407

The diagram of traffic accidents on roads of Latvia by type per year is presented in Figure 4.
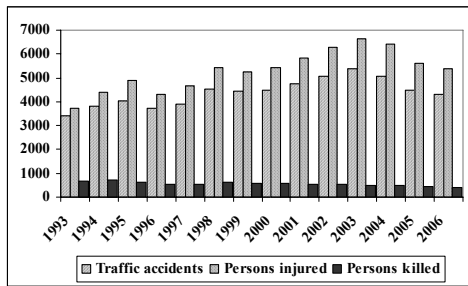
Figure 4: The diagram of traffic accidents on roads of Latvia [7, 8]

A new approach of investigation of a large scale socio-economic problem is shown in [9]. Let us consider what scan statistics is on the basis of examples. Scan statistics is used as an attempt to define clusters of events, using the saved up information distributed in space and time. We assume the scientific definition of variability (development) of an object (a changing object) has been non-uniformly distributed in time and space of the information field.

Thus, an important achievement in research is to reveal the structure of corresponding information fields. Scan statistics allows defining clusters of factors which describe heterogeneity. For better understanding of the idea of scan statistics, let us illustrate an example which is typical for statistical research of events which are distributed in time and space [4].

The distribution of road accident black spots in a small fragment of one of the regions of Latvia is presented in Figure 5.
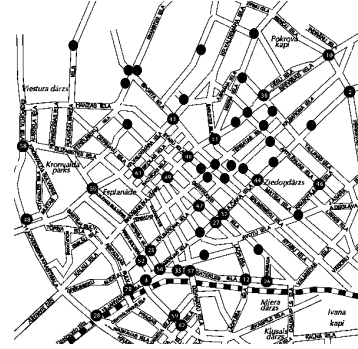
Figure 5: Illustration of road accident black spots in a small fragment of one of the regions of Latvia [7, 8]

APPLICATION OF SCAN STATISTICS METHOD

Let us have N events, distributed at time interval (0, T). Denote S_w as maximal number of events at a time interval with length w (the window of fixed length w of time). The maximum cluster S_w is called the scan statistics from the viewpoint that one scans the time period (0, T) with a window of size w and observes a large number of points (see Figure 6). W_k is the shortest period of time containing a fixed number of k events. The interval W_{r+1} is called the minimum r^{th} order gap, or r-scan statistics. The distributions of the statistics S_w and W_k are related. If the shortest window that contains k points is longer than w, then there is no window of length w that contains k or more points:

$$P(W_k > w) = P(S_w < k). \tag{1}$$

Let us illustrate the distribution of number of accidents on the roads by an example of some region at the time period from 2002 till 2007. In reviewing the data note that there is a 1 year period (from 1 March 2004 through 1 March 2005) when eight accidents were registered (see Figure 6).
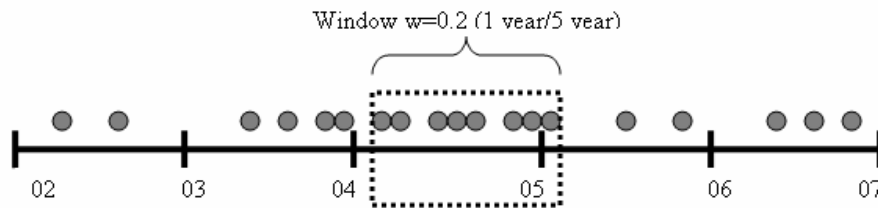
Figure 6: Scanning the unit time interval with the window of length w=0.2. Black points represent times of occurrence of N=19 events, $S_{0.2}=8$. The centers of the occurred "points" C_i have coordinates $t_1, t_2, …, t_N$

In Figure 6 we see concentration of 8 points at the time interval from 1 March 2004 to 1 March 2005. There is a question whether it is possible to explain such concentration proceeding from a null hypothesis. If it is impossible to explain the given concentration of points (accidents) by means of a null hypothesis it is necessary to recognize that the given concentration of points (accidents) has a special character. It means that process of occurrence of failures in the given region is influenced by additional factors. The given conclusion is an objective signal for decision-making in the sphere of traffic management in Latvia. We might explain this as follows: each of the 19 cases could either fall in the period (from 1 March 2004 to 1 March 2005) or not, independently of the other cases. The probability $b(k, N, w)$ found by computing the binomial probability for $N = 19$, $p = 1/5$:

$$b(k,N,w) = \binom{N}{k} w^k (1-w)^{N-k} =$$
$$b(k,19,0.2) = \binom{19}{k} 0.2^k (1-0.2)^{19-k} \quad (2)$$

gives the following probabilities for $k = 8$ and $k > 8$ (see Formula 2). Probabilities in Table 2 indicate that the cluster under investigation is somewhat unusual but this calculation does not answer the question posed.

Table 2: Probabilities for k = 8 and k > 8

	k = 8	k $\geq$ 8
p	**0.01662**	**0.02328**

Managers (researchers) want to know how unusual it is to get any 1 year period (not a specific year) containing eight cases. It is possible to solve this problem by dividing a period of five years into five disjoint 1 year periods and use the distribution of the maximum number of cases falling in any 1 year period. However, this does not answer the researchers question since the specific year observed overlaps two disjoint years (see Figure 7). Researchers are not limited by calendar years. The researchers are in effect scanning the 5 year period with a window of length of 1 year focusing on the maximum number of points in the scanning window (see Figure 7).

Figure 7: Illustration of the scanning window of fixed length w=0.2

It is easy to understand that there is an infinite number of sliding windows during the time interval (2002- 2007). To solve this problem in a constructive way we must assume some finite set of sliding windows (Figure 8).

Figure 8: Illustration of the scanning window of two fixed lengths w=0.1 and w=0.2. The centers of scanning windows are points with time coordinates t_1, t_2, …, t_N

Let us denote P(k, 19, 0.2) the distribution of the maximum number of cases in a year under the null hypothesis model for the unrestricted continuum of 1 year periods. The null hypothesis model for 19 cases C_i (i=1, 2,…, 19) were distributed independently by the binomial distribution function) and completely at random over the 5 year period.

The answer to the researchers question is provided by a scan statistics probability. Here the cluster is k=8, the total number of cases over the whole 5 year period is N=19 and the window size w is one year out of five, or w=0.2. The answer is provided by the probability P (8, 19, 0.2). Using formula (2.3) from the [11] probability P (k, N, w) can approximately be calculated as:

$$P(k,N,w) \approx (N-k+1)b(k-1,N,w)$$
$$- (N-k-1)b(k,N,w) + 2G_b(k+1,N,w), \quad (3)$$

where

$$b(k,N,w) = \binom{N}{k} w^k (1-w)^{N-k}$$
$$G_b(k,N,w) = \sum_{i=k}^{N} b(i,N,w)$$

Probabilities P (k, N, w), calculated with the help of formula (3), are presented in Table 3 (k = 7, 8,…,13). In our example: P(8, 19, 0.2) ≈ 12b(7, 19, 0.2) – 10b(8, 19, 0.2) + 2Gb(9, 19, 0.2) = 12·0.0443 - 10·0.0166 + 2·0.00666 = 0.379. It shows that the cluster observed is not unusual. How accurate is this

value (0.379) for this example? Very precise! The exact tabled result from [10] is 0.376. Exact formulas for the two special cases, P(2, N, w) and P(N, N, w) appear in various probability texts and had been known for many years. An exact formula for P(N, N, w) the cumulative distribution function of W_N the sample range of the N points is derived from Burnside [2].

Table 3: Values of functions b(k,N,w), G_b(k,N,w), P(k,N,w)

k	b(k, N, w)	G_b(k,N,w)	P(k, N, w)
7	0.04432	0.06760	0.8000264
8	0.01662	0.02328	0.3789698
9	0.00508	0.00666	0.1402787
10	**0.00127**	**0.00158**	**0.0412472**
11	**0.00026**	**0.00031**	**0.0097084**
12	**0.00004**	**0.00005**	**0.0018309**
13	**0.00001**	**0.00001**	**0.0002752**

For the function P(N,N,w) in case k=N, there is an exact formula:

$$P(N,N,w) = Nw^{N-1} - (N-1)w^N . \qquad (4)$$

In our case k = N = 19, w = 0.2 from (3), we derive the value for P(N, N, w) = P(19, 19, 0.2)-see Table 4.

Table 4: Values of functions b(k,N,w), G_b(k,N,w), P(k,N,w) in case k=18, 19

k	b(k, N, w)	G_b(k,N,w)	P(k, N, w)
18	0.00000	0.00000	0.0000000002870
19	**0.00000**	**0.00000**	**0.0000000000040**

An exact formula for P (2, N, w) the cumulative distribution function of W2, the smallest distance between any of the N points, is derived from Parzen [6] by a direct integration approach:

$$P(2,N,w) = 1 - [1-(N-1)w]N,$$
$$0 < w < 1/(N-1) \qquad (5)$$
$$P(2,N,w) = 1, \ for \ 1/(N-1) < w < 1 .$$

Analysing the statistics of accidents on roads of Latvia it is seldom possible to get an exact analytical solution for the distribution of road accidents. In this case there exists only one possibility and it is to use computer modelling. In the paper we have considered to use the Monte-Carlo method of scan statistics for calculation of p-value and testing null hypothesis H_0 (no clusters). The authors are using the approach by Wallenstein and Naus [10] in assuming the null hypothesis model which can be used for investigations of similar problems. Let S[x, x + w) = $S_{x,w}$ denote the number of events in [x, x + w). The scan statistics S_w is defined as [11]:

$$S_w = \sup_{0 \le x \le 1-w} S[x, x+w) \qquad (6)$$

This is often suggested as statistics (with an appropriate window length w) for testing the presence of clustering. Indeed it arises from the generalized likehood ratio test of uniformity (H_0) against the alternatives (H_1):

$$f(x) = 1/\{1+(\mu-1)\cdot w\}, \quad 0 \le x < T,$$
$$= \mu/\{1+(\mu-1)\cdot w\}, \quad T \le x < T+w, \qquad (7)$$
$$= 1/\{1+(\mu-1)\cdot w\}, \quad T+w \le x < 1 ,$$

where f(x) is the density function, $\mu > 1$ and T are unknown but w is known.

For our case the computer programs only look at a constant background rate of events and for the scenarios of grouped data. The authors calculate the scan statistics S_w for continuous data where it has been assumed that N = n events are occurring on a time-line (0, T). The authors generate uniform samples from this time-interval and construct an empirical distribution of the Pr(S_w > k) where k is the maximum number of events in a subinterval of width w (scanning window). There is an infinite number of sliding windows at the time interval (0, T). However, our approach was to consider only the N = n points in a sample derived from this time-interval, and then: (Step 1) sliding windows on the ordered sample data values (t_1, … , t_n) using the windows [t_i, t_i + w) for i = 1, … , n has been computed. The left square bracket "[" indicates inclusion of the lower point; the open right parenthesis ")" indicates exclusion of the upper point, which is a typical convention in mathematical analysis. In Step 2 the authors looked at the reverse chain of the sliding windows [t_i , t_i - w) for i = n, n-1, … , 1. By generating a large number of samples from the uniform distribution we derived an empirical distribution of Pr(S_w > k). We are interested in finding the value of "k" which shows a small p-value, typically 0.05 or smaller. The modelled p-value can be used for testing null hypothesis the samples of which are uniformly distributed against a clustering alternative. We assume that the given scan statistics (see Formula 6) is a fixed number N of events that have occurred at random in a fixed time period. This conditional (on N) case is called retrospective because in typical applications the researcher scans and makes inferences about N events that have occurred. In practice it is important to investigate the situation when a number of events in the time period are not viewed as a fixed number N that has already occurred, but rather as a random variable with a known probability distribution. We will consider a more popular discrete distribution for Poisson process – Poisson distribution function with an average of λ events per unit time. The typical application is prospective. The researchers seek to use scan statistics to monitor future road accident data or to design a Latvian road management system with the purpose to minimize the number of road accidents.

The Poisson process has been used for modelling real systems dealing with the occurrence of events in time or space. First useful applications of spatial scan

statistics are shown in [5]. Spatial scan statistics is a powerful method for spatial cluster detection. With spatial scan statistics it is possible to search over a given set of spatial regions, find those regions which are most likely to be clusters and correctly adjust for multiple hypothesis testing. Figure 9 illustrates a suspicion cluster – region in S with a high level of intensity q_{in} = 0.02 of accidents [5]. Scan statistics gives answer to the question – is this cluster real or is it a "visual illusion"?
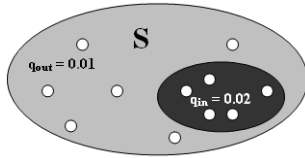
Figure 9: Frequency model of cluster - critical region

The simplest frequency model for this situation (Figure 9) can be written as:

Null hypothesis H_0 (no clusters) q_{in} = q_{all} everywhere (use maximum likelihood estimate of q_{all});

Alternative hypothesis H_1 (cluster in region S), q_{in} = q_{out} elsewhere (use maximum likelihood estimates of q_{in} and q_{out}, subject to $q_{in} > q_{out}$).

In one dimensional case Poisson scan statistics can be calculated as the computation of a scan statistics S_w for grouped data where we assumed that N = n events occur over T disjoint time intervals. Poisson samples $(x_1, \dots, x_T)$ with the intensity rate λ= N/T were generated. Then we summed the values x_j for all j in each interval [i-1, i-1 + w), for i = 1,…,T and found the maximum of these sums. The intervals here are finite since there are only T intervals if size w = 1, etc. For w = 2 we computed sliding windows of size 2 from (2, 4 , …, T) and then (1, 3, …, T) in steps of 2. Similarly scan statistics for other values of increasing w has been calculated. Generating a large number of samples from the Poisson distribution with rate "λ" an empirical distribution for $Pr(S_w \geq k)$ has been derived. It is important to see which value of "k" showed a small p-value, typically 0.05 or smaller. Modelled p-value can be used for testing null hypothesis. That means that accident events are distributed from the Poisson distribution with rate "λ" against a clustering alternative. A seed (initial number which random number generator uses to start

random number generation) is automatically inserted as the current time (in hours, minutes, and seconds).

The intensity of the accident development in one of the regions of Latvia is shown in Figures 10, 11. The relocation of the maximum intensity factor from one sector to another sector of the region was investigated.

Preliminary results of investigating accidents on the roads of Latvia show the following specific features of their distribution:

- lack of traffic saturation in rural areas does not allow to correctly identify clusters of road accidents in these regions;

- significant clusters occur in big cities of Latvia showing possibilities of improving road management system in Latvia.
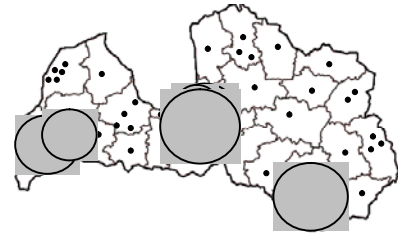
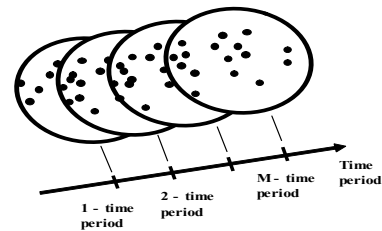Figure 10: Spatial-time scan statistics with a circle window

Figure 11: Scan statistics in time

Every year it is necessary to analyse the level of intensity of road accidents applying the method of scan statistics. The paper illustrates the scanning process with a circle window with fixed radius. It is necessary to scan accidents on roads of Latvia with windows of different circles with different radii. The illustration of the Monte-Carlo algorithm for cluster detection is presented in Figure 12.
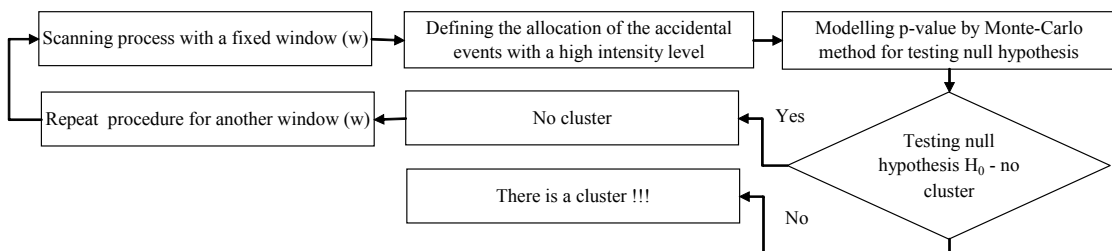

Figure 12: Illustration of the Monte-Carlo algorithm for cluster detection

Alongside with the Monte-Carlo method for calculation of p-value it is possible to apply the direct

method of statistics modelling taking into consideration the empirical rules of road accident

distribution on roads of Latvia. In this case both parametric and non-parametric methods of describing road accidents may be used.

CONCLUSION

The opportunity of using scan statistics methodology for research of road accidents on highways of Latvia is presented in this paper. The stochastic modelling of p-value by the method of Monte-Carlo for identifying the black spots of road accidents of Latvia is used.

The application of scan statistics enabled:

- to make analysis of road accidents in some big towns and regions of Latvia;
- to detect clusters with utmost intensity of road accidents applying scan windows of different sizes;
- to check significance of detected clusters with highest frequency of road accidents on the basis of null hypothesis equal to value of 0.05;
- to analyse the dynamics of changes of clusters detected, taking into consideration the time factor (in this case the research was limited by lack of information). Notwithstanding the limitation mentioned above, the possibility of analysing the existing clusters of road accidents in space and time, as well as those emerging in the future, is addressed by the authors of this paper. The findings of the research made about road accidents on Latvian highways highlight the following specific features of their distribution:

- insufficient transport vehicle saturation in rural areas does not allow to identify clusters of road accidents in these regions;
- significant clusters are detected in big cities of Latvia that provide an opportunity to improve the road system management in Latvia.

REFERENCES

1. Berman, M. and. Eagleson, G. K. Dec., 1985. "A Useful Upper Bound for the Tail Probabilities of the Scan Statistics." Journal of the American Statistical Association, Vol. 80, No. 392, doi: 10.2307/2288548, 886-889.

2. Burnside, W. 1928. "Theory of Probability". Cambridge, University Press, 106.

3. D'Antuono, M.F. 2006. ScanoR – "Computations of Scan statistics using R": A language and environment for statistical computing. Department of Agriculture and Food, Western Australia. http://www.agric.wa.gov.au (Email:mdantuono@agric.wa.gov.au).

4. Glaz, J.; Naus, J. and Wallenstein, S. 2001. "Scan Statistics". Springer Series in Statistics, 367 -370. http://books.google.lv/books?id=CHUwtWl6zOYC&pg=PR7&vq=scan+statistics&sig=r-YsGg5HK7TEVh6ZnsWypRQVPqI#PPR13,M1,

5. Kulldorff, M. 1997. "A spatial scan statistics. Communications in Statistics". - Theory and Methods, Volume 26, Issue 6, 1481 – 1496.

6. Parzen, E. Modern Probability "Theory and Its Applications". John Wiley & Sons Inc. December 1960, 464.

7. Road traffic accidents. http://www.csb.gov.lv/csp/content/?lng=en&cat=354

8. Road Traffic Accidents. http://www.sam.gov.lv/satmin/content/?cat=148

9. Tuia, D.; Kaiser, C.; Da Cunha; A., Kanevski, M. 2007. "Socio-economic cluster detection with spatial scan statistics". Case study: services at intra-urban scale. Geocomputation, National University of Ireland, Maynooth, 3-5 September 2007. http://www.clusterville.org/?CaseStudies/ScanStat.

10. Wallenstein S. and Naus J. 2003. "Statistics for temporal surveillance of bioterrorism". In: Syndrome Surveillance: Reports from a National Conference, Morbidity and Mortality Weekly Report 2004; 53, 74-78.

11. Wallenstein, S. 2005 "Scan Statistics". http://c3.biomath.mssm.edu/wscan.html

12. Ward, M.P. and Carpenter, T.E. 2003 "Methods for Determining Temporal Clusters in Surveillance and Survey Programs". In: Animal disease Surveillance and Survey Systems. Methods and Applications. Ed. Salman, M.D. Iowa State Press, Ames Iowa, 87-99.

AUTHORS BIOGRAPHIES

VITALIJS JURENOKS was born in Riga, Latvia. In 1976 he graduated from the Faculty of Engineering Economics of Riga Technical University, and for ten years has worked in an industrial enterprise in Riga. Since 1986 he has been lecturing at the Riga Technical University, and in 1987 was awarded the doctoral degree in economics (Dr.oec.). The main field of research pursued is planning, simulation and optimization of economic processes and systems. E-mail: vitalijs.jurenoks@rtu.lv.

VLADIMIRS JANSONS was born in Daugavpils, Latvia and is a graduate of the University of Latvia, where he studied mathematics and obtained his degree in 1970. For eight years he has worked in the Computing Centre of the University of Latvia. Since 1978 he has been lecturing at the Riga Technical University, where in 1983 he was awarded the doctoral degree in mathematics. The main field of research pursued is simulation and optimization of economic systems. E-mail: vladjans@latnet.lv.

KONSTANTINS DIDENKO was born in Jelgava, Latvia. In 1969 he graduated from the Faculty of Engineering Economics of Riga Technical University. Since 1969 he has been lecturing at the Riga Technical University where in 1985 he obtained the doctoral degree in economics. In 2006 he was elected a corresponding member of the Latvian Academy of Sciences. The main field of research pursued is planning and optimization of economic processes and systems. E-mail: ief@rtu.lv

LOCAL DELIVERIES TIME OPTIMIZATION FOR CITIES WITH UNSTABLE TRAFFIC

Pavels Patlins
Riga Technical University
Faculty of Engineering Economics
pavels.patlins@rtu.lv.

KEY WORDS

Delivery, Optimization, Route, Unloading.

ABSTRACT

The paper deals with delivery and vehicle unloading time importance for local distribution problems within practicable conditions. It is necessary to optimize both vehicle's driving time and unloading time to organize local circular route in optimal way.

The paper's author recommends registering vehicle unloading time for the each object within the particular route. It is also recommended to divide all roads into small sections. After that it is easy to define driving time for each section in particular hours.

The author also advises the investigation of distribution processes' service quality as well as distribution costs to improve circular route planning for local deliveries.

INTRODUCTION

Accuracy factor is really important for production delivery nowadays. Customers want to receive cargo on time and without delays. Carriers wish to use their vehicles in optimal way. Designing of the optimal circular route especially for cities with intensive traffic is a complicated problem. Traffic congestions in big cities are not constant; they may change by hour of day or day of week. So, often it is impossible to use common mathematic methods for route planning, because these methods will not provide the optimal result in practical conditions. If a specialist analyses all possible variants, he will spend a lot of time for planning process. Distribution process planning for cities is a sophisticated process due to loading of city's roads and unstable vehicle unloading time for particular objects.

CIRCULAR ROUTE PLANNING PROBLEMS

Routing is the cornerstone of modern logistics and transport environment. It is necessary to plan routes in optimal way in order to serve customers better than competitors. Actually, if the company has the optimal route, it also provides higher level of customer service for delivery of goods. On the other hand, optimal planning allows decreasing company's costs, connected with delivery of cargo. Sometimes one vehicle should serve 10 or more customers within the particular route (Figure 1). Normally circular route (connecting large number of customers) planning is a complex process for cities and other built-up areas, because there are many different ways of completing it. Often operators make a typical mistake during the routing process; they try to minimize only vehicle's way, serving customers within the particular route.

| A | Route opening point (depot) |
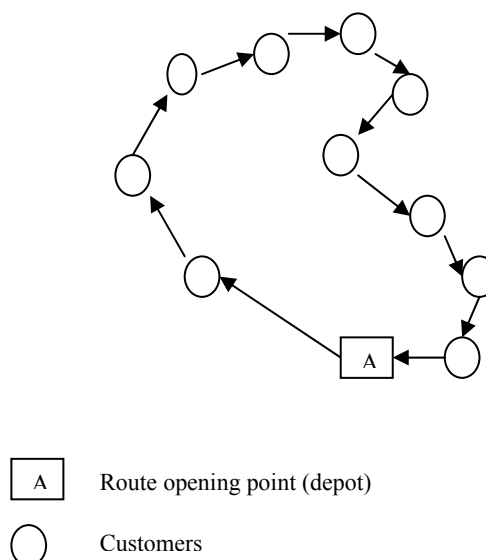
○ Customers

Figure 1: One Vehicle Serves Large Number of Customers (Circular Route).

The total time of delivery within one route consists of the 2 main elements: vehicle moving time and vehicle unloading time (figure 2, formula 1).
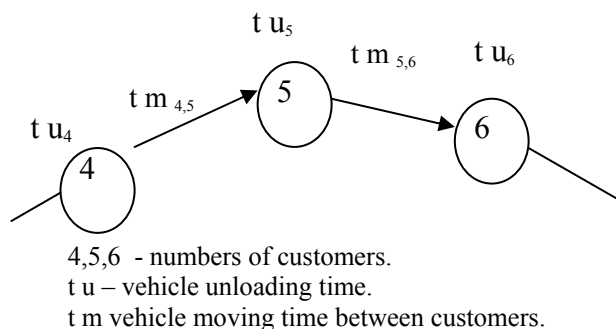

4,5,6 - numbers of customers.
t u – vehicle unloading time.
t m vehicle moving time between customers.

Figure 2: Circular Route's Fragment.

$$T_d = \sum_{i,j=1}^{n} tm_{ij} + \sum_{i=1}^{m} tu_i \qquad (1)$$

T d – total delivery time
tm$_{ij}$ - Vehicle moving time between the route's points i and j.
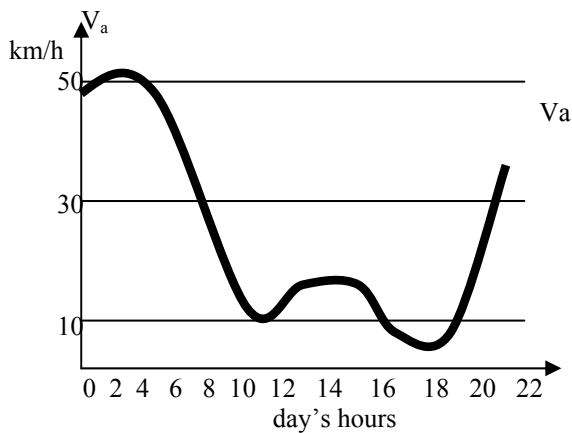t u$_i$ –Vehicle's unloading time for customer i.

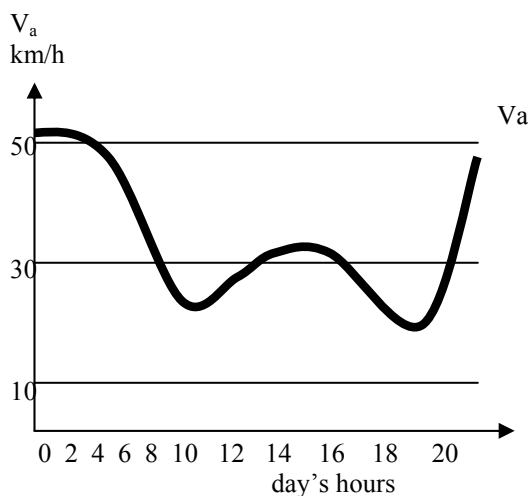Figure 3: The Average Vehicle's Speed (Va) Variables For Particular Crossroad in Riga by Hour (on Monday).

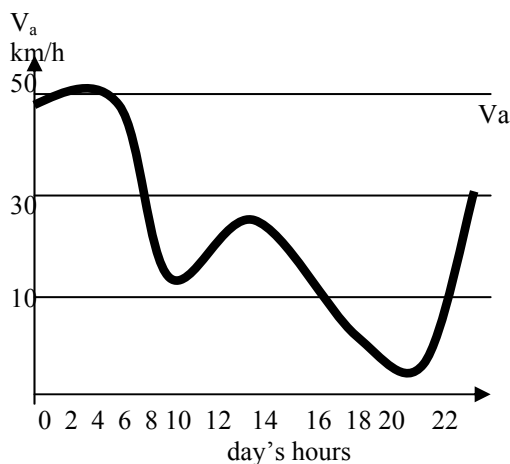Figure 4: The Average Vehicle's Speed (Va) Variables For Particular Crossroad in by Hour (on Tuesday).

Figure 5: The Average Vehicle's Speed (Va) Variables For Particular Crossroad in Riga by Hour (on Friday).

Basically, while planning local deliveries it is necessary to solve two problems. On the one hand, it is necessary to organize accurate deliveries. On the other hand,

operator should reduce the time, the vehicle spends, serving the particular route.

Today, designing an accurate delivery time is quite involved, because traffic congestions are unstable, they may change depending on days of the week or particular hours of the day [3]. The average vehicle's speed also changes – (figure 3-5)

The average vehicle's speed in the particular stage of Riga's street is unstable; it changes not only at the particular hours, but also on the particular days of week. The highest speed is at night (between 11 p.m. and 4 a.m.), but it not high at least at morning hours. The level of average speed is the lowest at 6.a m.-10 a m. and 17 p.m.-19 p.m. in several streets of Riga on Monday. This index changes on other days of the week (figure 4-5).

Scientific methods and computer programs may be used to solve particular time-planning problems, connected with routing, though every method or program has different cons, for delivery planning within cities. For example, mathematical methods need fixed information about time, speed and other factors, but in our case these factors vary, making it impossible to achieve the optimal result using only mathematical methods. In this case it is possible to achieve the optimal result, only by using special methods, programs and specialists' experience combined.

VEHICLES MOVING TIME PLANNING PROBLEM'S SOLUTION FOR CITIES WITH HEAVY TRAFFIC

Generally, it is possible to use only micro-elements method to improve route planning in cities with heavy traffic. To explain it we may investigate the following example.

The time of delivery consists of two basic positions: time of driving and time of loading/unloading. Specialist's task is to consider these elements for the particular route. It is possible to plan driving time precisely, if we use the following approach for solving routing problem.

1.step. To create "roads' passport". It is necessary to divide the particular territory's roads into different elements. Normally, may use different division basics– between two crossroads, between two customers, between a crossroad and a customer, between two traffic lights etc. (figure 6)

2.step. Notice, what is the average speed of driving at the particular hours for the each element. (figure 6)

3.step. Creating a database of average speed for the each element into particular hours of the day and day of the week. (1 table)

According to the information of Table 1, for instance, the average speed for road's elements A2, B3, B2,C2 and C1 at 7-8 a m. is 5 km/h, but at 10-11 a.m. it is 5-10 km/h.

The average speed for road's elements A1 and B1 at 7-8 a.m. is 5-10 km/h, but at 10-11 a.m. it is 15-20 km/h.
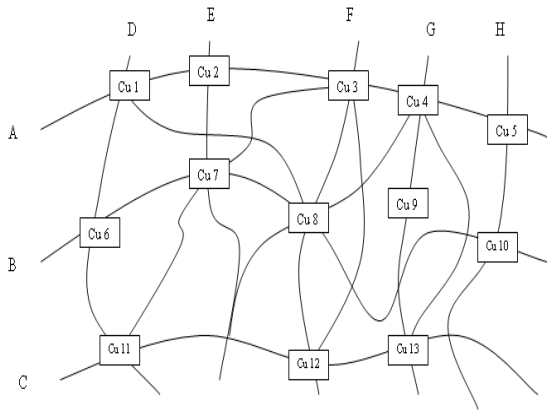
A,B,C- main roads' names;
D,E,F,G,H – other roads names.
Cu 1 – Cu 13 –customers names

Figure 6. Roads, Divided Into Particular Elements.

Table 1: Example of Vehicle Average Speed in the Particular Roads Elements Depending on Hours of a Day.

a)

Hours of a day / The average speed	7-8 a m	8-9 a m	9-10 a m	10-11 a m	11-12 a m
5km/h	Cu6-Cu1, Cu1–Cu2, Cu2-Cu7	Cu8-Cu7, Cu1–Cu2, Cu2-Cu7	Cu8-Cu7, Cu1–Cu2, Cu2-Cu7	Cu8-Cu7,	Cu1–Cu2,
5-10	Cu3-Cu8,	Cu1–Cu2,		Cu8-Cu7, Cu3-Cu8	Cu1–Cu2, C2-C7
10-15	Cu2-Cu3 Cu6-Cu7	Cu2-Cu3 ,	Cu6-Cu7	Cu2-Cu3 Cu6-Cu7	Cu2-Cu3 Cu6-Cu7
15-20				Cu1–Cu2, Cu2-Cu7	
20-25					
25-30					
30+	Cu5-Cu10, Cu9–Cu13, Cu4-Cu5 Cu4-Cu9	Cu4-Cu5 Cu4-Cu9	Cu4-Cu5 Cu4-Cu9	Cu4-Cu5 Cu4-Cu9	Cu5-Cu10 Cu9–Cu13, Cu4-Cu5 Cu4-Cu9

b)

Hours of a day / The average speed	12-14	14-15	15-16	16-17
5km/h				
5-10	Cu8-Cu7, C11-C7	Cu1–Cu2, C11-C7	Cu2-Cu7	
10-15	Cu2-Cu3	Cu6-Cu7	Cu2-Cu3 Cu6-Cu7	Cu6-Cu7
15-20	Cu2-Cu7	Cu8-Cu7,	Cu1–Cu2,	Cu2-Cu7
20-25	Cu5-Cu10			
25-30				
30+	Cu9–Cu13, Cu4-Cu5 Cu4-Cu9	Cu4-Cu5 Cu4-Cu9	Cu4-Cu5 Cu4-Cu9	Cu5-Cu10, Cu9–Cu13, Cu4-Cu5 Cu4-Cu9

Cu 1 – Cu 13 – Customers names (from figure 6)

Creating such a database is an involved process; specialists should spend a lot of time to complete it. Using this database helps operator to plan vehicle's moving time in optimal way. Table 1a) is the first part of database, involved information about the period of time between 7 a.m. and 12 a.m. Table 1b) includes information about average speed in the particular roads elements for the period of time between 12 and 17. If the distance of the particular road's element is known as well as the vehicle's speed in the particular hour, we may plan the time we spend to drive through this element. Certainly, it makes route planning process easier and helps to create the optimal delivery precisely. In this way it is possible to decrease vehicle moving time uncertainty, but it is not enough for delivery optimization in general. It is also necessary to plan vehicle's unloading time for each customer.

VEHICLES UNLOADING PROCESSES TIME CONTROL OPTIMIZATION

According to formula 1, in order to achieve the best delivery time planning, it is necessary to improve planning of vehicle unloading time for each customer. If there are 50 customers within the particular route and the operator makes a five-minute mistake while planning of unloading time for each customer, the total mistake exceeds 4 hours (approx. half of the total working hours). There are 2 possible alternatives in this case:

-vehicle's driver serves all customer 4 hours earlier and returns to the depot. (Vehicle has 4 additional hours of idle time in this case);

-vehicle's driver does not have enough time to serve customers during his working hours.

Both situations are unsuitable for the operator; and it is necessary to improve vehicle's unloading time control to satisfy company's customers.

It is essential to create vehicle's discharging time system for all company's customers.

First of all, one may divide unloading process to separated elements:

1. vehicle wait time (in a queue) while other vehicles will be discharged by the particular customer. (T w)
2. vehicle's maneuvering time by the particular customer.(T m)
3. time needed for driver to park a vehicle. (T p)
4. time needed for driver to get off from the cabin. (T off)
5. time needed for driver to go around vehicle. (T go)
6. time needed for driver to open vehicle's body. (T op)
7. time needed for driver to take a box with goods from the vehicle's body. (T t)
8. time needed for driver to carry the 1 box till a place, where customer receive it. (T c)
9. time needed for customer to check the cargo. (T ch)
10. time to sign documents . (T s)
11. time needed for driver to return into the vehicle.(T r)
12. vehicle's maneuvering time to drive out from customer's territory. (T ma)

Using formula 1, we may specify vehicle's unloading time.

$$Td = \sum_{i,j=1}^{n} tm_{ij} + \sum_{i=1}^{m} tu_i \qquad (1)$$

$$tu_i = T w + T m + T p + T off + T go + T op + T t + T c + + Tch + T s + T r + T ma. \qquad (2)$$

It is possible to calculate precisely all these elements, excluding vehicle's wait time (in a queue) while other vehicles will be unloaded by the particular customer, because usually it is too difficult to forecast this index. To optimize delivery time planning it is essential to arrange company's customers.
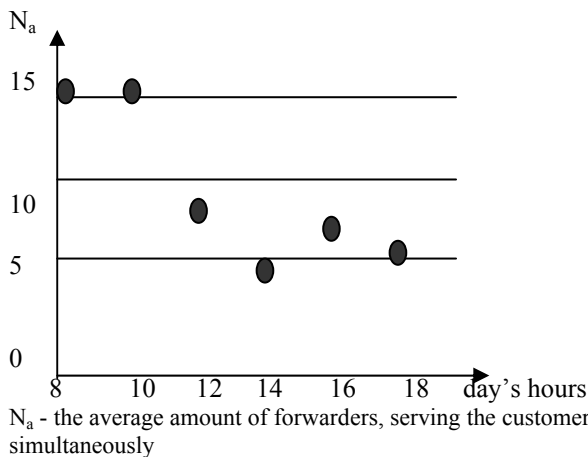

N_a - the average amount of forwarders, serving the customer simultaneously

Figure 7: The Average Amount of Forwarders Served the Customer Simultaneously at the Particular Day's Hours.

Usually the average number of forwarders, serving the particular customer simultaneously, varies depending on particular hours of a day. Perishable products are usually delivered in the morning. Some warehouses' working time is in progress day and night, but others work only till 5 or 6 p.m. Generally, caterers' concentration is different for each customer depending on particular time of a day.

First of all, it is necessary to investigate the average level of suppliers' concentration near the particular client.

Table 2: Wait (in a Queue) Time Before the Unloading Process Near Company's Customers on the Particular Hours of a Day.

Hours of day / wait (in a queue) time	8-9	9-10	10-11	11-12	12-14	14-15	15-16	16-17
5 minutes	Cu1, Cu2, Cu3.	Cu1, Cu2, Cu3	Cu8, Cu7,	Cu4,Cu5,	Cu1, Cu2, Cu3	Cu1, Cu2, Cu3	Cu1, Cu2, Cu3	Cu7,Cu11, Cu12
5-10 min.	Cu5, Cu8,		Cu1, Cu2, Cu3	Cu6, Cu8	Cu7, Cu11, Cu12	Cu6, Cu8	Cu4, Cu5	
10-15 min.	Cu4	Cu6-Cu7	Cu6, Cu5, Cu12	Cu2, Cu9	Cu9	Cu7	Cu7	Cu4, Cu5
15-20min.			Cu11		Cu8	Cu10 Cu4	Cu11	Cu2, Cu9
More than 20 min					Cu4		Cu13	

Cu 1 – Cu 13 – Customers names (from figure 6)

After this information is collected, it is essential to process it, creating a special system. It is possible to investigate wait (in a queue) time by the each customer (figure 6) at the particular day's hours.

After creating this separated database (2 Table), operator may minimize whole unloading process planning time as well as make the total delivery planning process more precise. Operators or other persons should control vehicle unloading time near the each company's customer on the particular hours of a day. It is not a relational database and it is possible to update it manually changing information about the particular customer.

It is necessary to improve delivery service quality as well as to reduce delivery costs in general.

CONCLUSIONS

It is essential to use micro-elements method, while planning circular routes for cities with intensive traffic. Creation of roads elements average speed's database makes route planning process easier.

Vehicle's unloading time also influences the total time of delivery. Usually it is difficult to register this element of the delivery time because vehicle's wait(in a queue) time may vary for each customer depending on the particular hours of a day. It is essential to divide the total unloading time into separated micro-elements, as well as to create wait time database in order to optimize unloading time control process for local deliveries.

REFERENCES

1. Patlins, P. 2005. Universal Routing Algorithm for Cities and Other Built-Up Areas. In: Zadnik Stirn, Drobne, Nemec (Eds.). 8th International Symposium on Operational Research in Slovenia, Nova Gorica, Slovenia, SOR '05 proceedings. Ljubljana: Slovenian Society Informatika, Section for Operational Research, 2005, pp.247.-252.
2. Rushton A. Croucher P. and Baker P. 2006. The Handbook of Logistics and Distribution Management 188-194.
3. Sprancmanis N, Patļins P. 2006. Preču fiziskās sadales organizēšana intensīvas satiksmes apstākļos. RTU Zinātniskie raksti, 3.sērija, 12. sējums. 144-149.
4. Sprancmanis. N. 2003. Biznesa loģistika. Rīga, Vaidelote, 359.

AUTHOR BIOGRAPHY.

Pavels Patlins was born in Riga, Latvia and studied Economics and Logistics and obtained his Master degree in 2003 in Riga Technical University. He has been working for the Riga Technical University since 2003, conducting Logistics and Transport Managements lectures. His e-mail is: pavels.patlins@rtu.lv.

SYNTHESIS OF DEVELOPMENT SCENARIOS OF COMPLEX SYSTEMS

Vladimir V. Kul'ba and Dmitry A. Kononov
Trapeznikov Institute of Control Sciences
of Russian Academy of Sciences
65, Profsoyuznaya str. Moscow, Russia, 117806,
E-mail: kulba@ipu.rssi.ru

Galina Merkuryeva
Riga Technical University
1 Kalku Street
Riga, Latvia, LV-1658
E-mail: gm@itl.rtu.lv

KEYWORDS

Scenario, control, safe development.

ABSTRACT

This paper presents new theoretical results of the methodology elaboration for the scenario-based research of complex systems. Basic techniques and tasks of the scenario-based research of social economic systems are formulated. Basic concepts of behaviour scenario, as well as of its elements and their characteristics are defined. Tasks of optimal synthesis of synergetic and attractive scenarios of social economic system development are determined.

INTRODUCTION

According to the suggested formalised methodology of the scenario-based research of social economic systems (SES), the process of modelling, planning and control implementation is performed on the basis of integration of system logical, structure-social and scenario-based approaches (Kononov et al. 2005).

The system logical approach assumes constructing and studying objects of research from the position of systems analysis whose underlying concept is *a formal system object*. This allows one to study objects and processes from the viewpoint of formal logical and mathematical methods.

The structure-social approach assumes constructing and studying social economic objects of control on the basis of definition and structuralization of human activity types. This enables studying objects and processes in social systems at different strata of society's social structure and distinguishing main social objects and social structures as well as describing different social processes on that basis.

A combination of system logical and structure-social approaches makes it possible to study human activity as the main motive reason of social economic system development, considering it as the structured spectrum of formalised processes of state changes of the social economic objects and subjects of action.

The scenario-based approach supposes investigation of the processes occurring in social economic systems on the basis of construction and investigation of behaviour scenarios (synergetic scenarios) of social subjects of action and scenarios of control (attractive scenarios) of social objects.

A combination of system logical, structure-social and scenario-based approaches allows scenario-based research on the basis of construction of the scenario system representing both general and specific characteristics of SES, its elements, the discovered relationships among them and indicated features of those relationships. An effectively constructed scenario system helps solve multi-aspect problems, perform scenario-based analysis and synthesise scenarios of rational behaviour of different social subjects of action, which makes it possible to pass on to the construction of a safety ensuring system of the given object, structure or process of SES.

In (Kononov 2007), impulse synergetic and attractive scenarios of direct control of SES development are discussed. It is found that in principle there exist only six possible scenarios of such development. This paper continues describing results of mathematical investigation of SES behaviour model constructed using the language of operator directed graphs (Roberts 1976).

SYNTHESIS OF SES DEVELOPMENT SCENARIOS

The typology of applied systems described in (Kul'ba et al. 2004) enables SES structuralization and studying its subsystems using methods of the scenario-based analysis. Examples of SES development modelling give evidence that even at the classification level a number of their characteristic structural peculiarities can be distinguished. They can be grouped by features which determine the unique properties and relationships among the macroindicators of the revealed classification group. Using formal operations in the considered scenario spaces, it is then possible to make an effective analysis and accomplish the synthesis of optimal development scenarios and thus construct a transformation scenario.

The scenario $\Re$ as an object of research is a complex hierarchically constructed structure (Kononov 2001). Let us distinguish its basic elements:

- *General elements of a model:* a metaset $\mathbf{M} = (M_O; M_E; M_D, M_{MO}; M_{ME}; \hat{A})$, sets of internal states $Y \subseteq E^m$ and environmental states of SES $X \subseteq E^n$, expanded phase space $X \times Y = Z \subseteq E^{n+m}$ and

its expert significant partition (ESP) $\Xi = \{\{Z^{(\alpha)}\}, A^{(e)}\}$;

• *Spatial information elements:* trajectory scale ZT, event scale $\Re T$, an event significant to expert (ESE) $\mathfrak{I}_{ev}^{\alpha}(\mathbf{z}, t)$, depth $N \in \Re T$, horizon $t_N = T \in ZT$, conditional decision $\varsigma(t_i) = (\vartheta(t_i), \gamma(t_i))$ at time instant $t_i \in ZT$, including also controllable operated factors (the CO factors) $\vartheta \in M_0 \subseteq E^d$, uncontrollable factors $\gamma \in \Gamma \subseteq E^\gamma$ and a set of conditional decisions $S^{(cs)} = M_0 \times \Gamma$.

The scenario $\Re$ with the initial event ESE $\mathfrak{I}_{ev}^{(\alpha(0))}(t_0) = \{Z^{(\alpha(0))}, \mathbf{z}(t_0), t_0\}$, is constructed with regard to trajectory scales ZT and event scales $\Re T$, using these rules:

$$\Re = \Re\{(I(t_i), t_i)\}, \qquad (1)$$

where $t_i \in \widehat{A}$ for $i = 0, 1, ..., N; t_0 = 0$,

$$I(t_i) = (S(t_i), M^{(QIH)}(t_i)), \qquad (2)$$

where $t_i \in \widehat{A}$ for $i = 0, 1, ..., N; t_0 = 0$.

Conditions $I(t_i)$ at $t_i \in ZT \subseteq R$ are characterised by situation $S(t_i)$ and a quasi-information hypothesis (QIH), $M^{(QIH)}(t_i)$.

A situation is a set of events which occurred before time instant t:
$$S(t) = \{\mathfrak{I}^{(i)}(\mathbf{z}^{(i)}(t_i), t_i) \; 0 \le t_i \le t, \; i = 0, 1, ..., s; \; t_0 = 0\} \; (3)$$
whereas the expected event is the expected realisation of $\mathbf{z}(t) \in Z$ factors of the expanded phase trajectory in the elementary partition (EP) $Z^\alpha \in \Xi$ at time instant t:

$$\mathfrak{I}_{ev}^{\alpha}(t) = \{Z^\alpha, \mathbf{z}(t), t\}. \qquad (4)$$

The occurrence of the next in turn ESE $\mathfrak{I}_{ev}^{(\alpha_{i+1})}(t_{i+1})$ of scenario $\Re$ at time instant t_{i+1} with regard to scale ZT, or at time instant $i+1$ with regard to the discrete scale $\Re T$, is determined by *scenario forming elements*: situation $S(t_i)$, QIH $M^{(QIH)}(t_i)$ and scenario formation strategy $\breve{C}(t_i)$ at time instant t_i as a ternary relationship:

$$\mathfrak{I}_{ev}^{(\alpha_{i+1})}(t_{i+1}) = (S(t_i), M^{(QIH)}(t_i), \breve{C}(t_i)). \qquad (5)$$

Each of the above mentioned elements characterises the process of its construction in a certain aspect:

$S(t_i)$ —situation described by a set of most important SES parameters; determines phase conditions under which the researcher has to make a decision;

$M^{(QIH)}(t_i)$ —quasi-information hypothesis; it represents the researcher's knowledge structure and

actually represents a model of the uncertainty taken into account (Kul'ba et al. 2004);

$\breve{C}(t_i)$ —scenario forming strategy; it determines those scenario continuation techniques which the operating party is able to afford.

In general, scenarios of three basic types can be determined and constructed:

• *Synergetic scenarios*, which model behavioural aspects of the system under consideration and describe the development spectrum provided that there are no control influences on SES functioning processes;

• *Direct control scenarios,* which model the development spectrum in connection with the chosen set of arrangements realising the direct control task;

• *Attractive scenarios*, which characterise the behaviour of a SES in accordance with the desirable - reasonable from the viewpoint of the decision maker's aim - control influences, i.e. realising the inverse control task. Control is determined as a result of realisation of the current situations which, in particular, occur under the influence of external events and of decision maker's decisions. Within that scheme, a scenario can be synthesised as an instrument of formal analysis of alternative situation development variants at the specified aims in the presence of uncertainty.

The occurrence of the next in turn ESE $\mathfrak{I}_{ev}^{(\alpha_{i+1})}(t_{i+1})$ of the synergetic scenario is determined by general elements of the model and expert scenario forming elements according to rule (5). The occurrence of the next in turn ESE of the attractive scenario additionally requires that the rules of operation with spatial information components aimed at making control conditional decision must be clarified according to the following scheme:

— rule $\mathbf{A}^{(A)}(t_i) \subseteq \widehat{A}$ for evaluating the effectiveness of the attractive strategy $\breve{C}^{(A)}(t_i)$, which enables finding conditional decision $\varsigma(t_i) = (\vartheta(t_i), \gamma(t_i)) \in S^{(cs)}$;

— rule $\Lambda^{-1}(\vartheta(t_i), t_i)$ for determining elements of the CO factor;

— rule $\mathbf{A}^{(fixm)}(t_i) \subseteq \widehat{A}$ for fixing the time of the next in turn ESE $\mathfrak{I}_{ev}^{(\alpha_{i+1})}(t_{i+1})$.

Formally, a strategy is defined as a ternary relationship:

$$\breve{C}^{(A)}(t_i) = (\mathbf{A}^{(A)}(t_i), \Lambda^{-1}(\vartheta(t_i), t_i), \mathbf{A}^{(fixm)}(t_i)). \qquad (6)$$

As soon as at time instant $t \in ZT$ a QIH is fixed in the form of the structured subset $M^{(QIH)}(t) \subseteq S^{(cs)} = Z^0$ of the expanded phase space Z, the next in turn ESE can be determined by indicating the CO factor $\vartheta = \Lambda(\mathbf{u}, \mathbf{p}, \Delta)$ in accordance with the chosen strategy $\breve{C}$ for scenario $\Re$ formation. $\breve{C}$ is constructed using rules $A^{(s)} \subseteq \widehat{A}$. Each point $\mathbf{z} = (\mathbf{x}, \mathbf{y}) \in Z$ along with the

fixed rules of trajectory $\mathbf{A}^{(fixm)}(t_i) \subseteq \bar{A}$ discretisation at each time instant $t_i \in ZT$ $i = 0,1,...$ (or on a certain horizon of scenario $T \in ZT$) and elements of metaset $\mathbf{M}$ determine object $\Re(\mathbf{z}, I(t_i), t_i)$ behaviour scenario, outgoing from point $\mathbf{z}$, as a sequence of QIH, scenario formation strategies, rules for choosing them and expected events corresponding to them. The representation of scenario $\Re$ as a sequence of ESE and QIH, that is conditions 0, is oriented towards the decision maker or external -with regard to the process of problem analysis- user. In contrast to 0, a detailed technology for obtaining a sequence of that kind in the form of a set of strategies and rules of choice 0 is intended for the expert dealing with the statement and solving of the problem under study. The problem of the formalised synthesis of SES functioning and development scenarios is a topical task. The underlying idea of the suggested process of complex system development modelling is the elaboration of scenarios of their behaviour for the specified target analytical purposes, which will result in the synthesis of the scenario with the specified properties, including an optimal scenario.

The optimality of a scenario can be considered from two viewpoints.

The first one consists of the verbal statement of the problem of scenario generation as a choice of the most rational (optimal) expert description of problem situations fixing the most essential properties of the considered process of complex system behaviour. The optimality property is underlying for the scenario to be formed, as it distinguishes that description technique from related concepts: a forecast, a plan and a trajectory. This is the choice of the next in turn expert meaningful event - but not of all possible ways of development or goal achievement in terms of the expanded phase space- which determines the scenario to be formed. The indication of the appropriate sequence of ESE at the given assumptions about the course of object development depends on the goals of research (the synergetic approach) or on control (the attractive approach). Thus, we should speak about *the optimality of scenario formation*. The principles and a sequence of optimal scenario formation according to the concept are described in (Kononov 2001).

An alternative concept of scenario optimality from the viewpoint of the scenario-based research is formulated as choosing an optimal scenario out of the specified scenario set in the scenario space under consideration. Such scenarios can be obtained irrespective of the researcher; his task is to determine "the best" of them.

The multiplicity of system behaviour scenarios becomes apparent in qualitatively different sides of the system under study, which are fixed to determine a strategy in the ascertained quasi-information hypothesis of the operating party in synthesising the scenario. Within the formalized approach, the manysideness of study object

is represented in the diversity of its characteristics and properties.

By a characteristic we understand the qualitative side of study object formalised in the form of a formal model. A numeric function f corresponding to the chosen scale is called *a numeric characteristic- or measurement indicator- of an object*; a Boolean characteristic of an object is a model *of object's property*; other models of the characteristics are also possible.

A set of characteristics $\mathbf{f}^{(e)}$ of element $e(\Re)$ of scenario $\Re$ as well as other components of the scenario system can serve as an initial set of synthesis characteristics.

Metric characteristics of elements of scenario determine its properties as an element of the expanded phase space $Z \subseteq E^{n+m}$, of the scenario space $Z^{(sc)}$, as well as of metric space $\breve{M}(Z, \rho_Z)$ and normalised space $Nor(Z, \| \ \|_Z)$ constructed on their basis.

A number of characteristics of general elements of the model of the scenario system are described in (Kul'ba et al. 2004).

Within the ESP-model (Kul'ba et al. 2004), an ESE $\Im_{ev}^\alpha(t)$ can be considered to be an element of space

$$\Im_{ev} ZT = Z \times Z \times ZT \subseteq E^{2(n+m)+1}$$

$$(Z^\alpha \subseteq Z, \mathbf{z}(t) \in Z, t \in ZT), \qquad (7)$$

which will be called the space of expert significant events (SESE).

To calculate characteristics of ESP and ESE, it is necessary to define a metric space $\breve{M}(M\Im_{ev}, \rho_{ev})$, the distance in which is determined by metric ρ_{ev} (for instance, introduce metric $\rho_Z = \rho(\mathbf{z}^{(1)}, \mathbf{z}^{(2)})$ in the expanded phase space Z)

$$M\Im_{ev} \subseteq \Im_{ev} ZT. \qquad (8)$$

Examples of synergetic and attractive characteristics of scenarios in the metric space 0 are given in (Kul'ba et al. 2004). While forming a scenario, different strategies can be used, whose application depends on the circumstances of searching for the next in turn ESE: the situation, the model of accounted uncertainty (QIH), rules of taking it into account etc. It is proposed to consider the search for an effective strategy in two stages: accounting for the uncertainty and optimisation. A number of characteristics of uncertainty $H^{(I)}$ accounting techniques and a strategy of functional optimisation $H^{(O)}(\vartheta)$, including optimism measure, variativity measure etc., are defined in (Kul'ba et al. 2004, Kononov 2001). As a more complicated example of QIH characteristic, the structure related to the reflexive model of uncertainty

constructed in (Novikov and Chartishvili 2003), can be considered.

A set of characteristics

$$\mathbf{f}(\Re)=\{ f_\tau(\Re) \; \tau \in \Sigma \} \qquad (9)$$

of scenario $\Re$ is as a rule a convolution

$$f_\tau(\Re)=\Theta_\tau \big(\mathbf{f}^{(e)}(\Re)\big) \qquad (10)$$

of partial characteristics $\mathbf{f}^{(e)}$ of elements $_e(\Re)$ constituting scenario $\Re$. Most frequently, the permissibility and/or desirability of the formed scenario are indicated as a set of its properties. In different applications, these properties are distinguished:

• achievement of the given observability extent of the object and its surrounding;
• achievement of the specified completeness of object behaviour description;
• achievability scenario;
• safety scenario;
• scenario realisticity etc. (Kul'ba et al. 2004).

Due to that, determination of transformation 0 is an independent task.

Assume direction $\mathbf{a} \in Z$ and a positive number δ are set. Let us give some spatial characteristics of scenarios:
-*current remoteness of scenario* $\Re(t_0,T)$ *from vector* $\mathbf{a}$

$$d_E^{(t)}(\Re(t_0,T),\mathbf{a})=\rho_E(\mathbf{z}(t),\mathbf{a})=\|\mathbf{z}(t)-\mathbf{a}\|_E;$$

--*current angular remoteness of scenario* $\Re(t_0,T)$ *from vector* $\mathbf{a}$

$$d_a^{(t)}(\Re(t_0,T),\mathbf{a})=\rho_a(\mathbf{z}(t),\mathbf{a})=\left\| \frac{\mathbf{z}(t)}{\|\mathbf{z}(t)\|_E} - \frac{\mathbf{a}}{\|\mathbf{a}\|_E} \right\|_E;$$

-*minimal remoteness of scenario* $\Re(t_0,T)$ *from vector* $\mathbf{a}$ *on the horizon* T

$$d_E^{\min}(\Re(t_0,T),\mathbf{a})=\min_{\mathbf{z}\in\Re(t_0,T)}\|\mathbf{z}-\mathbf{a}\|_E=\min_{t\in t_0,T} d_E^{(t)}(\Re(t_0,T),\mathbf{a});$$

-*minimal angular remoteness of scenario* $\Re(t_0,T)$ *from vector* $\mathbf{a}$ *on the horizon* T

$$d_a^{\min}(\Re(t_0,T),\mathbf{a})=\min_{\mathbf{z}\in\Re(t_0,T)}\rho_a(\mathbf{z},\mathbf{a})=\min_{t\in t_0,T} d_a^{(t)}(\Re(t_0,T),\mathbf{a});$$

-*maximal remoteness of scenario* $\Re(t_0,T)$ *from vector* $\mathbf{a}$ *on the horizon* T

$$d_E^{\max}(\Re(t_0,T),\mathbf{a})=\max_{\mathbf{z}\in\Re(t_0,T)}\|\mathbf{z}-\mathbf{a}\|_E=\max_{t\in t_0,T} d_E^{(t)}(\Re(t_0,T),\mathbf{a});$$

-*maximal angular remoteness of scenario* $\Re(t_0,T)$ *from vector* $\mathbf{a}$ *on the horizon* T

$$d_a^{\max}(\Re(t_0,T),\mathbf{a})=\max_{\mathbf{z}\in\Re(t_0,T)}\rho_a(\mathbf{z},\mathbf{a})=\max_{t\in t_0,T} d_a^{(t)}(\Re(t_0,T),\mathbf{a})$$

-*a set of periods when scenario* $\Re(t_0,T)$ *exceeds the bounds of* δ *-vicinity of vector* $\mathbf{a}$

$$T_E^{(t,out)}(\Re(t_0,T),\mathbf{a},\delta)=\underset{t_0\leq t\leq T}{Arg}(d_E^{(t)}(\Re(t_0,T),\mathbf{a})\geq\delta);$$

-*the first moment when scenario* $\Re(t_0,T)$ *exceeds the bounds of* δ *-vicinity of vector* $\mathbf{a}$

$$T_E^{(\min,out)}(\Re(t_0,T),\mathbf{a},\delta)=\min_{t_0\leq t\leq T}(T_E^{(t,out)}(\Re(t_0,T),\mathbf{a},\delta));$$

-*the last moment when scenario* $\Re(t_0,T)$ *exceeds the bounds of* δ *-vicinity of vector* $\mathbf{a}$

$$T_E^{(\max,out)}(\Re(t_0,T),\mathbf{a},\delta)=\max_{t_0\leq t\leq T}(T_E^{(t,out)}(\Re(t_0,T),\mathbf{a},\delta));$$

-*a set of periods of scenario* $\Re(t_0,T)$ *staying in* δ *-vicinity of vector* $\mathbf{a}$

$$T_E^{(t,in)}(\Re(t_0,T),\mathbf{a},\delta)=\underset{t_0\leq t\leq T}{Arg}(d_E^{(t)}(\Re(t_0,T),\mathbf{a})\leq\delta);$$

- *the first moment of scenario* $\Re(t_0,T)$ *falling in* δ *-vicinity of vector* $\mathbf{a}$

$$T_E^{(\min,in)}(\Re(t_0,T),\mathbf{a},\delta)=\min_{t_0\leq t\leq T}(T_E^{(t,in)}(\Re(t_0,T),\mathbf{a},\delta));$$

- *the last moment of scenario* $\Re(t_0,T)$ *staying in* δ *-vicinity of vector* $\mathbf{a}$

$$T_E^{(\max,in)}(\Re(t_0,T),\mathbf{a},\delta)=\max_{t_0\leq t\leq T}(T_E^{(t,in)}(\Re(t_0,T),\mathbf{a},\delta));$$

- *current velocity of scenario* $\Re(t_0,T)$ *in the direction of vector* $\mathbf{a}$

$$v_E^{(t)}(\Re(t_0,T),\mathbf{a})=d_E^{(t)}(\Re(t_0,T),\mathbf{a})/t;$$

- *current angular velocity of scenario* $\Re(t_0,T)$ *with regard to vector* $\mathbf{a}$

$$v_a^{(t)}(\Re(t_0,T),\mathbf{a})=d_a^{(t)}(\Re(t_0,T),\mathbf{a})/t;$$

- *minimal velocity of scenario* $\Re(t_0,T)$ *in the direction of vector* $\mathbf{a}$

$$v_E^{(\min)}(\Re(t_0,T),\mathbf{a})=\min_{t_0\leq t\leq T} v_E^{(t)}(\Re(t_0,T),\mathbf{a});$$

-*minimal angular velocity of scenario* $\Re(t_0,T)$ *with regard to vector* $\mathbf{a}$

$$v_a^{(\min)}(\Re(t_0,T),\mathbf{a})=\min_{t_0\leq t\leq T} v_a^{(t)}(\Re(t_0,T),\mathbf{a});$$

-*maximal velocity of scenario* $\Re(t_0,T)$ *in the direction of vector* $\mathbf{a}$

$$v_E^{(\min)}(\Re(t_0,T),\mathbf{a})=\max_{t_0\leq t\leq T} v_E^{(t)}(\Re(t_0,T),\mathbf{a});$$

-*maximal angular velocity of scenario* $\Re(t_0,T)$ *with regard to vector* $\mathbf{a}$

$$v_a^{(\min)}(\Re(t_0,T),\mathbf{a})=\max_{t_0\leq t\leq T} v_a^{(t)}(\Re(t_0,T),\mathbf{a}).$$

Let us set vector $\mathbf{a} \in Z$. Assume that in the expanded phase space on the horizon T there are set scenarios $\Re_1$ of SES development with the initial ESE $\Im_{ev}^{(\alpha_1(0))}(t_0)=\{Z^{(\alpha_1(0))},\mathbf{z}^{(1)}(t_0),t_0\}$ and scenarios $\Re_2$ with the initial ESE $\Im_{ev}^{(\alpha_2(0))}(t_0)=\{Z^{(\alpha_2(0))},\mathbf{z}^{(2)}(t_0),t_0\}$, constructed according to rules 0-0.

Definition 1. *Scenario* $\Re_1$ *will be called **a scenario of outstripping development** in the direction of vector* **a** *on the horizon* T *as compared to scenario* $\Re_2$, *if*

$$\rho_a\left(\mathbf{z}^{(1)}(t),\mathbf{a}\right)\leq\rho_a\left(\mathbf{z}^{(2)}(t),\mathbf{a}\right) \text{ for all } t\in[0,T]. \quad (11)$$

Vector **a** is a vector of the desirable direction of SES development. Condition (11) means that on the horizon T scenario $\Re_1$ is closer to vector **a** as compared to scenario $\Re_2$, with regard to the angular distance. Definition 1 enables scenario comparison of by their closeness to the specified direction of SES development in the expanded phase space Z.

Let us assme that
$$\Delta(\rho_a,\mathbf{a},\Re_1,\Re_2,t)=\rho_a\left(\mathbf{z}^{(1)}(t),\mathbf{a}\right)-\rho_a\left(\mathbf{z}^{(2)}(t),\mathbf{a}\right), (12)$$
i.e. the angular mismatch between the scenarios with regard to the direction **a** at time t.

Definition 2. *Scenario* $\Re_1$ *will be called **a scenario of overtaking development** in the direction of vector on the horizon* T *with growth rate* δ, *as compared to scenario* $\Re_2$, *if for* $\delta>0$ *the following holds:*

$$\Delta(\rho_a,\mathbf{a},\Re_1,\Re_2,t)\geq\delta \text{ for all } t\in[0,T]. \quad (13)$$

Definition 2 makes it possible to compare scenarios by the rate of their approach to the set development of SES in the expanded phase state Z.

Let us assign a set of characteristics as functionals from $M\ \Im_{ev}$ in E^S

$$\Omega\,\mathbf{f}(\Im_{ev})=\{f_\tau(\Im_{ev}),\ \tau=\overline{1,S}\}\subseteq E^S (14)$$

of events significant to the expert $\Im_{ev}\in M\ \Im_{ev}$, as well as a set of subsets

$$\widehat{M}^{\Re}\,\mathbf{f}(\Im_{ev})=\{\widehat{M}^{\Re}f_\tau(\Im_{ev}),\ \tau=\overline{1,S}\}\subseteq E^S,$$

$$\breve{M}^{\Re}\,\mathbf{f}(\Im_{ev})=\{\breve{M}^{\Re}f_\tau(\Im_{ev}),\ \tau=\overline{1,S}\}\subseteq E^S \quad (15)$$

interpreted as permissible or desirable values of characteristics **f**.

Definition 3. *An expert significant event* $\Im_{ev}$ *is called permissible with regard to characteristic* f_τ *in scenario* $\Re$, *or* f_τ *-permissible if*

$$f_\tau\in\widehat{M}^{\Re}f_\tau(\Im_{ev}). \quad (16)$$

An expert significant event $\Im_{ev}$ *is called desirable with regard to characteristic* f_τ *in scenario* $\Re$, *or* f_τ *- desirable, if*

$$f_\tau\in\breve{M}^{\Re}f_\tau(\Im_{ev}). \quad (17)$$

Denote a set of f_τ-permissible ESE as $\widehat{M}^{\Re}_{ev}f_\tau\subseteq\Im_{ev}ZT$. We will call an ESE $\Im_{ev}$, permissible by all characteristics f_τ $\tau=\overline{1,S}$, **f** *-permissible for scenario* $\Re$.

Denote a set of **f** -permissible for scenario $\Re$ ESE as $\widehat{M}^{\Re}_{ev}\,\mathbf{f}$.

Denote a set of f_τ-desirable ESE as $\breve{M}^{\Re}_{ev}f_\tau\subseteq\Im_{ev}ZT$. An ESE $\Im_{ev}$, permissible by all characteristics f_τ, will be called **f** *-desirable for scenario* $\Re$.

Let us denote a set of **f** - desirable for scenario $\Re$ ESE as $\breve{M}^{\Re}_{ev}\,\mathbf{f}$.

Example 1. Permissible and prioritative ESE.

Let there be assigned a set of conditionally permissible states $\widehat{Q}$, i.e.

$$\mathbf{z}(t)\in int\,\widehat{Q}. \quad (18)$$

Expression 0 models necessary conditions of SES development trajectory membership in the set $\widehat{Q}$.

Let there be set an ESE 0. Let us introduce a Boolean characteristic

$$q(\Im_{ev},\widehat{Q})=\begin{cases}0,\ if\ \mathbf{z}(t)\in int\,\widehat{Q}\\1,\ if\ \mathbf{z}(t)\notin int\,\widehat{Q}\end{cases}. \quad (19)$$

Then conditions 0 can be written in the language 0 as follows:

$$q(\Im_{ev},\widehat{Q})=0, \quad (20)$$

i.e. ESE $\Im^\alpha_{ev}(t)=\{Z^\alpha,\mathbf{z}(t),t\}$ is q-permissible for scenario $\Re$ if 0 holds.

Let a set of prioritative states $\breve{R}$ be given, that is condition

$$\mathbf{z}(t)\in\breve{R} \quad (21)$$

is the most desirable to the decision maker.
Assume an ESE 0 is assigned. Let us introduce a Boolean characteristic

$$r\left(\mathfrak{I}_{ev},\breve{R}\right)=\begin{cases}0, & if \ \mathbf{z}(t)\in\breve{R}\\1, & if \ \mathbf{z}(t)\notin\breve{R}\end{cases}. \qquad (22)$$

Then conditions 0 could be written using the language 0 as

$$r\left(\mathfrak{I}_{ev},\breve{R}\right)=0, \qquad (23)$$

i.e. ESE $\mathfrak{I}_{ev}^{\alpha}(t)=\{Z^{\alpha},\mathbf{z}(t),t\}$ is r-permissible for scenario $\mathfrak{R}$ if 0 holds.

Definition 3 enables determination of optimal with regard to criterion 0 (prioritative) and permissible by criterion 0 ESE as ESE out of the set

$$\mathfrak{I}_{ev}\in M_{ev}^{\mathfrak{R}}(q,r)=M_{ev}^{\mathfrak{R}}q\cap M_{ev}^{\mathfrak{R}}r. \qquad (24)$$

Such ESE $\mathfrak{I}_{ev}$ are permissible and optimal for scenario $\mathfrak{R}$.

Definition 3 and Example 1 help generalise definitions of ESE optimality and permissibility in the case when an ESE 0 does not belong to set (24), i.e. the ESE is either nonpermissible or non-prioritative for scenario $\mathfrak{R}$.

Example 2. Generally permissible and optimal ESE.
Let us set numbers ε and δ and introduce characteristics of ESE 0 in space Z

$$q\left(\mathfrak{I}_{ev},\widehat{Q}\right)=\rho_Z\left(\mathbf{z},\widehat{Q}\right)=\inf_{\mathbf{v}\in Q}\rho_Z(\mathbf{z},\mathbf{v}), \qquad (25)$$

i.e. the distance to the set of conditionally permissible states and

$$r\left(\mathfrak{I}_{ev},\breve{R}\right)=\rho_Z\left(\mathbf{z},\breve{R}\right)=\inf_{\mathbf{v}\in R}\rho_Z(\mathbf{z},\mathbf{v}), \qquad (26)$$

i.e. the distance to the set of conditionally prioritative states and

Definition 4. We will consider that ESE $\mathfrak{I}_{ev}$ is δ-permissible if

$$q\left(\mathfrak{I}_{ev},\widehat{Q}\right)\le\delta, \qquad (27)$$

i.e. the distance from $\mathfrak{I}_{ev}$ to set $\widehat{Q}$ does not exceed δ.

Definition 5. We will call ESE $\mathfrak{I}_{ev}$ ε-optimal if

$$r\left(\mathfrak{I}_{ev},\breve{R}\right)\le\varepsilon, \qquad (28)$$

which means that the distance from $\mathfrak{I}_{ev}$ to the set of prioritative states does not exceed ε.

Thus a 0-permissible ESE is contained in the set of conditionally permissible states but a 0-optimal ESE is contained in the set of prioritative states in the case if these sets are closed.

Definitions of QIH, strategies and characteristics of a scenario allow one to formally state *the task of observation and analysis* of the process of factor transformation as well as its representation as a totality of $M\ \mathfrak{R}^{(syn)}$ *synergetic scenarios* each of which indicates a fundamental way of SES development.

Assume there is assigned a collection 0, as well as a set $\widehat{M}^{\mathfrak{R}}\mathbf{f}$ of their permissible values.

Task 1. To accomplish a synthesis of the synergetic scenario $\mathfrak{R}^{(syn)}$, that is on the assigned horizon of the scenario, $t=[t_0,t_1]$ to show a sequence of conditions

$$\mathfrak{R}^{(syn)}=\mathfrak{R}^{(syn)}\{\left(I^{(syn)}(t),t\right)\}, \qquad (29)$$

where $t=\overline{0,T}$, $\mathfrak{R}^{(syn)}\in M\ \mathfrak{R}^{(syn)}$,

but $I^{(syn)}(t)$ — conditions at time instant t for which a collection of characteristics 0 is $\mathbf{f}$-permissible.

Determination of controls, resources and rules for choosing effective strategies of ESE formation makes it possible to state *tasks of control* of the process of parameter transformation and its representation in the form of *an attractive scenario* $\mathfrak{R}^{(atr)}$.

Task 2. To construct an attractive scenario $\mathfrak{R}^{(opt)}$, that is to carry out the synthesis of optimal control in the form of a sequence of $\mathbf{f}$-permissible ESE

$$\mathfrak{R}^{(atr)}=\mathfrak{R}^{(atr)}\{\left(I^{(atr)}(t),t\right)\}, \text{ where } t=\overline{0,T}. \quad (30)$$

where $t_0=\min_{s\in g}t_0^{(s)}$, $t_1=\max_{s\in g}t_1^{(s)}$.

A scenario of that kind is formed according to this rule:

$$\mathfrak{R}^{(atr)}=\mathfrak{R}^{(atr)}\{\left(I^{(atr)}(t),t\right)\}, \text{ где } t=\overline{0,T}. \quad (31)$$

where $I^{(atr)}(t)$ — conditions at time instant t. In contrast to the synergetic scenario, the procedure of control formation is its obligatory element.

CONCLUSIONS

The proposed statement of the task of synthesis of synergetic and attractive scenarios allows one to consider tasks of that kind in a general aspect for different subject areas, different employed models describing the behaviour of the object and its environment as well as for different kinds of uncertainty accounted within the scenario-based research of SES.

REFERENCES

Kononov D.A., Kul'ba V.V., Malugin V.D., Shubin A.N. 2005. *Information processes and information control.* Proceedings „*Human Factors in Management*", SINTEG, Moscow. [*In Russian*].

Kononov D.A., 2007. *Synthesis of Social Economic System Development Scenarios with Operator graph Apparatus.* Avtom. Telemekh., No. 1. [in Russian]

Roberts, F.S., 1976. *Discrete Mathematical Models with Application to Social, Biological, and Enviromental Problems.* New York: Prentice Hall, 1976.

Kul'ba V.V., Kononov D.A., Kosjachenko S.A., Shubin A.N., 2004. *Design Techniques of Development Scenarios of Social Economic Systems.* SINTEG, Moscow. [*In Russian*].

Kononov D.A., 2001. *Effective Scenarios Synthesis Strategies in Automatic Control Systemems in Emergency Situation.* Avtom. Telemekh., No. 2, pp, 170-181. [in Russian]

Novikov D.A., Chartishvili, 2003. *Reflexive games.* SINTEG, Moscow. [*In Russian*].

AUTHOR BIOGRAPHIES

VLADIMIR V. KUL'BA, Academician, Dr.habil.sc., Professor, Trapeznikov Institute of Control Sciences Russian Academy of Sciences. He has made significant contribution in areas of control theory, optimal modular data processing, local and distributed data bases, network computing systems, and others.

DMITRY A. KONONOV, Dr.Math., Professor, Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, Moscow, Russia. His research area is development of mathematical models and methods for analysis and synthesis of complex systems' scenario.

GALINA MERKURYEVA, DSc., Dr.sc.ing., Professor at the Department of Modelling and Simulation, Riga Technical University, Latvia. Research interests are in the fields of discrete-event computer simulation and metamodelling methodology, knowledge-based simulation systems, simulation-based training and fuzzy systems theory.

Visual Validation Of PLC Programs

Min.S Ko
Department of Industrial
Engineering
Ajou University
Suwon, Korea

Sang C. Park
Department of Industrial
Engineering
Ajou University
Suwon, Korea

Gi Nam. Wang
Department of Industrial
Engineering
Ajou University
Suwon, Korea

KEYWORDS

PLC verification, plant model, virtual device model, virtual factory simulation

ABSTRACT

Proposed in this paper is the architecture of a PLC programming environment that enables a visual verification of PLC programs. The proposed architecture integrates a PLC program with a corresponding plant model, so that users can intuitively verify the PLC program in a 3D graphic environment. The plant model includes all manufacturing devices of a production system as well as corresponding device programs to perform their tasks in the production system, and a PLC program contains the control logic for the plant model. For the implementation of the proposed PLC programming environment, it is essential to develop an efficient methodology to construct a virtual device model as well as a virtual plant model. The proposed PLC programming environment provides an efficient construction method for a plant model based on the DEVS (Discrete Event Systems Specifications) formalism, which supports the specification of discrete event models in a hierarchical, modular manner.

1.INTRODUCTION

Generally, industrial production lines are dynamic systems whose states change according to the occurrence of various events, thus exhibiting the characteristics of a discrete event system. If manufacturers are to remain competitive in a continuously changing marketplace, they must not only continue to improve their products, but also strive to improve production systems continuously (B.K Choi et al. 2000). Thus, an efficient prototyping environment for production systems is crucial. A modern production line is a highly integrated system composed of automated workstations such as robots with tool-changing capabilities, a hardware handling system and storage system, and a computer control system that controls the operations of the entire system. The implementation of a production line requires much investment, and decisions at the design stage have to be made very carefully to ensure that a highly automated manufacturing system will successfully achieve the intended benefits.

Simulation is an essential tool in the design and analysis of complex systems that cannot be easily described by analytical or mathematical models (Klingstam. 1999; Al-Ahmari 1999.). It is useful for calculating utilization statistics, finding bottlenecks, pointing out scheduling errors and even for creating manufacturing schedules. Traditionally, various simulation languages, including ARENA® and AutoMod®, are used for the simulation of manufacturing systems. Those simulation languages have been widely accepted both in industry and in academia; however, they remain as analysis tools for the rough design stage of a production line, because their simulation models are not realistic enough to be utilized for a detailed design or for implementation purposes. For example, real production lines are usually controlled by PLC (Programmable Logic Controller) programs (Rukkan A 1997), as shown in Figure 1, but conventional simulation languages roughly describe the control logic with independent entity flows (job flows) between processes.
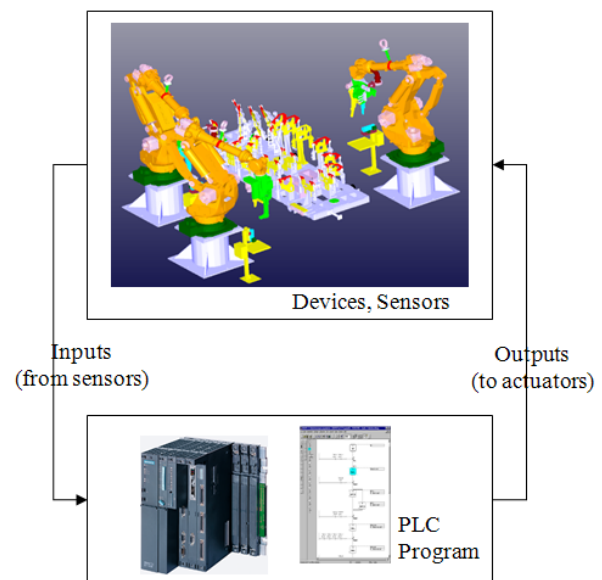

Figure 1: Production system controlled by a PLC program.

For a detailed design (virtual prototyping) of a production line, it is necessary to create a much more detailed simulation model that can forecast not only the production capability of the system but also the physical validity and efficiency of co-working machines and control programs. As shown in Figure 1, various

machines that operate simultaneously in an industrial manufacturing system are usually controlled by PLCs, currently the most suitable and widely employed industrial control technology (Chuang. 1999; Manesis. 2005; Rullan. 1997; Jang. 1997). A PLC (Programmable Logic Controller) emulates the behavior of an electric ladder diagram. As they are sequential machines, to emulate the workings of parallel circuits that respond instantaneously, PLCs use an input/output image table and a scanning cycle. When a program is being run in a PLC it is continuously executing a scanning cycle. The program scan solves the Boolean logic related to the information in the input table with that in output and internal relay tables. In addition, the information in the output and internal relay tables is updated during the program scan. In a PLC, this Boolean logic is typically represented using a graphical language known as a ladder diagram (Rullan. 1997).

Various software tools have been developed for the modeling and verification of PLC-based systems via the use of timed automata, such as UPPAAL2k, KRONOS, Supremica and HyTech, mainly for programs written in a statement list language also termed Boolean (Manesis. 2005). Those software tools verify PLC programs to a certain extent; however, they remain limited. In particular, it is not easy for users to determine whether the PLC programs actually achieve the intended control objectives.

The objective of this paper is to propose the architecture of a PLC programming environment that enables the visual validation of a PLC program. The proposed PLC programming environment employs a virtual plant model consisting of virtual devices, so that users can easily verify the PLC program. The overall structure of the paper is as follows. Section 2 illustrates the architecture of the proposed PLC programming environment, while Section 3 describes an efficient construction methodology for a plant model, which can be synchronized with a PLC program. Section 4 shows an example and illustrations. Finally, concluding remarks are given in Section 5.

2. VISUAL VALIDATION OF PLC PROGRAMS

To design the architecture of the PLC programming environment, it is important to understand the basic procedure used to construct a PLC program (ladder diagram). (Chuang et al. 1999) proposed a procedure for the development of an industrial automated production system that consists of nine steps. They are: 1) Define the process to be controlled; 2) Make a sketch of the process operation; 3) Create a written sequence listing of the process step by step; 4) On the sketch, add the sensors needed to carry out the control sequence; 5) Add the manual controls needed for the process-setup or for operational checks; 6) Consider the safety of the operating personnel and make additions and adjustments as needed; 7) Add the master stop switches required for a safe shutdown; 8) Create a ladder logic diagram that will be used as a basis for the PLC program; and 9) Consider the possible points

where the process-sequence may go astray. The most time-consuming task for the control logic designers is the 8-th step, which is usually done by the repetitive method of 'Code writing, testing and debugging' until the control objectives are achieved (Manesis. 2005). The bottleneck of the 8-th step is that the conventional PLC programming environments are not especially intuitive, particularly for the testing and debugging of a PLC program, as they show only the status of a PLC without providing any links to the target system (production line). For the validation of a PLC program, engineers need to imagine the state changes of a production line from the input and output ports of a PLC. That is the reason conventional PLC programming environments are often inefficient and prone to human error. As the configurations of production lines and their control programs become more complicated, there is a strong need for a more intuitive PLC programming environment. It is hoped that this paper will take positive steps in this direction.
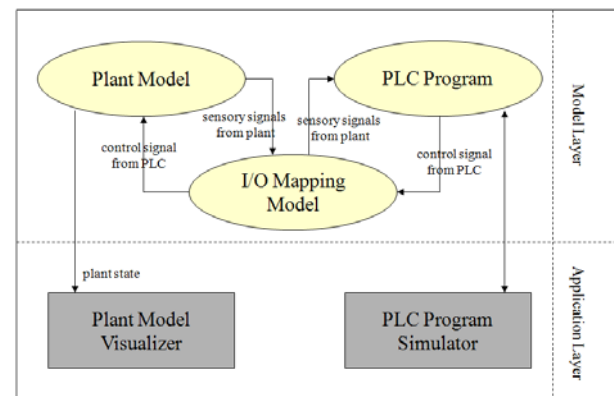

Figure 2: The proposed PLC programming environment

Figure 2 shows the architecture of the proposed PLC programming environment. It consists of two layers, a model layer and an application layer. The model layer has three models, a plant model (virtual factory model), a PLC program (control model) and an I/O mapping model. The plant model includes all manufacturing devices of the production system as well as the corresponding device programs to perform their tasks in the production system, and the PLC program contains the control logic for the plant model. For the integration of the plant model and the PLC program, it is necessary to define the mapping between the plant model and the PLC program, which is described by the I/O mapping model. The application layer simultaneously provides two interfaces to users. The 'PLC simulator' performs the simulation of the control program, and the 'plant model visualizer' shows the corresponding plant model (3D graphic models) reflecting the changing states of the production system during the PLC simulation. Thus, it becomes much easier for users to verify the PLC program though the plant model visualizer.

Among the three models of the model layer, the plant model plays a key role in the proposed PLC programming environment. As mentioned earlier, the

plant model should contain all devices as well as the device control programs. Thus, it can be considered as a 'virtual factory model', a model executing manufacturing processes in computers as well as the real world (M. Onosato. 1993; K Iwata. 1995; L. Ye. 1997). To implement a virtual factory, it is necessary to construct digital models for all the physical and logical elements (entities and activities) of a real factory.

The plant model consists of manufacturing devices with their positions in the layout. To represent such a manufacturing device, this paper employs the concept of a virtual device: a digital model imitating the physical and logical aspects of a real device. A virtual device needs to maintain its relationships with other devices or PLC programs as well as the inherent attributes of the device, such as the kinematics and geometric shape. To do so, a virtual device is split into two parts; a shell and a core. The shell part can adapt to the different configurations of production systems, and the core part undertakes the inherent properties of the device. The concept of a virtual device is shown in Figure 3.
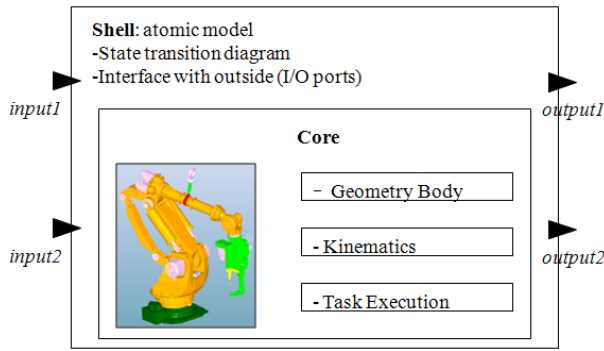
Figure 3: Virtual device model

The reusability of a virtual device is very important, as a virtual device can be used for many different configurations of production systems. Without the effective reusability characteristic of a virtual device, it is more common to create new components from scratch than to search for useful elements in other systems. For the reusability of a virtual device, it is essential for the shell part of a virtual device to be flexible enough to be compatible with any configuration of a production system. To build such a shell part, this paper employs Zeigler's DEVS (Discrete Event Systems Specifications) formalism (B.P.Zeigler. 1984; T.G.Kim. 1994), which supports the specification of discrete event models in a hierarchical, modular manner. The semantics of the formalism are highly compatible with object-oriented specifications for simulation models. Within the DEVS formalism, one must specify two types of sub-models: 1) the atomic model, the basic models from which larger models are built, and 2) the coupled model, how atomic models are connected in a hierarchical manner. Formally, an atomic model M is specified by a 7-tuple:

$$M = < X, S, Y, \delta_{int}, \delta_{ext}, \lambda, t_a > \quad (1)$$

X: input events set;
S: sequential states set ;
Y: output events set;
δ_{int} : $S \rightarrow S$: internal transition function;
δ_{ext} : $Q * X \rightarrow S$: external transition function
$Q = \{(s,e)| s \in S, 0 \leq e \leq t_a(s)\}$: total state of M;
λ : $S \rightarrow Y$: output function;
t_a : $S \rightarrow Real$: time advance function.

The four elements in the 7-tuple, namely $\delta_{int}, \delta_{ext}, \lambda$ and t_a, are called the characteristic functions of an atomic model. The second form of the model, termed a coupled model, shows a method for coupling several component models together to form a new model. Formally, a coupled model DN is defined as:

$$DN = < X, Y, M, EIC, EOC, IC, SELECT > (2)$$

X: input events set;
Y: output events set;
M: set of all component models in DEVS;
$EIC \subseteq DN.IN * M.IN$: external input coupling relation;
$EOC \subseteq M.OUT * DN.OUT$: external output coupling relation;
$IC \subseteq M.OUT * M.IN$: internal coupling relation;
$SELECT$: $2^M - \emptyset \rightarrow M$: tie-breaking selector,

where the extensions .IN and .OUT represent the input port set and the output port set of the respective DEVS models. For the implementation of the plant model, the shell part of a virtual device is represented as an atomic model, and the entire plant model is represented as a coupled model, including the atomic models (virtual devices) and the coupling relationships between them. The detail specifications for the plant model are addressed in the following section.

3. PLANT MODEL CONSTRUCTION

The objective of the proposed PLC programming environment is to provide an intuitive PLC programming and verification environment by connecting the plant model to the PLC program. To achieve this objective, it is essential to develop an efficient construction procedure of a plant model. Figure 4 shows the interactions among three models of the PLC programming environment. The three models are a plant model, an I/O mapping model, and a PLC program. The plant model is controlled by the PLC program through the I/O mapping model.

Given that a plant model consists of virtual devices, the construction method of a virtual device is described

before explaining the construction method of a plant model. As explained earlier, a virtual device consists of a shell part and a core part.

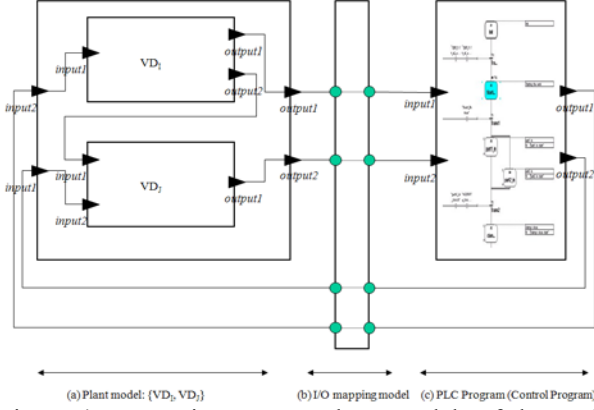(a) Plant model: {VD₁, VD₁} (b) I/O mapping model (c) PLC Program (Control Program)

Figure 4: Interactions among three models of the PLC programming environment

The shell part, enclosing the core part (the inherent properties of a device, such as kinematics, geometric shape and the execution of device-level commands), should allow a virtual device model to adapt to different plant configurations. This part is modeled as an atomic model of the DEVS formalism, which is a timed-FSA (finite state automata). To define the shell part of a virtual device, first it is necessary to identify the set of tasks that are assigned to the device. The activation of each task is normally triggered by an external signal from either the PLC program or other virtual devices. Once the set of tasks is identified for a virtual device, it is then possible to extract the state transition diagram, which defines an atomic model of the DEVS formalism. Figure 4-(a) shows a simple example of an AGV (Automatic Guided Vehicle) with two tasks, *T1* (movement from p1 to p2) and *T2* (movement from p2 to p1). As the two tasks should be triggered by external events, the shell part of the AGV must have two input ports, termed here as *Signal_1* and *Signal_2*, as shown in Figure 4-(b). From the set of tasks, it is possible to instantiate the state transition diagram automatically. For this example, there are four states, *P1, DoT1, P2* and *DoT2*. While *P1* and *P2* take external events from the input ports (*Signal_1, Signal_2*) for state transitions, *DoT1* and *DoT2* take internal events that are the end events of the two tasks (*T1* and *T2*). The DEVS atomic model of the virtual device, corresponding to the AGV, can be described as follows:

Shell of a virtual device:

$$M = < X, S, Y, \delta_{int}, \delta_{ext}, \lambda, t_a > \qquad (3)$$

$X = \{Signal\_1, Signal\_2\}$
$S = \{P1, DoT1, P2, DoT2\}$
$Y = \{T1Done, T2Done\}$

$\delta_{int}(DoT1) = P2$

$\delta_{int}(DoT2) = P1$

$\delta_{ext}(P1, Signal\_1) = DoT1$

$\delta_{ext}(P2, Signal\_2) = DoT2$

$\lambda(DoT1) = T1Done$

$\lambda(DoT2) = T2Done$

$t_a(DoT1) = Time\_1$

$t_a(DoT2) = Time\_2$.
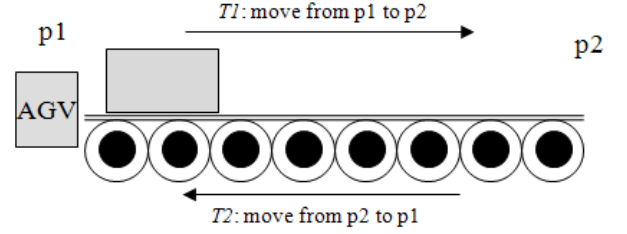
Set of tasks = {*T1*, *T2*}
Triggering signal of *T1*: *Signal_1*
Triggering signal of *T2*: *Signal_2*
Task execution time of *T1*: *Time_1*
Task execution time of *T2*: *Time_2*

Figure 5-(a): Task identification of a device
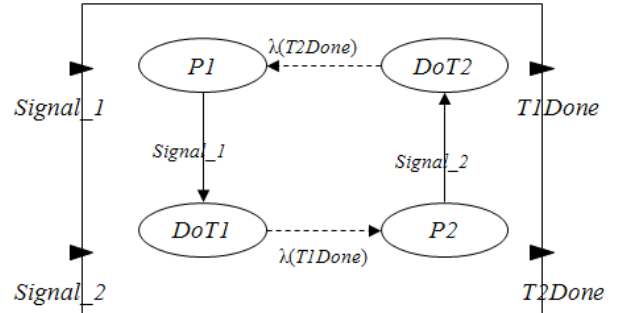
Figure 5-(b): State transition diagram

Once virtual device models are constructed, a plant model can be defined by combining the virtual devices. While virtual devices are described as atomic models, the entire plant model is modeled as a coupled model, including those atomic models and coupling relationships between them. Figure 4-(a) shows a simple example of a plant model including two virtual devices, VD₁ and VD₁. The DEVS couple model of the plant model, shown in Figure 4-(a), can be described as follows:

Plant Model
$$DN = < X, Y, M, EIC, EOC, IC, SELECT > \qquad (4)$$
$X = \{input1, input2\}$, $Y = \{output1, output2\}$
$M := \{VD_I, VD_J\}$
$EIC = \{(DN.input1 * VD_J.input2),$
$\qquad (DN.input2 * VD_I.input1)\}$

$EOC = \{(VD_I.output1, DN.output1),$

$\qquad (VD_J.output1 * DN.output2)\}$
$IC = \{(VD_I.output2 * VD_J.input1)\}$
$SELECT: 2^M - \emptyset \rightarrow M$: tie-breaking selector.

The proposed methodology for the construction

of a plant model has two major benefits. The first is the reusability of a virtual device model, signifying that the structure of a virtual device model achieves independence from the configurations of a production system. The second benefit is the intuitiveness in defining the state transition diagram of the virtual device model. Users with only a passing knowledge of discrete event system modeling can easily define a virtual device model simply by identifying the set of tasks.

4. EXAMPLES & ILLUSTRATIONS

The prototype of the proposed PLC programming environment was implemented and tested with several examples. The C++ language in a Visual Studio environment was used, with OpenGL for the graphical rendering. An extrusion system, shown in Figure 6, consists of four sub-systems; 1) a feeder system, 2) a ram-die system, 3) a start button and 4) a stop button. Extrusion is a compression process in which the work metal is forced to flow through a die opening to produce a desired cross-sectional shape. The process can be likened to squeezing toothpaste out of a toothpaste tube (Groover. 2006). The control logic of the extrusion system is as follows:

(1) If the '*I-start*' signal is triggered by a user when the system is idle, then the PLC sends out a signal to the input port of the feeder system, '*I-pick*' = TRUE.

(2) If the '*I-pick*' becomes TRUE, then the feeder system starts the job for picking a billet. The output port of the feeder system '*O-picking*' becomes TRUE until the end of the job. When the picking is finished, '*O-picking*' becomes FALSE.

(3) If the job of picking a billet is finished ('*O-picking*' = FALSE), then the PLC sends out two signals, '*I-pick*' = FALSE and '*I-extrude*' = TRUE.

(4) If the port '*I-extrude*' becomes TRUE, then the extrusion system begins to compress the work metal (billet) to make it flow through a die opening. The output port of the extrusion system '*O-extruding*' becomes TRUE until the end of the extrusion. When the extrusion is finished, '*O-extruding*' becomes FALSE.

(5) When the extrusion of a billet is finished ('*O-extruding*' = FALSE), the PLC then sends out a signal, '*I-extrude*' == FALSE, to make the system state idle.

(6) Whenever the '*I-stop*' signal is triggered by a user, then all activities of the system should be stopped.
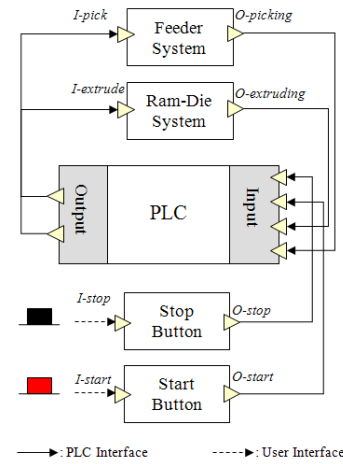
Figure 6: Configuration of an extrusion system

The control logic of the extrusion system should be converted into a PLC program for the implementation. Figure 7 shows the PLC program, written in the format of the SFC (Sequential Function Chart), which controls the extrusion system shown in Figure 6. The boxes along the vertical line (S1, S2, S3, S4 and S5) represent the states of the extrusion system, and the thick line segments between states (T1, T2, T3, T4, and T5) are transitions. Each transition has a ladder diagram on its left side that represents its firing condition. On the right side of the state boxes, there are additional boxes connected with a dotted line. Those boxes are the execution commands (R: reset, S: set) issued by the PLC that control the extrusion system. For the validation of the control logic, it is necessary to have a plant model enabling the visual validation of the PLC program. The plant model is shown in Figure 8, and it includes all devices (the feeder system, ram-die system, start button and stop button) of the production system as well as the corresponding device programs to perform their tasks in the production system.
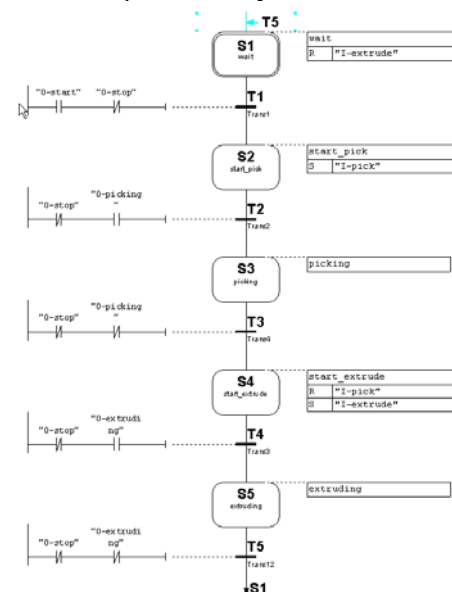

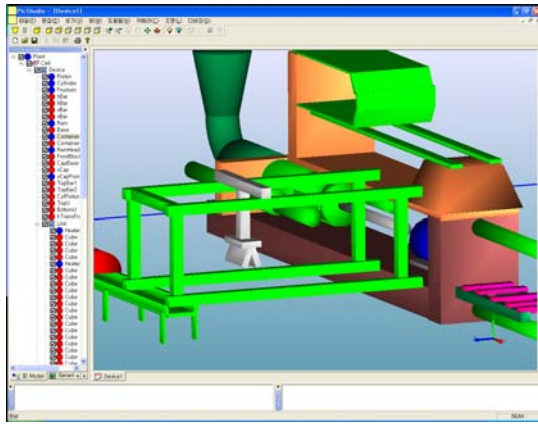Figure 7: PLC program that controls the extrusion system

414

Figure 8: Plant model of the extrusion system

5. DISCUSSION AND CONCLUSIONS

This paper proposes the architecture of a PLC programming environment that enables a visual verification of a PLC program by synchronizing a PLC program with a corresponding virtual plant model. The model layer of the proposed architecture consists of three models: a plant model (virtual factory model), a PLC program (control model) and an I/O mapping model. The plant model includes all manufacturing devices of the production system, and the PLC program contains the control logic for the plant model. The I/O mapping model functions as a communication link between these two models. As the plant model plays a key role in the proposed PLC programming environment, it is essential to develop a practical methodology in the construction of a virtual device model as well as a virtual plant model. To do so, this paper addresses an efficient construction method of a plant model based on the DEVS (Discrete Event Systems Specifications) formalism, which supports the specification of discrete event models in a hierarchical, modular manner. The proposed methodology for the construction of a plant model has two major benefits. The first is the reusability of a virtual device model, signifying that the structure of the virtual device model achieves independence from the configurations of a production system. The second benefit is intuitively the defining of the state transition diagram of a virtual device model. It is not necessary for users to have in-depth knowledge of discrete event system modeling, as they simply have to identify a set of tasks in order to define a virtual device model.

REFERENCES

Al-Ahmari AMA, Ridgway K. 1999. "An integrated modeling method to support manufacturing system analysis and design." Computers in Industry. No.38:225-238.

Anglani A, Grieco A, Pacella M. 2002. "Tolio T. Object-oriented modeling and simulation of flexible manufacturing system: a rule-based procedure.", Simulation Modeling Practice and Theory, No.10:209-234.

B.K. Choi, B.H. Kim. 2000. "New trends in CIM: Virtual manufacturing systems for next generation manufacturing", *Current Advances in Mechanical Design and Production*

Seventh Cairo University Int. MDP Conf., Cairo, February 15-17, 425-436

B. P. Zeigler. 1984. "*Multifacetted modeling and discrete event simulation.*", Academic Press, Orland .

Chuang C P, Lan X, Chen J C. 1999. "A systematic procedure for designing state combination circuits in PLCs.", Journal of Industrial Technology, No.15(3):2-5.

Groover M P. 2006. "*Fundamentals of modern manufacturing.*", Wiley.

Jang J, Koo P H, Nof S Y. 1997. "Application of design and control tools in a multirobot cell.", Computers and Industrial Engineering, No.32:89-100.

K. Iwata, M. Onosato, K. Teramoto, S. Osaki., 1995. "A modeling and simulation architecture for virtual manufacturing systems.", CIRP, No. 44(1):399-402.

Klingstam P, Gullander P. 1999. "Overview of simulation tools for computer-aided production engineering.", Computers in Industry, No.38:173-186.

L. Ye, F. Lin., 1997. "Virtual system simulation – A step beyond the conventional simulation", *22nd Int. Conf. on Computer and Industrial Engineering*, 304-306.

Manesis S, Akantziotis K. 2005. "Automated synthesis of ladder automation circuits based on state-diagrams.", No.36:225-233.

M. Onosato, K. Iwata., 1993. "Development of a virtual manufacturing system by integrating product models and factory models.", CIRP, No.42(1):475-478.

Rullan A. 1997. "Programmable logic controllers versus personal computers for process control.", Computers and Industrial Engineering, No.33:421-424.

T. G. Kim., 1994, "*DEVSIM++ User's Manual*", Department of Electrical Engineering, KAIST, Korea, 1994.

AUTHOR BIOGRAPHIES

MIN SUK. KO was born in Korea, Je-Ju Island and went to the Ajou University of Suwon, where he studied Industrial Information System Engineering. He worked for a couple of years for the Digital Manufacturing Lab in 2007 to the University of Ajou where he is graduated student. His e-mail address is : sebaminsuk11@ajou.ac.kr and his Web-Page can be found at : http://udmtek.co.kr.

SANG CHUL. PARK. Is a professor in the Department of Industrial and Information System Engineering at Ajou University. Before joining Ajou, he worked for Daimler- Chrysler Corp. and CubicTek Co., developing commercial and in-house CAD/CAM/CAPP software systems. His research interests include geometric engineering knowledge management, and discrete event system simulation. His e-mail address is : scpark@ajou.ac.kr and his Web-Page can be found at : http://udmtek.co.kr.

GI NAM. WANG Is a professor in the Department of Industrial and Information System Engineering at Ajou University. His e-mail address is : gnwang@ajou.ac.kr and His Web-Page can be found at: http://udmtek.co.kr.

DISCRETE EVENT SIMULATION OF NETWORKED PRINT PRODUCTION

Wolfgang Kuehn
Department of Electrical Information and Media Engineering
University of Wuppertal
D-42199, Wuppertal, Germany
E-mail: wkuehn@uni-wuppertal.de

ABSTRACT

In print media industries networked production gains increasing importance. The relatively young XML based communication standard (JDF/JMF) allows to specify workflow models and data exchange. The proposed concept of a simulation framework for modelling of networked print production is based on an object-oriented approach implementing modules for the resources and controls applied in this particular industry. The framework uses the JavaSim-simulation library and shall enable modelling of networked print production using the same JDF-/JMF-communication algorithms as applied in real-world system communication. This approach allows to integrate the framework in a digital factory concept using simulation for the factory planning phase and also for the operative planning and control of networked print production.

INTRODUCTION

The print media industry is facing tough competition. Overcapacity and sluggish demand are cutting margins. At the same time, customers' expectations regarding flexibility, quality, speed and reliability have never been higher. This situation calls for processes to be reviewed, organisation to be optimised and manufacturing costs to be cut even further. The various machines and processes have already been widely optimized in the past. Process integration and networked production are the leading directions for the next years. Simulation technology enables to analyse and improve the planning even of complex production networks.

In various industries there are already simulation approaches (Dangelmaier et al. 2006, Kapp et al. 2005, Mönch 2007, Yalcin and Namballa 2004) of networked production systems even in digital factory environments (Bley et al. 2005, Schloegl 2005, Wenzel et al. 2005, Westkämper et al. 2005). So what is especial in print production networks. In print media industries there are mainly small and medium businesses, large companies are exceptional and the production planning and control differs from other industries. The main difference compared to other industries is that production planning

performed is mainly job related and not product related. Often with start of planning a production job the product is not completely defined. Further in printing industries often the argumentation is used that there are no standardised products and each job differs from others. Print products are often not defined with a product structure containing a bill of material.

In print media industry the communication technology is changing. The newly developed vendor independent Job Definition Format (JDF) and Job Messaging Format (JMF) are established as a new industrial communication standard to networked print production (Kühn and Grell 205). Supported by the PDF document format a new generation of fully integrated network workflow solutions can be developed that enable process integration along the entire value added chain. Faster job preparation, clear job tracking and enhanced cost transparency are the goals. All job-relevant information can be distributed to the various production areas and all production data can be transfers to central control system. The fact that the production system has one general interface cuts the technical implementation workload and eases the maintenance outlay for networked print production considerably.

Discrete event simulation technologies enables to test, analyse and optimize networked production in advance. The use of discrete event simulation for modelling process in a print production workflow is not new (Bäck et al., 1997) (Nordqvist and Fällström 1996). Heidelberger Druckmaschinen AG has developed BizModel, a simulation software for printshop planning (Heidelberger Druckmaschinen AG 2008). BizModel takes the actual situation within a print shop and delivers options for optimized print shop development with regard to technical and business-management but the software doesn't cover detailed networked production aspects. For networked productions and new communication standards the demand has changed and an advanced approach for simulation is mandatory (Buckwalter 2005a/b). This paper presents a concept for a simulation framework focussed on networked print production. An important feature of this framework is the use of JDF-/JMF-communication algorithms in simulation, which is compatible to the real system communication.

JOB DEFINITION FORMAT (JDF)

The definition of JDF is based on the idea of creating a standardized, manufacturer independent and comprehensive data format for the print media industry to standardize information exchange between different applications and systems in and around the graphic arts industry. The concept for this industry standard originally was presented by Adobe, Agfa, Heidelberger Druckmaschinen and MAN Roland. In between it is maintained and developed by the International Cooperation for the Integration of Processes in Prepress, Press and Postpress Organization (CIP4), a not-for-profit association, which currently consists of more than 300 manufacturers, organizations and print service providers. Details are available at www.cip4.org. The CIP4 reference model is shown in Figure 1.
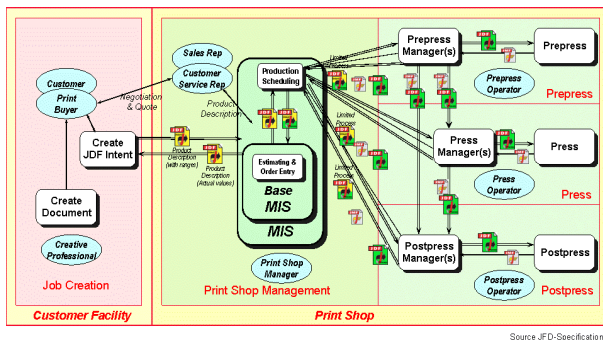
Figure 1: CIP4 Reference Model

- JDF is a comprehensive XML-based file format for job ticket specifications combined with a message description standard and message interchange protocol,
- JDF enables the integration of commercial and planning applications into the technical workflow,
- JDF provides a clear, fully-integrated structure for the entire workflow including the relevant target and actual data,
- JDF enables end-to-end production control, even across corporate boundaries,
- JDF enables both the horizontal and the vertical integration between order management, production planning and control, and production resources.

JDF has become an accepted and requested standard in the industry among suppliers and customers.

Due to the job related approach the JDF-structure (Figure2) contends product nodes, process nodes and group nodes in the same structure (CIP4 2005a). JDF also has an integrating effect on the different market segments of the print media industry. In the future, messages in both newspaper printing and commercial printing will be transmitted via a common standard. The establishment of a uniform, comprehensive standard significantly reduces the effort required to develop and implement networking solutions. Print service providers
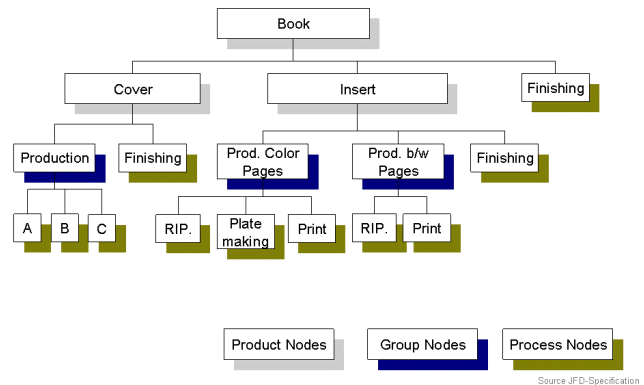
Figure 2: JDF-Structure

are able to deploy software applications from different manufacturers without having to make any compromises when it comes to networking. However, because the JDF specification (CIP4 2005a) is very comprehensive (more than 900 pages) it is not very useful for each single device to implement the complete JDF-specification in all details. Therefore subsets are designed.

Interoperability Conformance Specification (ICS)

For a simplified and clear communication Interoperability Conformance Specifications (ICS) are defined. (CIP4 2005c/d) These are well-specified subsets of JDF, each defining an interface between pairs of communication partners in the workflow.
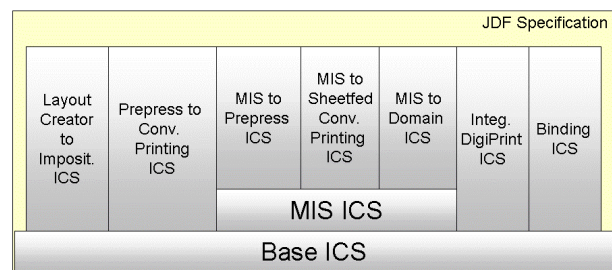

Figure 3: Interoperability Conformance Specifications

The ICS documents provide communication standards for individual classes of devices. ICS documents do not add to the JDF specification, but provide additional constraints and are specific to the particular interface. The ICS specification follows an hierarchical concept (Figure 3). In addition to a base ICS document that applies to all JDF-enabled devices, ICS documents have been published to cover a variety of interfaces, including:

- Binding
- MIS to Prepress
- Prepress to Conventional Printing
- MIS to Conventional Printing - Sheetfed
- Integrated Digital Printing
- Layout Creator to Imposition

ICS documents look at the interface between the "manager" of a job, such as a production planning system or a pressroom management system and the "worker", the system or software that will perform the desired process. The ICS documents define the roles of the "manager" and "worker", including their ability to read and write JDF, minimum support for JMF options. Additionally it is defining how job files are to be exchanged or identified, and how support for particular JDF processes and resources is required. When a JDF-enabled product meets the "manager conformance requirement" of a particular ICS, it achieves interoperability with other JDF-enabled products that meet the corresponding "worker conformance requirements" of the same ICS. The discrete event simulation of networked print production systems has to apply this ICS approach in order to be compatible with the real-world communication.

JOB MESSAGING FORMAT (JMF)

The Job Messaging Format (JMF) complements the Job Definition Format and is used to transfer messages. Up-to-the-minute information defined using JMF can be exchanged while production processes are in progress. Job lists are created using JMF to report on device capabilities (resolution, formats, etc.) and transmit status messages (CIP4 2005a). JMF uses the http transfer protocol.

CIP4 SOFTWARE DEVELOPMENT KITS (SDK).

The CIP4 Consortium makes free open source libraries for the programming languages C++ and Java available (CIP4 2006) These libraries are intended to make it easier for software developers to write JDF applications, reducing development times and therefore cutting costs. However in practise the use of the common libraries doesn't guarantee that JDF is always interpreted consistently. Due to the complexity of JDF the communication allows different XML's even using the specification correctly. A CIP4-certificate for tested communication interfaces is not available yet.

NETWORKING ARCHITECTURE

The Job Definition Format is a comprehensive standard for data exchange. But the JDF specification does not describe what information is to be maintained where, nor who has the right to access and, where necessary, modify it. Basically, the following logical components are provided in a JDF-based networking system:
- Agents are tasked with writing a JDF or extending or modifying an existing one,
- Controllers route a JDF to the specified location,
- Devices form the interface between software applications and machines. Devices interpret the JDF at their specific nodes and control the relevant machines,
- Machines are components of the workflow that are controlled by a device and carry out the processes.

The agents, controllers and devices should not just be able to process JDF though. They should also provide a JMF interface with bidirectional communication mechanism for messaging. Figure 4 shows a JDF-/JMF-Device. The input for the job to be performed is communicated via JDF. JMF messages are sent and received during the process. After processing the JDF is sent back containing the actual data of particular job processing.
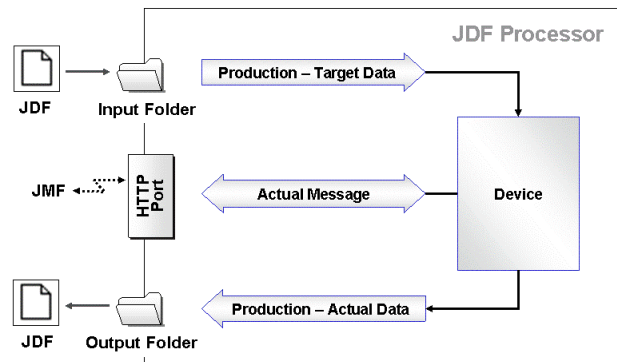

Figure 4: JDF-/JMF-Device

Due to the complexity of production it is not very useful to transfer the JDF-File as a job ticket just from one resource to another. The following production scenario illustrates quite clearly a typical example in networked production: In a particular production workflow, a print job is to be split over three different presses, two of which are located within the company and a third at a cooperating print service provider. Finishing is to be carried out with batches overlapping on two different production lines, i.e. finishing work is to start before all the sheets have been printed. Due to deadline problems, the decision to use a second line for finishing is taken at short notice and leads to a revision in planning after printing has started. This situation shows that a sequential transfer of a job ticket can't be an effective solution. In fact on the control level there is a powerful software required, dealing safely with all these cases.

The networking architecture must comply with the following requirements, among others:
- Secure and unambiguous regulation of data transfer and data management
- Data consistency must be maintained,
- Fast data transfer and short reaction times must be guaranteed,
- Unnecessary data transfer must be avoided,
- The entire system must function without error, even when products from different manufacturers are integrated in a workflow,
- The system should include robust mechanisms to make sure that processes are not aborted if problems arise.

SIMULATION OF NETWORKED PRODUCTION

Simulation is a powerful tool to improve decision-making on a reliable base. Network projects are very complex. These networked print production projects don't focus on one particular area of a company only; all areas and levels of a company and even the network between companies have to be involved. A company can't just buy a network. To establish a networked production requires to analyse and standardise most of the processes in a company or respectively in all involved companies. This is an enormous effort. In practise it takes nearly two years to establish a network production project in a medium size printing company with 100 people. The experience shows that there are a lot of problems in detail, which are seen first during the praxis test at the customer's place.

Simulation shall enable to model print production networks in advance, to identify good process strategies and to find problems at an early state. This improves and accelerates the benefits available in networked production systems.

Simulation of networked job preparation

Simulation of networking job preparation shall improve benefits such as:
- Reduce job throughput times,
- Eliminate inefficiencies due to duplicated data capture in different software applications,
- Reduce costs incurred due to errors by means of standardized designations, up-to-date order information and previews.

Simulation of networked machine presetting

The presetting produced digitally in order management or prepress can be used to set up electronically controllable machine components. Simulation helps to test and improve the networking machine presetting operations:
- Reduction in setup times,
- Reduction in waste, improvement in production reliability and quality,
- Reassignment of the work to just a few specialists.

Simulation of networked production planning and control

Simulating the networking production planning and control shall improve benefits, such as:
- Eliminating duplicated inputs,
- Eliminating follow-up questions on the production status,
- Reducing the effort required for planning meetings,
- Increasing the reliability of production by ensuring that paper, ink and plates are available when needed at the correct press,
- Increasing efficiency by a feedback loop based on actual data.

Simulation of networked operating data logging and actual costing

Simulation offers to check and analyse networking operating data logging and actual costing:
- Eliminating duplicated inputs and checks,
- Providing prompt status messages and high-quality management information,
- Recording the actual costs.

Simulation of the networked colour workflow

Networking the colour workflow can be checked and improved by simulation:
- More productivity from the first step to the finished print,
- Increased reliability and consistency in the print quality, true-colour proofs,
- Shorter setup time and less waste,

Simulation of customer networks

In print production the integration of the customer in networked solutions has a lot of benefits. By use of simulation these networks can be tested and improved:
- The customer's project members have a complete overview of the project information they need. This includes the product description, the time and place of delivery for order tracking and access to the stock system for requisitioning preprinted materials.
- Accelerated communication thanks to digitalisation of ordering processes with agencies and print service providers.
- Deliver management information for customers covering inquiries, print orders issued, etc. This represents a source of quick data for checking budgets.

Evaluation of the investment decision

Process integration based on the Job Definition Format shall help to eliminate inefficiencies in the production process. This benefit is achieved at the cost of investments in software, hardware and services. In addition, management and the staff undergoing training are tied up to a significant degree during the introductory phase of networking. The benefits and cost of networking must be weighed against one another depending on the order and operating structure. Discrete event simulation can support the decision-making process.

For the implementation of production networks the technical and commercial aspects have to be considered. Simulation allows to analyse technical aspects, such as resource layout and communication architecture. Further input data for the cost calculation can be validated. The payback period is determined by considering two areas separately. These are, on the one hand, networking of job preparation, production

planning and control and operating data logging and actual costing – grouped under the term operational data networking – and, on the other, networking of machine presetting and colour workflow.

SIMULATION FRAMEWORK

In order to enable models with acceptable effort and sufficient level of detail a concept for an applied simulation framework shall be introduced. A simulation framework focussing on print production networks has to fit the specific requirements. Goal of the Java-based simulation framework is to model the production and information flow including production planning and control in detail.
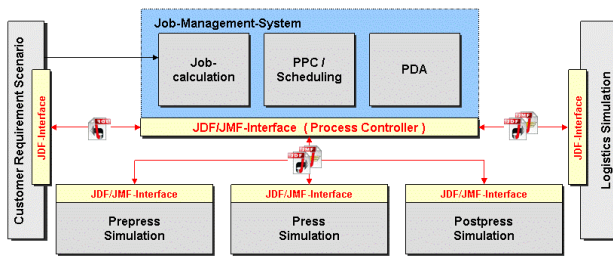
Figure 5: Simulation Communication Structure

In order to obtain usable results the communication between the resources shall be compatible with the real-world communication. Figure 5 shows an overview of the principal communication structure between the job management system and the simulation modules. The job management system takes a job, a JDF instance, from the job queue. It simulates the execution of the job in the particular area, consuming input resources and producing output resources accordingly and then sends the updated JDF instance for further processing in the workflow. During process execution, all state changes within the simulation model are sent to the production planning and control systems monitoring the process or tracking the job. In the first approach the production processes, such as printing, cutting, folding or needle stitching have not be modelled in detail. The simulation framework concept is based of the JavaSim library developed at the University of Wuppertal (Kuehn 2002) and the Java-based JDF Communication Library available at CIP4 (CIP4 2006).

Framework Concept

In the proposed framework concept all modules of resources shall be seen as an object containing modelling the physical behaviour of the resource including the low level control and an object containing the communication interface. The communication with the resource is performed through the communication interface only and the communication interface has to be compatible with the real-world communication, at least on a certain level.

The clear separation of modelling the material flow, the controls and the communication will lead to a better integration of simulation into the planning and implementation processes. For the integration of simulation in digital factory environments this is very important. Compatible interfaces between simulation modules and the control systems enable to switch between simulation modules and system components.

By use of compatible communication it is also possible to build an entire print production workflow using only simulated processes, but use a real production planning and control systems to track and control the processes in the workflow. Such a setup allows to experiment with configurations of production systems very realistically because the real control logic of the real production planning software is applied. Further in simulation the effort can be reduced significant, because the control system has not to be emulated in the simulation model itself.

This framework concept can be applied in the digital factory environment to plan and to evaluate complex networked print production systems. In a second stage it even offers the possibility to enhance the operative production planning and control as an integrated process from the top level to the factory floor control.
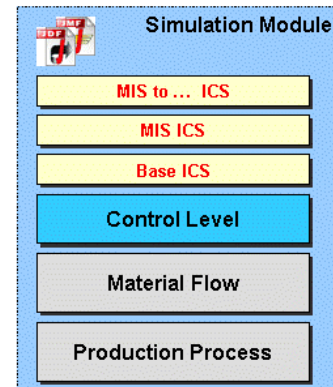

Figure 6: ICS-Module

Figure 6 shows the principal structure of a simulation module which performs a MIS to … communication. Based on the simulation of the production process and the material flow, the control level links the physical behaviour of the module to the communication level with Base ICS, MIS ICS and MIS to … ICS. This concept using compatible communication structures, by applying JDF-/JMF-communication to simulation modules and real production environments as well, allows even to integrate and simulate heterogeneous systems in the print production workflow.

JavaSim Library

The JavaSim library is an advanced simulation library, developed for the simulation of discrete event processes

(Kuehn 2002). The library contains central classes such as the *GeneralSimModel* with the *SimController* and the *ModelStructure*, the *CtrlObject* or the *InterfaceObject*. The JAVA-Sim Library follows an object-oriented design and offers general simulation objects, material flow objects, information flow objects, decision-making objects and applied material flow objects. The core object of each simulation model is the *SimController* which has to handle all events correctly. From the general class *GeneralSimModel* applied models such as a *SimModelPress* or a *SimModelPrePress* can be derived. An hierarchical model structure can be established using the class *Modul*.

The JAVA-Sim Library is not fixed on a certain simulation level like many simulation tools in the market. With its object-oriented approach it is open and modular. The library offers features similar to the flexibility of a simulation language, over the level of general building blocks, such as workstations and assembly stations, up to the efficiency of simulators with special building blocks. The JAVA-Sim Library strictly follows an object-oriented design. It contains public simulation classes, made available for the user of the simulation library, and internal simulation classes and utility classes for internal use in the library only. The library offers general simulation objects, material flow objects, information flow objects, decision-making objects and applied material flow objects. The internal class *InfFlowObject* is used to implement information flow. Special information flow elements may be derived from this class. The class *SimEventObject* is based on the class *GeneralObject* and contains the additional functionality for creating simulation events. From this class the general classes *EventGenerator* and *MathFlowObject* and *MathFlowObjecExt* are derived. Again from these the detailed material flow classes such as *Buffer*, *BufferFifo*, *BufferLifo* etc. are derived. For all simulation classes there are some general class features available, e.g. such as create object, delete object, constructor control and destructor control.

This JavaSim library with general material flow and control objects is available at the institute. It can be applied as a base for the simulation framework. Objects specific to print production networks have to be inherited from the general objects available.

JDF-/JMF-communication

The communication between modules in the networked print production simulation framework shall be compatible with the real word communication. Modelling of the network communication devices, such as resources, machines, etc. requires compatible JDF-/JMF communication structures. Therefore the Java-Sim library shall be extended by use of an JDF-/JMF-communication library. This communication library is available as open source at CIP4 (CIP4 2006). In order to be compatible with the real-world communication the simulation of networked print production systems has not only to apply just the JDF-/JMF library but even the complete ICS approach. However these ICS modules can be implemented stepwise.

Level of detail

In the framework the modelling level of detail regarding communication is high in order to be compatible with the real word communication. The level of detail regarding material flow simulation shall be very flexible. If the simulation focus is on communication and controls the material flow model level of detail can be on a relatively low level. In this case resources are just time-consuming objects, errors are produced in a deterministic sequence or randomly in order to enable the test of communication error messaging. For certain modules within a model this level of detail might be increased step by step. With a next level of detail it might be even possible to divide the physical behaviour and the low level control for certain resource module. Principally the framework concept is open for these kind of approaches, however the focus here is on communication and not on detailed processes or detailed process controls.

The advantage of the applied hierarchical framework concept is, that a simulation can be performed on a very low level of detail first in order to test the communication flow. In a second step the level of detail can be increased in order to check the dynamic of the modelled production network. And with a next step it is possible to increase the level of detail for bottleneck areas again in order to make really sure that these critical areas fulfil the demand sufficiently.

SUMMARY

The proposed simulation framework concept focuses on modelling networked print production. The use of compatible communication structures has the advantage that simulation and real-world modules can be combined in a test environment. This approach allows to integrate the framework in a digital factory concept using simulation not only in a factory planning phase but also for the operative planning and control of networked print production. The simulation framework is realised platform-independent in Java. The existing JavaSim simulation library from the University of Wuppertal and the open source JDF-/JMF communication library available from CIP4 are the base for the implementation. At this state these libraries are available and first tests have been performed. The next step is to implement and validate the specific objects of the framework and to extend the functionality and levels of detail stepwise. On each level tests with specific scenarios of networked print production have to be performed in order to validate the framework modules.

REFERENCES

Bäck, A.; Lehtonen, T.; Karttunen, A.; Kuusisto, M.; Launonen, R; 1997 "Evaluation of Printing Production in Networks by Means of Computer Simulation Models", *Advances in Printing Science and Technology*, vol. 24, 177–190

Bley, H.; Zenner, C.; 2005. "Coupling of Assembly Process Planning and Material Flow Simulation", *Proceedings of the 2005 IEEE International Symposium on Assembly and Task Planning (ISATP2005)*, Montréal, Canada, 2005

Bley, H.; Franke, C.; Zenner, C., 2003. "Integrated Data Management and Variant Management – Milestones on the Way to the Digital Factory", *Production Engineering*, Volume X/1 (2003), *Annals of the German Academic Society for Production Engineering*, 105-108

Bley, H.; Zenner, C.; 2005. "Handling of Process and Resource Variants in the Digital Factory", *CIRP Journal of Manufacturing Systems*, Vol. 34 (2005), No. 2, 187-194

Bley, H.; Franke, C.; Zenner, C., 2005. "Variant Management in Production Planning", *CIRP Journal of Manufacturing Systems*, Vol. 34 (2005), No 1, 1-8

Buckwalter, C.; 2005a. "A JDF-enabled Workflow Simulation Tool", *Proceedings, TAGA 2005, 57th Annual Technical Conference*, Toronto, Canada, 271-281

Buckwalter, C.; 2005b. "A Tool for Testing Compliance with CIP4's Interoperability Conformance Specifications.", *Proceedings of 2005 International Conference on Digital Production Printing and Industrial Applications(2005)*, 83-84

CIP4, 2005a. "JDF Specification Version 1.3", www.cip4.org/documents/jdf_specifications/JDF1.3.pdf [2008, Feb. 09]

CIP4, 2005b. "XML Schema for JDF Version 1.3" www.cip4.org/Schema/JDFSchema_1_3/JDF.xsd [2008, Feb. 09]

CIP4, 2005c. "Base Interoperability Conformance Specification" (Base ICS) www.cip4.org/document_archive/documents/ICSBase-1.0.pdf [2008, Feb. 09]

CIP4, 2005d, "ICS Registry" www.cip4.org/document_archive/ics.php, [2008, Feb. 09]

CIP4, 2006, "Java SDK", www.cip4.org/open_source/index. [2008, Feb. 09]

Dangelmaier W.; Aufenanger M.; Mahajan K.; Laroque C.; Huber D.; 2006. „Event simulation of supply chain networks - Dynamic detailing in the material flow simulator d3FACT insight.", *ESM, European Simulation and Modelling Conference*, (2006) 50-54

Fritz J.; Hellener G., 2006. "Planning data preparation - Key requirement for automated simulation modeling" *CIRP International Seminar on Manufacturing Systems*, 39 * (2006)

George, L.; Kovacs, A.; Paganelli, P.; 2003. "A planning and management infrastructure for large, complex, distributed projects beyond ERP and SCM", *Computers in Industry* Volume 51, Issue 2, Virtual Enterprise Management, June 2003, 165-183.

Heidelberger Druckmaschinen AG, 2008, "Print Shop Planning: Optimizing Business Processes Through Simulation", www.heidelberg.com [2008, February 10]

Kapp, R; Löffler, B.; Wiendahl, H.-P. Westkämper, E.: 2005. "The logistics bench: Scalable logistics simulation from the supply chain to the production process." *CIRP Journal of Manufacturing Systems*, 34 no. 1, 45-54

Kühn W., 2002, "JAVA-Sim - An advanced Discrete Event Simulation Library", *SCSC 2002, Summer Computer Simulation Conference*, 2002, San Diego, USA, p. 393-397

Kühn W., Grell M., 2005, "JDF, Process Integration, Technology, Product Description" *Springer 2005*, ISBN 3-540-23560-4

Mönch L.; 2007, "Simulation-based benchmarking of production control schemes for complex manufacturing systems.", *Control Engineering Practice*, Vol. 15 (2007) No. 11, 1381-1393

Niemann J.; Westkämper E.; 2006. „Dynamic life cycle performance simulation of production systems.", IPP, *CIRP International on Life Cycle Engineering Conference*, 12 * (2006) 419-428

Nordqvist, S.; Fällström, F.; 1996. "Simulation of Newspaper Production Processes - Decision and Management Support Tools", TAGA 1996 Conf. Proc., 299–317

Schloegl, W. 2005. "Bringing the digital factory into reality - virtual manufacturing with real automation data." *CARV, International Conference on Changeable, Agile, Reconfigurable and Virtual Production*, 187-192

Vancza, J.; Kis, T.; Kovacs, A.; 2004. "Aggregation - the key to integrating production planning and scheduling", *CIRP Annals - Manufacturing Technology* Volume 53, Issue 1, 2004, 377-380.

Venkateswaran, J.; Son, Y.-J.; Jones, A.: 2004. "Hierarchical Production Planning Using a Hybrid System Dynamic-Discrete Event Simulation Architecture.", *Proceedings of the 2004 Winter Simulation Conference*, 1094–1102

Wenzel, S.; Jessen, U.; Bernhard, J.; 2005, "Classifications and conventions structure the handling of models within the Digital Factory," *Computers in Industry*, Netherlands, vol 56, no 4, 334-46

Westkämper, E.; Gottwald, B.; Fisser, F.; 2005. "Migration of the digital and virtual factory to reality." *CIRP Journal of Manufacturing Systems*, no 34(5), 391-396

Yalcin, A.; Namballa, R. K.; 2004. "An object-oriented simulation framework for real-time control of automated flexible manufacturing systems", *Computers & Industrial Engineering* 48 (2005) 111–127

AUTHOR BIOGRAPHIES

WOLFGANG KUEHN studied mechanical engineering at the University of Brunswick, Germany. Then he worked with Blaupunkt for two years. At the University of Bremen he got his PhD in production engineering in 1991, and his habilitation in the area of simulation of production systems in 1997. From 1993 to 1995 he worked as Associate Professor at the Asian Institute of Technology in Bangkok. In 1996 he founded the SIPOC Simulation based Planning, Optimization and Control GmbH in Bremen. Since 1997 he has been working as a professor for production planning and control in the department of Electrical, Information and Media Engineering at the University of Wuppertal. His e-mail address is: wkuehn@uni-wuppertal.de and his Web-page can be found at http://wdppc1.kommtech.uni-wuppertal.de/joomla/

BUFFER MANAGEMENT FOR AUTOMATED MATERIAL HANDLING SYSTEMS IN SEMICONDUCTOR INDUSTRIES

Christian Fischmann, Fabian Böttinger, Roland Wertz and Christian Kunz
Department of Ultraclean Technology and Micromanufacturing
Fraunhofer Institute Manufacturing Engineering and Automation (IPA)
Stuttgart, Germany
E-mail: fischmann@ipa.fraunhofer.de

KEYWORDS
Discrete Event Modeling and Simulation in Logistics, Transport and Supply Chains

ABSTRACT

The automated material handling system (AMHS) is a highly automated transport solution of actual 300mm semiconductor waferfabs. The big logistic challenge in such a running system is the just-in-time delivery of lots (carriers). Therefore, beside tracks and vehicles, there are different storages (stocker) and buffers (under track storages and load ports) integrated in such a transport system. To conquer the logistic challenge, an optimum allocation and control strategy for all storage equipment is required. This paper describes the investigation of different storage and buffer strategies based on detailed simulation models of a dedicated semiconductor factory area.

INTRODUCTION

At present the most expensive investment costs of a waferfab is the process equipment. Therefore, the crucial key performance indicator of the process and automation engineers is the equipment utilization (Sokhan-Sanj et al. 1999). To achieve this objective, each wafer carrier (lot) has to be in the right place at the right time. Therefore an effective use of the extensive material transport systems with a dynamic storage component control is obligatory. The storage components in an AMHS are segregated to:

- Load ports,
- Stockers and
- Under-track-storages (UTS).

The primary buffer is the tool load port (LP), directly connected with the process equipment and mostly limited to 3 or 4 ports. To overcome this limitation, other buffers or storages are essential. One possible alternative can be a stocker. A Stocker is an automated high-rack storage area with manual and automated in- and output ports and conveyors. Once the carrier is inside the stocker a robot is handling it into a shelf (rack bin location). To minimize the needed cleanroom footprint, the height of the stockers is often increased to the maximum possible height. (Csatáry 2002) Beside the footprint, the stockers have another significant disadvantage. The dedicated stocker is usually far away

from the process equipment, where the lot is needed. This might lead to very long delivery times (DT) and as result to a drop in the equipment utilization (Hunter and Humphreys 2003). Additionally, stockers need valuable clean room area on the factory floor.

In order to resolve the specified disadvantages, so called under-track-storages (UTS) were developed. These UTS are single buffer storages, mounted over the head, under the rail-system, Fig. 1.
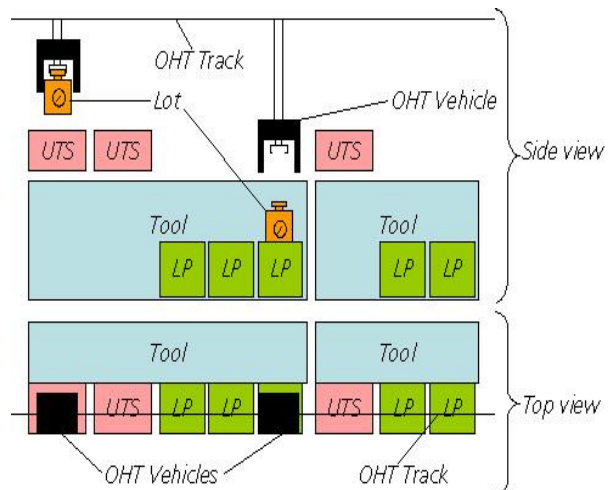
Figures 1: Example of UTS position

As the name already implies, UTS require no place on the ground of a waferfab. Furthermore the UTS are passive shelves with a lower cost of ownership due to reduced capital cost, increased reliability and elimination of scheduled maintenance in comparison to stockers (Antwerp 2004). Based on this information, the idea of the authors was to design a simulation model with fewer or no stocker transports, but with a lot of buffer application on base of UTS.

This paper presents a detailed simulation model of a dedicated waferfab area using aggregated process flow data. With this simulation model it is possible to:

- Execute different storage strategies in a short time due to small changes in a variable, adaptive source code
- Investigate probable efficient UTS placements
- Analyze possible control methodologies of the UTS allocation

The following section gives an overview of the basic data and describes the architecture of the fully integrated model. Section 3 explains the setup of system control. In section 4, the result with an optimized UTS allocation strategy is presented. Finally, section 5, concludes with an executive summary.

BASIC DATA

The scenarios are simulated in Applied Materials AutoMod™ Version 12.0 on two different models. Both models are designed as unified track layout systems. That allows tool to tool moves and eliminates the need for stocker transfers to route carriers between loops.

As mentioned in the abstract the authors use a less detailed model of a fab-segment. For that reason a stationary input and output is selected. At the input lots are generated and sent into the system. From here the lots are picked up by vehicles and routed through the whole system, including several process steps. Finally they are brought to the output, where they are unloaded and virtually destroyed. By that input and output represent the interface to other fab segments that are usually present and interacting with this segment in real life fabs. The layout of the models is oriented on similar models of already existing waferfab-segments with a unified system including highways (also known as main tracks having no load ports) and bay architecture.
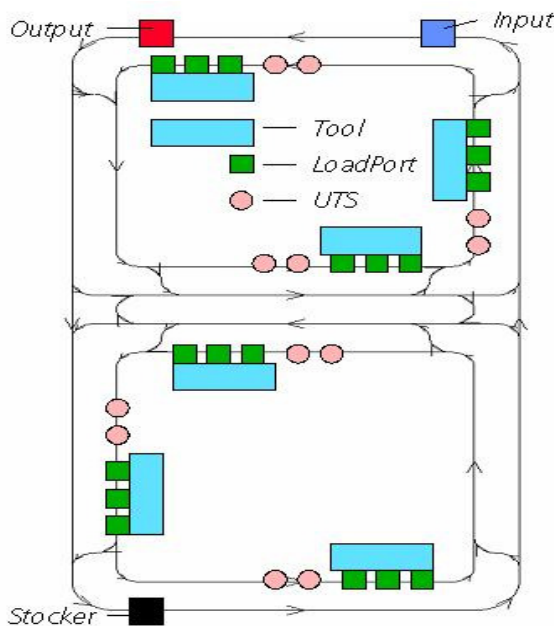
Figures 2: Exemplary extract of model 1

The first model has the UTS groups placed directly in front of the LPs as shown in Fig. 2. This model is used to analyze scenario 1, as it has a fixed allocation of ZFS groups to dedicated Tools.

In opposite to this, scenario 2 and 3 are evaluated on model 2 which has its UTS groups arranged on highways, see Fig. 3.
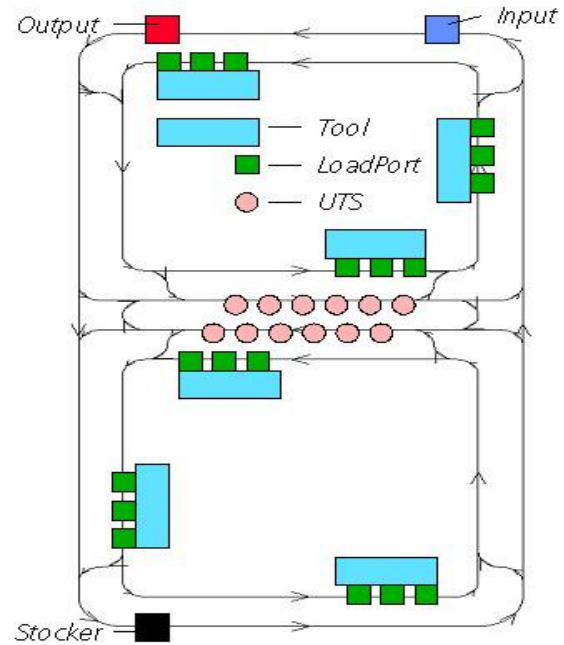
Figures 3: Exemplary extract of model 2

The only scenario including a stocker is scenario 4 which can run on both models. To be able to analyze the differences between UTS and Stocker based solutions, the UTS cannot be used as a storage facility in this case. The stocker is placed on the outer edge of the layout to simulate a longer travel distance to the process tools which accords to situations in reality.

In all scenarios 5 ToolGroups (TG) are represented, always based on the same basic data which is shown in Table 1.

Table 1: Basic data of all scenarios

TG	Tools	LPs	Process times / deviation [minutes]	Destination probabilities
1	5	3	8.9 / 2	100%
2	8	3	30 / 9	47%
3	2	2	4.6 / 0.4	66%
4	7	2	15 / 4	80%
5	6	2	10.5 / 3	100%

Destination probabilities represent the probability for one lot, to be transferred and processed at the corresponding ToolGroup. This approach simulates the existing variety in process plans and enhances model dynamics. Other basic data for all scenarios are:

• The start-cycle with 1.8 minutes per lot (Carrier) with a deviation of ±0.9 minutes.

- Each scenario includes 480 hours measurement 240h warm up and 240 hours in steady state
- 20 vehicles in use with 10 distributed parking positions
- 100% availability (no downtimes, no maintenances, no failures) of all transportation and process equipment
- One stocker

The difference between scenario 2 and 3 is the allocation of the UTS. Scenario 2 has the same allocation as scenario 1 which is dedicated to specific tools. In scenario 3 the UTS are located as in scenario 2 but are dedicated to the whole bay they are located in. This means that this UTS buffer can be used for any tools located in this bay.

Table 2: UTS capacity for each ToolGroup

TGs	# of UTS for Sc. 1 and 2	# of UTS for Sc. 3
1	20	80
2	24	92
3	12	34
4	28	74
5	24	80

SET UP OF SYSTEM CONTROL

All LPs, UTS and the output in the system are designed with a finite capacity of 1. In contrast to this the stocker is not limited in its capacity to be able to get a detailed view on the required capacities and to monitor the level of Work In Progress (WIP). At the input an unlimited queue is implemented in order to analyze the performance of the transport system in dependence of the scenario simulated. The process tools in the tool groups are equipped with one process chamber each, which allows the processing of one whole lot at the same time.

A lot is generated at the input. Depending on the probabilities given in the process flow the next tool group for this lot is selected. If there is a tool LP available in the selected tool group, it will be reserved and the lot will be delivered there. If there is no tool LP available an intermediate storage location has to be chosen. In scenario 1 to 3 a suitable UTS location is selected, if available. If there is no remaining space in the UTS – or in scenario 4 - the stocker is chosen. As soon as any tool is finished the next waiting lot is scheduled for this tool. This approach is based on a FIFO rule to ensure that no lots can be "forgotten" in any storage. The system is based on 20 Overhead Hoist Transport (OHT) vehicles which are located randomly in the layout at the beginning of each simulation run. As soon as there is an arising transport job the idle vehicle which is closest to the lot is dispatched to it.

If there is no vehicle available at this moment the transport job is queued in an infinite job list.

Vehicles that are not involved in any transportation task (deliver, retrieve or on its way to deliver or retrieve) are dispatched to any of the parking positions which are spread randomly in the layout.

Results

The overall performance of an AMHS can be evaluated by several metrics (Sturm et al. 2004).
On the one hand these metrics are time values like

- Average transport time (AvTT)
- Average delivery time (AvDT)
- Average waiting time for transport (AvWT).

These three time metrics are built up with several time stamps at each status change of the simulation. Fig. 4 shows how these times are calculated by means of status changes.
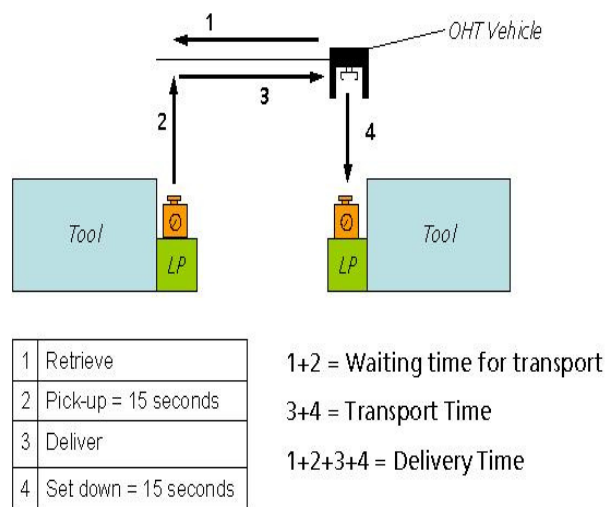
Figures 4: Time segments of move events

Within a simulation runtime of 480 hours (240 hours warm up and 240 hours results) these time segments are logged and analyzed. For the delivery time analysis of the AMHS only transport values by destination view are evaluated. Due to the necessarily high utilization of cost-intensive tools these transports need to be optimized. This means transports from tool to storage locations are not considered in the investigations.
On the other hand the performance metrics are capacity metrics like

- Vehicle utilization
- Stocker component utilization
- UTS utilisation
- System throughput MPH (moves per hour).

Basically, the vehicle utilization is divided into 4 statuses: "Deliver", "Retrieve", "Going to park" and "Idle". Based on project experiences at the Fraunhofer IPA the accumulated utilization percentage of "Deliver" and "Retrieve" should not exceed 65%. For all scenarios the average tool utilization is between 79% and 95%.

Results of Scenario 1

The throughput of scenario 1 results in 52950 moves in total with approximately 26% tool-to-tool (t-t) moves and 25% UTS-to-tool (u-t) moves. During the whole simulation run of this scenario there was no stocker move, which means the total storage is handled by UTS only. From all simulation runs, this scenario obtains the fastest average delivery time (69sec) for u-t moves, mainly caused by the short transport time of 18 seconds, see Fig. 5.
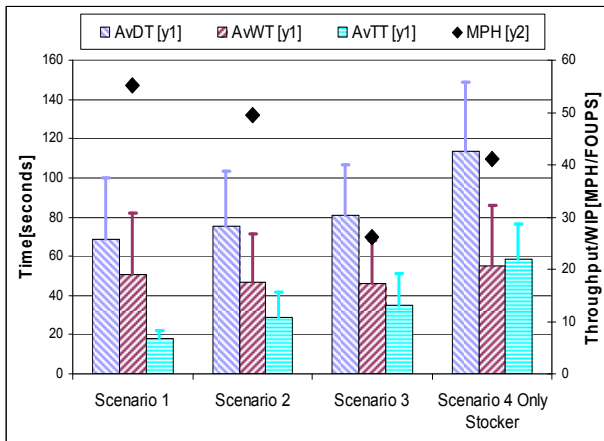
Figures 5: Delivery Time Analysis of all
u-t (Sc. 1-3) and s-t (Sc. 4) moves

Nevertheless the impact on the delivery time performance of all moves is marginal (see Fig. 6), caused by the relatively long average delivery time of 97 seconds for t-t moves.
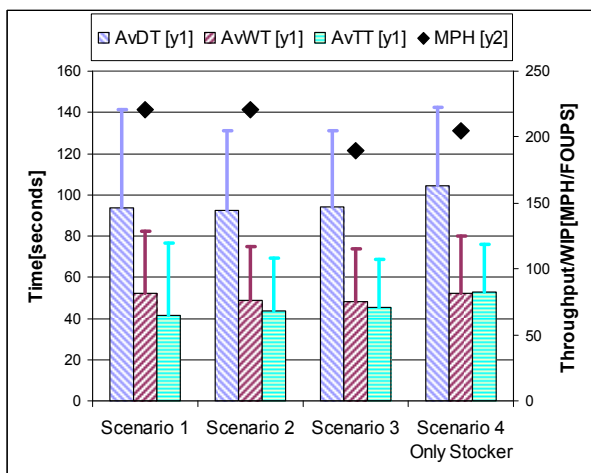
Figures 6: Delivery Time Analysis of
all moves

It seems that the UTS bin positions directly in front of the tool LPs lead to a kind of "stop-and-go" behaviour of the vehicles on the track. Each time when a vehicle wants to travel to a LP behind an active UTS it has to wait for the whole hoist operation of the vehicle in front,

which is currently picking up or setting down a load at the UTS. This results in the highest peaks for transport, waiting and delivery times of all scenarios except scenario 4. The vehicle utilization analysis of all scenarios shows a low utilization performance of less than 30% of delivering and retrieving statuses. However the number of OHT vehicles is not reduced to avoid the bottleneck on the vehicle side.

Results of Scenario 2

The second scenario results in 53047 moves after the warm up period with approximately 27% t-t moves, 22% u-t moves and about 3% stocker-to-UTS (s-u) moves. The reason for some lots moving to the stocker instead of tool or UTS is the bottleneck TG 2. However, this simulation run obtains with 92 seconds the fastest delivery times and the second fastest transport times (44sec) concerning all moves. This could be caused by stocker traffic which results in slightly less vehicle appearance within tool bays. This advantage in delivery times seems to be a disadvantage, resulting in more overall system traffic.

Results of Scenario 3

Scenario 3 obtains with 45583 moves the lowest throughput of all scenarios, based on only 14% u-t moves but 39% direct tool delivery (t-t moves). However, the analysis does not identify a bottleneck in the AMHS. The reason for this seems to be the different UTS allocation of this scenario. The high number of t-t moves indicates an improvement in traffic flow at tool LP as well as around the UTS.

For all moves, the average delivery time (94sec), average transport time (46sec) and average waiting time (48sec) is comparable with scenario 1 and 2. However, the simulation run obtains the longest delivery times, compared to the two other UTS scenarios for u-t moves. The reason seems to be the longer transport time of 35 seconds based on the changed UTS position in comparison to scenario 1 and the UTS allocation compared to scenario 2. The UTS allocation leads to a small increase of travel distances for the transport action due to the UTS group used by all tools within the bay.

Results of Scenario 4

The "only stocker" simulation, scenario 4, reached a throughput of 49232 moves with about 32% t-t moves and 20% s-t moves. Independent from the point of view (all moves or just the moves from tools as destination) the scenario needs the longest times of all scenarios. The reason could be found in the position of the stocker at the edge of the model. A closer look at the results shows a significant increase in delivering time by approximately 10% and the transport time by 20%. The reason for that are the higher times for the s-t moves in comparison to the u-t moves of the other scenarios, shown in Fig. 5. Surprisingly, the moves per hour are the second smallest of all scenarios.

Conclusion

Due to expensive investment costs of the process equipment in a waferfab, a just-in-time transportation of the process material is often the most relevant logistic key performance indicator. To achieve this indicator, not only an optimized transportation is necessary. Also the storage of the lots in a best possible solution is required. To investigate this challenge, the authors present a detailed simulation model of a typical waferfab area. Within this paper, different storage and allocation strategies are presented and analyzed by means of this simulation model.

The analysis of the different storage scenarios gives the impression that - for high throughput AMHS - buffers (like UTS) are definitely required. Nevertheless, to cover all storage peaks by UTS without any stocker seems unrealistic. The placement of UTS bins relies on availability of AMHS track area, vehicle traffic in front of tools or buffers and the process speed of the associate equipment. Additionally, the allocation of UTS-Groups for a dedicated area and not only single bins for each tool seems helpful for the system performance as well.

REFERENCES

Antwerp, K. V. 2004. "Automation in a Semiconductor Fab". *Semiconductor International*, 12/1/2004.

Csatáry, P. and K. Kümmerle. 2002. " 300mm Fab Conversions & the Chemispherical Challenge - a Model Case Study". In *Semiconductor FabTech 17th edition* 09/2002

Hunter, R. and C. Humphreys. 2003. "Trends in 300 mm Factory Automation". *Semiconductor International*, 6/1/2003.

Sokhan-Sanj, S.; G. Gaxiola; G. T. Mackulak; F.B. Malmgren. 1999. "A Comparison of the exponential and hyperexponential distributions for modeling move requests in a semiconductor fab". In *Proceedings of the 1999 Winter Simulation Conference* (Phoenix, AZ, Dec.5-8). IEEE, 774-778 vol.1.

Sturm R.; K. Redig; D. Glüer. 2004. "Robustness Evaluation and Improvement Method for Automated Material Handling Systems." *Semiconductor Fabtech, 21st Edition* Henley Publishing Ltd, London, 38-47.

AUTHOR BIOGRAPHIES

CHRISTIAN FISCHMANN is project manager and researcher at the department Ultraclean Technology and Micromanufacturing of the Fraunhofer Institute for Manufacturing Engineering and Automation, Stuttgart, Germany. His main field of work is the area of production logistics. He has been contributing to a number of international projects in semiconductor industry and to a growing number of projects in photovoltaic industry – successfully targeting challenges such as planning new production lines, developing, implementing and optimizing production logistic concepts and solutions.
His e-mail address is:
fischmann@ipa.fraunhofer.de.

FABIAN BÖTTINGER is project manager and researcher at the department Ultraclean Technology and Micromanufacturing of the Fraunhofer Institute for Manufacturing Engineering and Automation, Stuttgart, Germany. He is mainly involved in the area of production logistics and production IT. He has been contributing to a number of international projects in semiconductor as well as photovoltaic industry. In doing so, he successfully planed new production lines and made use of the tools of simulation for developing, implementing and optimizing production logistic concepts and solutions.
His e-mail address is:
boettinger@ipa.fraunhofer.de.

ROLAND WERTZ is project manager and researcher at the department Ultraclean Technology and Micromanufacturing of the Fraunhofer Institute for Manufacturing Engineering and Automation, Stuttgart, Germany. His field of work is the area of production concepts, particularly production logistics. He is involved in international projects in semiconductor and photovoltaic industry as well as in publicly funded projects. Thereby he is using simulation to plan new productions and to optimize already existing ones.
His e-mail address is:
wertz@ipa.fraunhofer.de.

CHRISTIAN KUNZ is studying production technology at the University of Applied Sciences, Dresden, Germany and was writing his seminar paper at the department Ultraclean Technology and Micromanufacturing of the Fraunhofer Institute for Manufacturing Engineering and Automation, Stuttgart, Germany. His task was to analyze the behavior of a simulated waferfab in regard to the location of buffers.

ACTIVITY-BASED TRAVELER AGENT BEHAVIOURAL MODEL FOR MIXED TRAFFIC FLOW

Ling Huang
School of Civil and Transportation
Engineering
South China University of Technology
510640, Guangzhou
E-mail: hling@scut.edu.cn

Jianping Wu
WSP, Senior Technical Director
Beijing University of Posts and Telecommunication, Professor
University of Southampton, Professor

E-mail: jianping.wu@wspgroup.com

KEYWORDS

Behavioural Model, Traveler Agent, Traffic Simulation, Mixed Traffic Flow

ABSTRACT

The modeling of traveler's daily travel behaviours in mixed traffic system is a complex problem, mainly due to the complex nature of travel behaviours and the situations in mixed traffic flow. This paper explores an Activity-Based Traveler Agent behavioural Model (A-TAM) for mixed traffic flow, combining the ideas of activity-based traffic demand model, the hierarchical structure in behaviours, agent approach and subjective utility optimisation method. In the case study, A-TAM has been applied to model cyclists' behaviour at unsignalised intersections under mixed traffic flow conditions, and the validation results of the cyclists' model which proved A-TAM promising.

INTRODUCTION

Recently traffic simulation technologies, especially microscopic traffic flow simulation models, have been widely used in many areas of urban traffic management and optimization. Yet most microscopic traffic simulation models are unable to realistically represent the urban traffic system with mixed traffic flows in developing countries such China, where bicycles, cars and pedestrians conflict frequently on urban roads and at intersections, especially at unsignalised ones. Due to the complex nature of travel behaviours and the situations in mixed traffic flow, the modeling of travelers daily travel behaviours in mixed traffic systems are a complex problem. However, new progress in computer science, traffic modeling and simulation have given inspiration to this problem, which are; agent approach in simulation, activity-based traffic demand model, microscopic behavioural models of different road users (driver, cyclist, pedestrian) and related behavioural studies in psychology and social science.

As Ralf Schleiffer (2002) points out that "intelligent agents ….their simple determinism is suitable to illustrate and to describe the dynamic structure of human behaviour with reference to traffic and transportation". Agent-based simulation appears to have the potential for modeling behaviours for different road users (driver, cyclist, pedestrian) in mixed traffic flow. Each road user could be modeled as an autonomous agent with its own knowledge and goals. For example, Dia (2002) gives a microscopic agent-based simulation of driver's route choice behaviour on real-time traffic information and Ronald and Sterling (2005) provides a BDI agent simulation model for pedestrians. Researchers also study the frame and structure for agent-based microscopic traffic flow simulation models (Sun, 2002; Yu, 2006). Hoogendoorn and Bovy (2004) proposed a route-choice and activity scheduling theory and model for pedestrians using hierarchical model framework and subjective utility maximisation theory. Beuck et al.(2007) reports on their recent case study of a multi-agent traffic simulation for real-world Berlin scenario, where they suggest the differentiate between the physical layer and the mental layer of a multi-agent simulation.

The purpose of our research is to build a general traveler agent behavioural model in mixed traffic system, as a behavioural model basis for developing an urban mixed traffic flow simulation model, with large-scale and great details (e.g. describing the conflictions of bicycles and motorcars at intersections). Thus the model has to cover human daily travel behaviours of all levels, from trip planning to specific moving (e.g. turning to avoid collision). A-TAM, the theoretical model we proposed here seems to meet all these requirements.

The paper first outlines the three-level model framework by analyzing a typical traveler's daily travel behaviour, and follows the descriptions of agent structure and behavioural models on each level. A case study in Beijing is presented, and ends with a discussion and brief conclusions.

MODEL FRAMEWORK

Followed are a typical travelers daily travel behaviours:

When a traveler α plans to go out, they must have at least one **goal**. Accordingly at least one *activity i* is to take place at D_i with enough time T^i for achieving the

goal(s). We assume that after all the travelers final destination is home (D_H), so D_i (i=1, 2…) and D_H form a "trip chain". Traveler α chooses and orders the activities *i* according to there preferences, goals, and information on road networks and thus forms a Trip Plan Activity (TPA), which includes the planned activities to be taken (*i*), their expected locations (D_i) and expected finish time T_i （$T_i \leq T^i$）, the traffic mode to the activity location D_i (M_i), trip route R and departure time T_O.

Above are traveler's pre-trip behaviours (trip planning), followed are in-route behaviours, including two major types: path finding and moving.

When α is in-route at a specific transport facility Ω (a link or intersection) during the trip, we assume that the traveler would make a list of **Expected Temporal Activities** (**ETA**) according to the specific traffic situations when finding a path in Ω. The **ETA** includes the ETA *j*, temporal destination TD_j (where ETA *j* taking place) and the expected finish time T_j. As the time for path finding process is very limited (usually a few seconds), and the vagueness of the traffic condition information, we assumed that the planned path is a polygon made of a series of *key points* (i.e. TD_j) between the original (α actual location) and destination. We can speculate that α may have an idea on when they will reach the *TDs*, while the actual moving process is up to the subconscious, i.e. the traveler moves to the *TD* automatically.

When α moves along the planned path, by the energy saving principle (Kölbl and Helbing, 2003), α usually tends to move directly from current position to the temporal destination TD_i, if there are no influences. Yet, sometimes unexpected situations appear, which would affect α's moving behaviour (i.e. unexpected conflicting), in A-TAM this unexpected conflicting is regarded as a special type of activity, called *Unexpected Activity* (**UA**), α would take **Reaction** (RA) to avoid the confliction, and at the same time try to adhere to the planned path as much as possible.

From the analysis on a typical traveler α's behaviour, we can see that daily travel behaviours are comprehensive and consist of behaviours of different complexity. Psychologists suggest a classification of stimuli into simple or standard situations and complex or new situations (Helbing, 1995). Clearly, the trip planning and path finding behaviours are behaviours in complex or new situations, and the moving (including walking, driving, riding) and reactions to unexpected conflictions are of simple or standard ones. In their pedestrian behavioural model, Hoogendoorn and Bovy (2004) distinguished pedestrian's choices at three levels. Similarly, the traveler's daily travel behaviours are divided in three levels:

1) **Strategic level**: including decision making/adjustment of *Trip Planned Activity* (*TPA*, consist of destinations, route **R**, traffic modes and departure time) pre-trip or in-route;
2) **Tactical level**: including decision making/adjustment of *Expected Temporal Activities* (ETA) and thus the path **P** at facility Ω according to the traffic situation and **TPA**.
3) **Operational level**: including all moving (riding/driving/walking) behaviours (speed, accelerations), along the planned path **P** and Re-Actions (**RA**) to Unexpected Activity (**UA**).

In this hierarchy structure, expected utilities at lower levels influence choices at higher levels. Choices at higher levels condition choice sets at lower levels. In this way, A-TAM model cover nearly all daily travel behaviours (see **Fig.** 1): from decision making, on trip goal and activity, traffic mode, departure time and route choice, to path finding (or path choice) at a certain facility, and the concrete moving behaviours (moving speed, acceleration, turning). The agent attribute square covers the models of three levels, showing that the agent's attributes such as gender, age, preferences for traffic mode and routes, travel patterns, home location, goals, car ownership, income, perception of network would affect behavioural models of all levels. The subjective utility optimisation theory is used to model traveler's decision making (choice) behaviour, which has been applied in some travel behaviours.
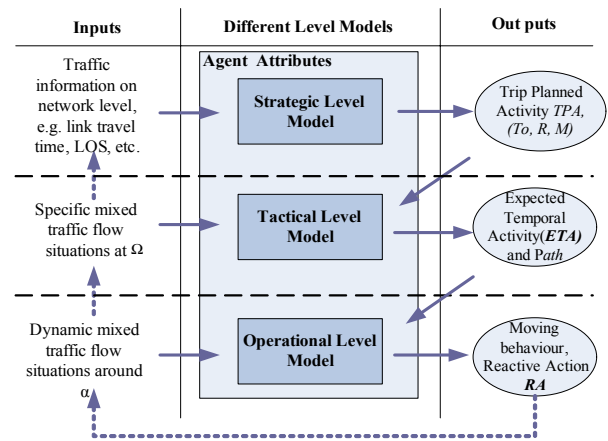

Fig. 1: Model Framework of A-TAM

By information feedback (dotted lines in **Fig**. 1), A-TAM model supports dynamic activity planning, route /path choice behaviours at strategic and tactical levels. Agent α could judge and update $ETA(t + \Delta T)$ and $Path(t + \Delta T)$ according to the changed traffic situations in Ω at an internal of ΔT. Similarly, α could judge and update there $TPA(t + \Delta T_s)$ according to the latest

network traffic information, such as travel time (T_Ω) and Level-of-Service (LOS_Ω) of the facility Ω.

MODEL DESCRIPTION

Intelligent Agent Architectures

The inspiration of the agent architectures proposed in this study stems from earlier research work by Dia (2002). The agents possess a mental state which is composed of various mental elements; beliefs, capabilities, commitments, and behavioural and commitment rules and a physical state which is composed of related attributes, dynamic situation and perceptions as shown in Fig. 2.

Beliefs: Belief is a fundamental part of the agent's mental model. For the purpose of traveler behavioural models, these beliefs will include information about the travelers travel patterns, preferences for routes, perceptions of the situations on network and around other travelers travel behaviours.
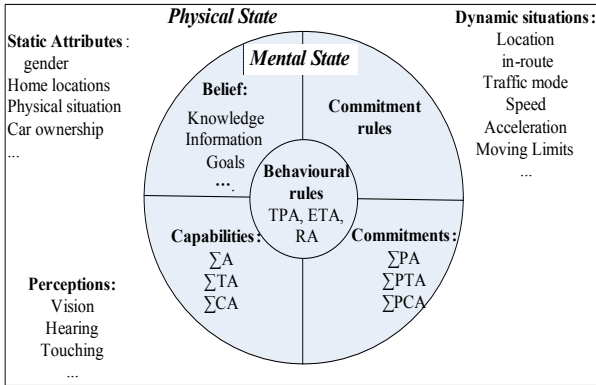
Fig.2: Intelligent Travel Agent Model

Capabilities: A capability is a construction used by the agent to associate an action with that actions necessary pre-conditions. For the purpose of traveler's behavioural models, capabilities represent the activity sets that the traveler can perform, presented as Activity set $\sum A$, Temporal Activity set $\sum TA$ and Conflicting Activity set $\sum PCA$ for each level model.

Commitments and commitment rules: A commitment is an agreement to attempt a particular activity at a particular time, if the necessary pre-conditions for that activity are satisfactory at that time. For the purpose of traveler behavioural models, commitments represent the plausible activity sets meeting certain conditions, presented as Plausible Activity set $\sum PA$, Plausible Temporal Activity set $\sum PTA$ and Potential Conflicting Activity set $\sum PCA$.

Behavioural rules: If the behavioural rules conditions are satisfied by the environment, then the rule is applicable and the action it specifies are performed. For the purpose of traveler behavioural models, actions represent the model outputs of different levels, presented as **TPA**(t) (Trip Planned Activity), **ETA**(t) (Expected Temporal Activity) and **RA**(t) (ReAction), which are all time dependent vectors.

Static attributes: Static attributes refer to the attributes such as gender, home location D_H (also O_H), physical situation, car ownership, etc. which will be unchanged in A-TAM, and the **Dynamic attributes** refer to the changeable variables such as location, in-route state, traffic mode, moving speed, acceleration, moving limits etc. Agents attributes (static and dynamic) would influence it's travel behaviours.

Perceptions: Perception is the information obtaining and filter function module, which determines what information of the environment reaches agents beliefs.

Strategic Level Model

Fig. 3 shows the model structure on the strategic level. The inputs of model on strategic levels include environment information as network situation. The activity choice set $\sum A$ is not an ordered set however, some activity could be order dependent, for example, if the happening activity i is the pre-condition of activity j, when i didn't happen, then j could not happen. Thus the activities sets meeting this order dependency form plausible activity set $\sum PA$. The model outputs are *ordered vector sets* Trip Planned Activity $TPA = \{(i, D_i, T_i, M_i)\}_{T_O, i \in \sum PA}$, $i=1,2\ldots k$, where k is the activity number of TPA, and route $R = \{\Omega_j\}$, $j=1$, $2\ldots m$, where m is the facility number in R, obviously $R = \{\Omega_j\}$ must form a continuous route on network. The agent applied subjective utility optimisation as its behavioural rules in this model.

Note that on a strategic level, position variables referred to traffic facilities, e.g. $x(T_{i-1})$ could be facility Ω_j.

And the current and near future location of α and the network traffic situations would affect **TPA** simultaneously. Meaning, the chosen route **R** and TPA are co-dependent, they will determine and affect each other, so we hypothesise, α makes a combined decision on **TPA** and **R** minimising the expected disutility or cost C:

$$C\left(\{(i, D_i, T_i, M_i)\}_{T_O, i \in \sum PA}, R\right)$$
$$= \sum_{i \in \sum PA} C_i(D_i, T_i, M_i, R, x(T_{i-1}))$$
$$\text{s.t. } \vec{x}(T_O) = O, \ T_i \leq T^i \qquad (1)$$

Where $\vec{x}(T_O) = O$ is the agent departure time boundary condition, and $T_i \leq T^i$ is the time condition of activity i; $C_i(D_i, T_i, M_i, R, x(T_{i-1}))$ stands for the combination of travel cost from $x(T_{i-1})$ the place where the activity finished (i-1) to D_i and $\phi_i(T_i)$ the utility of activity i:

$$C_i(D_i, T_i, M_i, R, x(T_{i-1}))$$
$$= \sum_{\Omega \in R_i} [T_\Omega(M_i, T_{i-1}) + F_\Omega(M_i, T_{i-1})] + \phi_i(T_i) \quad (2)$$

Where $T_\Omega(M_i, T_{i-1})$ and $F_\Omega(M_i, T_{i-1})$ are the travel time and fare at Ω of route R_i (the route from D_{i-1} to D_i) at T_{i-1} via traffic mode M_i).
Optimising subjective utility/cost function C, the model output is：

$$\{\{i\}^*, \{T_i\}_{i \in \sum PA}^*, \{M_i\}_{i \in \sum PA}^*, R^*, T_O\}$$
$$= \arg \min C (\{(i, D_i, T_i, M_i)\}_{T_O, i \in \sum PA}, R)$$
$$(3)$$
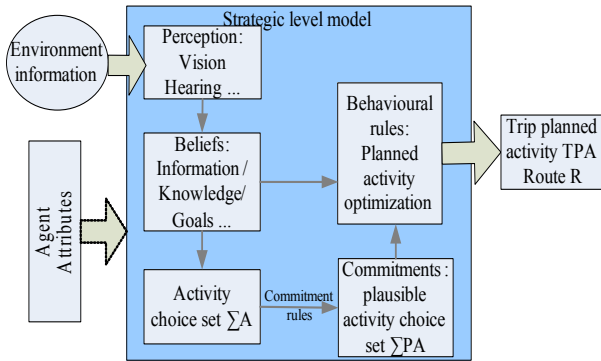
Fig. 3 Model on Strategic Level

Tactical Level Model

Fig. 4 shows the model structure on the tactical level. On this level the path finding behaviour is also activity based, except for the activity type is temporal activity, similar to the strategic level model, the temporal activity choice set is $\sum$**TA** and the plausible temporal activity choice set $\sum$**PTA**. The model inputs are the outputs of strategic level model and the dynamic traffic situations in Ω. The model outputs include ordered vector set. Expected Temporal Activity $ETA = \{(j, TD_j, T_j)\}_{j \in \sum PTA}$, $j = 1, 2 \ldots n$, where n is the activity number in **ETA**, and path $P = \{(TD_j, T_j)\}$ stands for the ordered pairs of the location and time of j.

Note that possible influence factors of **ETA** include the current and near future location of α and the dynamic specific mixed traffic situations at Ω. That means, the planned path **P** and **ETA** are co-dependent, they will determine and affect each other, so we hypothesise α makes a simultaneous decision on **ETA** and **P** minimising the expected disutility or cost **C**. On this level, positioned variables are two dimensional vectors with an accuracy of about 1 meter in the special coordination of Ω.
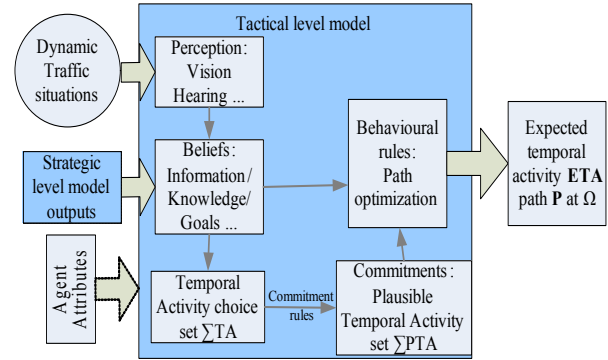
Fig. 4 Model on Tactical Level

The combined subjective disutility/cost **C** of **ETA** and path **P** is:

$$C (\{(j, TD_j, T_j)\}_{j \in \sum PTA}, P)$$
$$= \sum_{j \in \sum PTA} C_j(TD_j, T_j, \vec{x}(T_{j-1}))$$
s.t. $\vec{x}(t_o) = \vec{x}_0$, $\vec{x}(\tau) \in \Omega | t_0 \leq \tau \leq t_1 \quad (4)$

Where the first condition stands for the initial location of agent α; the second condition is the agents moving scope. C_j stands for the sum of travel cost moving from $\vec{x}(T_{j-1})$ to TD_j and the benefit of j.

Operational Level Model

The operational level model in A-TAM describes traveler's reaction level behaviours, such as the car following behaviour when driving a car which are usually rather automatic and taking place in a very short time (a few seconds or less). The *Social Force model* (Helbing, 1995, 1997) which stems from the famous *Social Field Theory* (Lewin, 1951) in psychology is adopted to describe the agent's operational level behaviour, moving (driving/walking/riding) alone, the planned path and conflicting avoidance behaviours. The *Social Field Theory* proposes that human behaviour is the function of both the person and the environment, expressed in symbolic terms as:

$$Behaviour = f (Person, Environments) \quad (5)$$

This means that the individual's behaviour is related both to the individual's personal characteristics and the social situation in which the individual finds themselves in. The Social Force model (SF model) has been applied in the pedestrian dynamic behaviours and gained good results in the last decade (Helbing, 1995, 1997; Lakoba et al., 2005).

The A-TAM social force model includes three function modules (see Fig. 5): **Intended motion**; describes agent's intention of trying to get to the nearest temporal destination TD_j at T_j as planned in the tactical level model. **Effective field;** with the traffic situation nearby perceived by **Perception;** module as effective influence source (exerting tensions) and, **Balance** module; synthesising and balancing all the tensions "social force" from the TD_j and effective objects from the above two modules puts out the reaction RA.
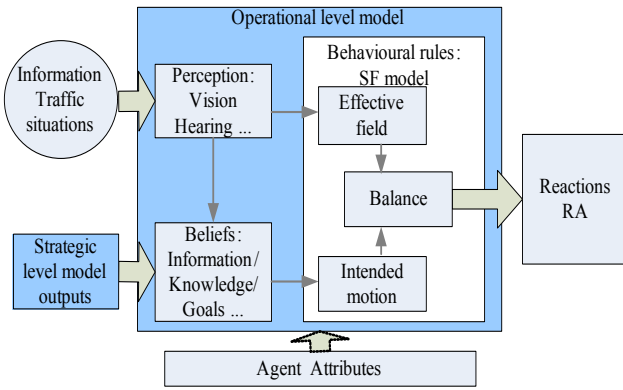
Fig. 5 Model on Operational Level

The SF model for the behaviour of individuals under the influence of a social field shows some analogies with the physical model for the behaviour of electrons in an electric field (e.g. of an atomic nucleus), and the SF model is generally formulated as:

$$\frac{d\vec{x}}{dt} \approx \vec{F}_\alpha(\vec{x},t) + \xi \qquad (6)$$

$$\text{with } \vec{F}_\alpha(\vec{x},t) = -\nabla \vec{w}_\alpha(\vec{x},t) \qquad (7)$$

Where $\frac{d\vec{x}}{dt}$ denotes the change rate of individual behaviour $\vec{x}$, and the fluctuation term ξ describes individual behavioural variations, the vectorial Quantity $\vec{F}_a(\vec{x},t)$, drives the systematic change of the behaviour $\vec{x}(t)$ of individuals α. $\vec{F}_a(\vec{x},t)$ is denoted as social force acting on α. The potential $\vec{w}_\alpha(\vec{x},t)$ can be understood as *social field*，and the social force $\vec{F}_a(\vec{x},t)$ is given by its derivative (by its gradient ∇).

Clearly, the force $\vec{F}_a(\vec{x},t)$ must represent the effects of the environment on the behaviour of α. However, the social force is not exerted by the environment on the traveler's physical body. It is rather a quantity that describes the concrete *motivation* to act. In the case of the cyclist riding or drivers driving or pedestrians walking behaviour, this motivation evokes the physical production of an acceleration or deceleration force as a reaction to the perceived information that the traveler obtains about there environment.

CASE STUDY

Huang and Wu (2007) studied the cyclists microscopic travel behaviour at unsignalised intersections in mixed traffic flow in Beijing, by applying parts of the A-TAM model framework (the tactical and operational level models). In this case, the cyclists Temporal Activity **TA** is defined as "avoid conflicting objects" (motorcars, bicycles, pedestrian or obstacles). Clearly, the cyclist planned path **P** would influence the **TA**. In order to construct a reliable subjective utility function for the tactical level model, stated preference (SP) data where collected in Beijing, interviewing approximately 1000 Beijing cyclists to obtain influence factors and their preferences in path finding behaviours at unsignalised intersections. Video data was collected at two typical unsignalised intersections (cross and three leg), and an image capture software *VSpeed* (Bai, 2005) calibrated via Differential Global Position System (DGPS), data was used to capture dynamic motion data of all moving objects (motorcars, bicycles, pedestrians) at the intersections see (Fig 6).
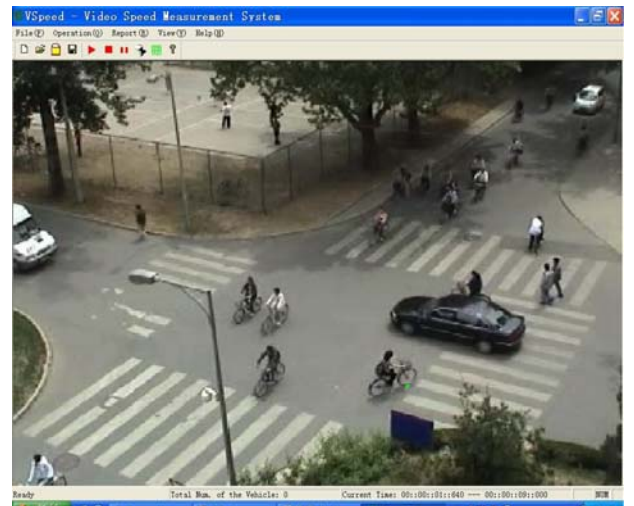

Fig. 6: Dynamic Data Capture

Data collected from over 100 conflicting cases were used in the model validation, including simple circumstances (with one conflicting motorcar/pedestrian/bicycle) and complex situations (with two or more conflicting

motorcars/pedestrians/bicycles). Validated items are bicycle (agent) accelerations, speeds and positions. The general model validation results are presented in Fig. 7, proving the proposed model is promising in describing the cyclists crossing behaviour at unsignalised intersections. Further details in one of the case studies refers to the paper of L. Huang and J.Wu (2007) listed below.
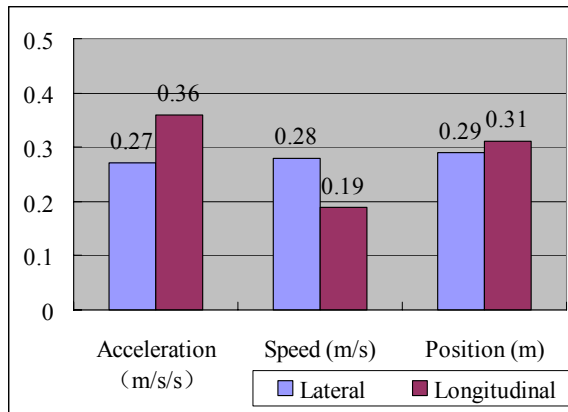

Fig. 7: Average Difference between the Model Outputs and Empirical Data

CONCLUSIONS

In this paper, we propose an Activity based Traveler Agent behaviour Model for mixed traffic flow, A-TAM. The hierarchical model framework and activity-based travel behaviour modeling ensure A-TAM could reasonably describe most daily travel behaviours, from trip planning to a specific moving behaviour. The application of agent approach and subjective utility method also enhance its modeling potential. The case study in Beijing of cyclist's behaviour simulation in mixed traffic flow at unsignalised intersections proved the A-TAM promising in describing cyclist's behaviours in mixed traffic flow.

A-TAM proposed here is an attempt on general travel behavioural model for urban mixed traffic flow simulation, yet there seems much work to do in specifying the model of all levels and the model framework itself.

ACKNOWLEDGMENT

REFERENCES

H. Bai and J. Wu "A Flexible Camera Calibration for Freeway Scenes:" *Proceedings of the ICNC'05-FSKD'05*, 2005.

D. Helbing and P. Molnár "Social Force Model For Pedestrian Dynamics". *Phys Rev E* 1995; 51: 4282-4286.

D. Helbing, J. Keltsch, and P. Molnár (1997) "Modelling The Evolution Of Human Trail Systems". *Nature* 388, 47-50.

H. Dia "An Agent-Based Approach To Modeling Driver Route Choice Behaviours Under The Influence Of Real-Time Information". *Transportation Research Part C*, 2002 ,10 (5) :331-334.

S.P. Hoogendoorn and P.H.L. Bovy. "Pedestrian Route-Choice And Activity Scheduling Theory And Models." *Transp. Res., Part B*, February 2004, pp. 169-190(22).

K. Lewin, *Field Theory in Social Science*, Harper, New York, 1951.

L. Huang and J.Wu "Microscopic Mixed Traffic Flow Simulation Model At Unsignalized Intersections" *ITS WORLD Congress*. Oct. 2007. Beijing.

N. Ronald and L. Sterling, "A BDI Approach To Agent-Based Modelling Of Pedestrians", *19th European Conference On Modelling and Simulation,* June, 2005 Riga, Latvia.

R. Kölbl and D. Helbing (2003) "Energy laws in human travel behaviour". *New Journal of Physics* 5, 48.1-48.12.

Schleiffer Ralf. "Intelligent agents in traffic and transportation".*Transportation Research Part C* (S0968-090X), 2002, 10: 325-329.

J.W. Sun, M.S. Li and Z.M. Er "The Agent-based Architecture and Simulation of Intelligent Traffic", *Computer Sinulation* Vol.19.No.3,May 2002pp46-49

T.I. Lakoba, D.J. Kaup and N. M. Finkelstein, "Modifications of the Helbing-Molnár-Farkas-Vicsek Social Force Model for Pedestrian Evolution", *Simulation*, Vol. 81, No. 5, 339-352 (2005)

U. Beuck, K. Nagel, M. Rieser, D. Strippgen and M. Balmer. "Preliminary results of a multi-agent traffic simulation for Berlin". *The Dynamics of Complex Urban Systems, An Interdisciplinary Approach*, 2007.Publisher: Physica-Verlag Heidelberg. pp75-94.

Q. Yu, J.J. Shi, J. Rong and F.T. Ren, "Frame of Micro Traffic Flow Simulation System Based on Agent", *Computer Simulation*, 2006. Vol.23, pp210-214

LING HUANG received her PhD degree in System Engineering from the Beijing Jiaotong University in 2007. She is currently a faculty in the School of Civil and Transportation Engineering in South China University of Technology (SCUT), her research interests include microscopic traffic flow simulation, travel behaviour modeling. Her email address is hling@scut.edu.cn.

J. WU is a Senior Technical Director for WSP and Professor and Director of UK-China ITS Centre (UCIC) in University of Southampton, and Chon Kong Scholar Professor in Beijing University of Posts and Telecommunications (BUPT). His research interests include fuzzy modeling, traffic flow simulation, travel behaviour study, data fusion. His email address is jianping.wu@wspgroup.com .

INVESTIGATION OF THE INFLUENCES OF ARTICLE AND ORDER STRUCTURE ON THE DIMENSIONING OF ZONE-PICKING-SYSTEMS

Dipl.-Inf. (Univ.) Alexander Ulbrich
Dipl.-Wirtsch.-Ing. (Univ.) Stefan Galka
Prof. Dr.-Ing. Dipl.-Wirtsch.-Ing. Willibald A. Günthner
Institute for Materials Handling, Material Flow and Logistics
Technische Universität München
Boltzmannstr. 15, 85748 Garching, Germany
E-mail: ulbrich@fml.mw.tum.de / galka@fml.mw.tum.de

KEYWORDS

order-picking systems, rough planning, simulation modules, zone-picking, dimensioning, decision making, influences on picking systems, logistics, material flow

ABSTRACT

"First, the order picking then the stock planning – In the order picking area the most stuff is usually employed. Here the customer service and the logistics quality are decided. The highest costs incur here." [7]. The planning of an order picking system is characterized by the complexity of the system. Through the use of simulation in rough planning, more precise statements about the performance of order picking systems can be made. The planning process with the PlanKom tool, which uses the simulation, is described at the beginning of this paper. Focus is the investigation from the so-called zone-picking. The planning of such a system raises the question of how long a zone should be, so that performance is maximized. In this context, the influence of article structure, storage strategy, order structure and number of zones on the performance is examined. For this study 1.080 simulation runs have been performed. The results show that the article structure has a large impact on performance by picking in a small number of zones. With increasing number of zones, the influence of the article structure is less. Furthermore, the results show that the storage strategy "concentration of fast moving parts" has a higher performance than a chaotic storage strategy.

PLANNING AND OPTIMIZATION OF ORDER PICKING SYSTEMS

The order picking is the central function of the warehouse logistics and has significant influence on areas like production and distribution [8]. Despite the trend towards automation, order picking is a costly area in modern logistics systems [12]. This is mainly due to the high personnel section [11]. In order picking systems from a total quantity of parts (the assortment) subsets are assorted by a customer order and then sent to the customer [11, 14]. These are the most difficult tasks of the in-house (intra) logistics [7]. This is due to the complexity of order picking systems, because there is a multitude of ways to realize the picking task [11, 4, 10].

Due to the requirements of the order picking, the most efficient mix of the spectrum of order picking technologies has to be selected. It is important to achieve a high delivery quality and simultaneously a high economic efficiency [8]. Often the right solution does not consist only of one specific order picking technology, but from multiple different order picking technologies, which can be arranged in a useful combination [1, 4, 10]. Such hybrid or heterogeneous order picking systems allow to adapt the complete system to the specific requirements.

The requirements for an order picking system become higher and higher. The requirements increase due to smaller individual orders, followed by a higher delivery frequency [9, 4]. Furthermore, the growing diversity of parts and the high standards of material availability advance the requirements. The delivery times are shorter and shorter and thus the cycle times too [1, 4, 8, 6]. Also additional services, such as the labelling of parts for customers, increase the requirement for order picking systems [9].

The planning process is complex due to the very different requirements, so it can be advantageous to use planning software. One possibility is the use of simulation. Within the 21st ECMS we have explained the advantages of simulation in rough planning and the necessity of standardisation [13]. Figure 1 illustrates the planning process. The planning tool supports the planner in phase of preparation and rough planning. Simulation can be used for investigation of the performance of order picking systems.
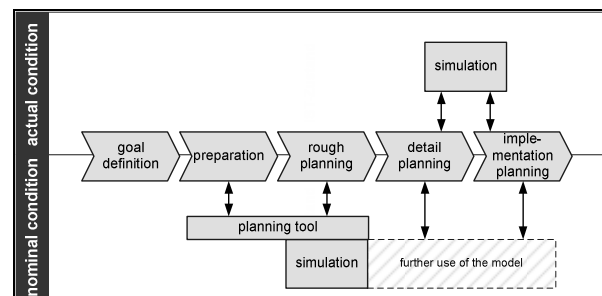

Figure 1: Usage of simulation environments in planning processes

The planning tool called PlanKom can be used for different kinds of investigation. The main field of

application is the investigation of system versions. But systematic investigations for generation of shaping notes can also be performed.

RESEARCH OBJECTIVE

In this study the so-called zone picking is investigated. Thereby the parts are provided statically in e.g. flow racks. The entire rack front is subdivided in sections (so-called zones). Figure 2 shows a zone picking system with two zones. Every zone has a basis. At the basis the order bins are available for the picker.
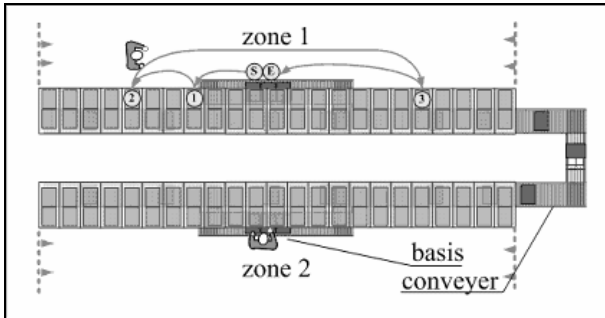

Figure 2: Real layout from a zone picking system with two zones

For the full development of an order, the bins must be filled with parts in one or more zones. Therefore the bin is transported in the zones by conveyor technique. In a zone a picker (human) receives the bin at the basis. For a better understanding the way a picker has to walk for an order in a zone is described next. In our case the picker takes the bin along (compare "S" in figure 2). A picking list is in the bin. The picker has to pick three orderlines (3 parts). He walks to the first part (1) and takes it. He puts the part in the bin and confirms the taking at the picking list. After this, he walks to the second (2) and 3rd (3) part and takes them. Following that, he walks back to the basis and places the bin on the conveyer (E). The basis is in the middle of a zone. Parts can be stored left or right from the basis. The way is determined by the longest distance from the basis to the storage place of a part in the left and in the right side.

In the planning process the length of a zone and therefore the number of zones must be defined. The length of a zone affects the performance of the picker. One planning task is to find the optimal length of zone, which provides the best performance. Further influencing variables have to be considered, because they also affect the performance. These are predominantly:

- the article structure (distribution of access frequency),
- the order structure (number of order lines) and
- the storage strategy.

The last one can be defined in the planning process, but the others cannot be affected by the planner.

The aim of study is to find out the impact of the various influencing variables on the performance. Based on the

simulation results shaping notes for zone picking systems are derived.

MODELLING THE ZONE-PICKING SYSTEM

For this investigation one of our standard modules (Zone-Picking-Module) is used and allocated with different parameters focuses the number of zones. Each different layout has to be modelled as a different model. To show the differences between the lengths of zones, the number of zones is duplicated from version to version beginning with 1 and ending with 32. In figure 3 an example of the module allocated with four zones is shown.
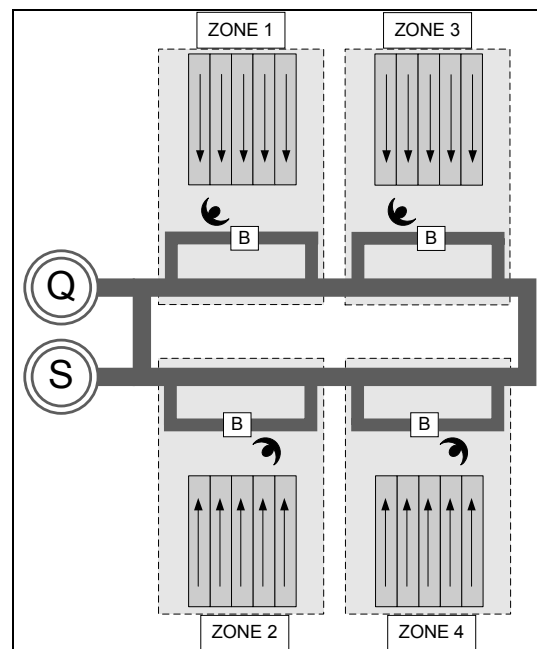

Figure 3: Example of the module with four zones

The planner is supported during the whole modelling process by PlanKom-Software and can allocate parameters independent from his simulation expertise. Figure 4 shows the modelling area, implemented as Drag & Drop GUI (Graphical User Interface), in the PlanKom-Software.
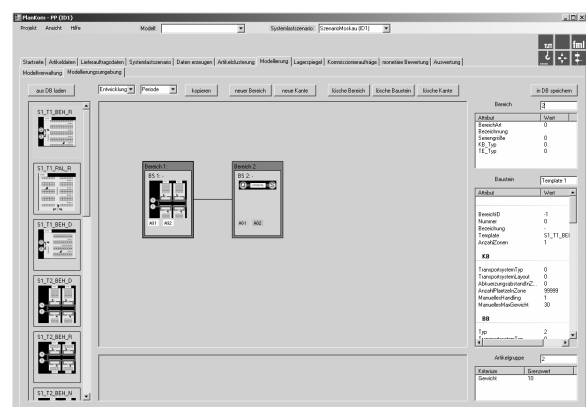

Figure 4: Modelling environment of PlanKom

INPUT DATA FOR SIMULATION

The input for every simulation run contains the picking orders, the storage place for every part (bin location) and the allocation of employees to the zones. Picking orders depend on the customer orders and the model and version respectively can be generated by PlanKom. The allocation of parts and storage places due to the storage strategy can also be created by PlanKom. The allocation of employees to the zones has to be done by the planner in the modelling environment of PlanKom for every version, which should be investigated. For the handling process the times required for simulation have to be defined by MTM (methods time measurement).

Thus, the influence of input data on the systems' performance can be investigated by gradually changing the article structure and order structure. In addition to these changes the number of zones and the storage strategy (chaotic storage and concentration of fast-moving parts (abbreviated cfm) is varied.
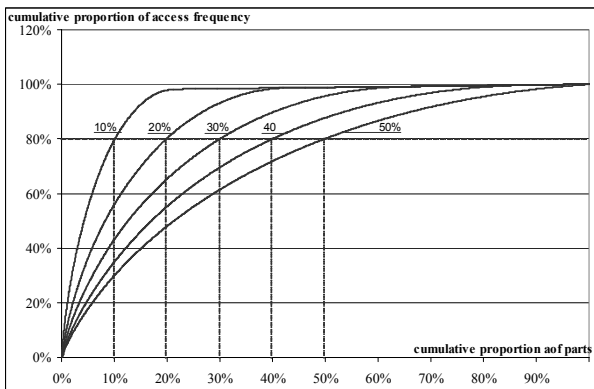
Figure 5: Different article structures - access frequency described by an exponential distribution

Figure 5 shows five different article structures. The article structure is defined by the percentage of articles ordered from the highest to the smallest access frequency, which lead to 80% of the total access frequency. The article structure can be described by an exponential distribution with the parameter lambda (λ). The value of Lambda is chosen in such a way that 10%, 20%, 30%, 40% or 50% of the parts lead to 80% of the total access frequency.

The customer order structure is raised by the variation of orderlines per order. In total 360 combinations of influencing variables are investigated. Figure 6 shows the combinations of influencing variables.

Figure 6: Combinations of influencing variables for analysis

For a high confidence level three simulation runs have been executed for every combination of influencing variables. From this it follows that 1.080 simulation runs have been performed for this study.

The picking and basis time were calculated by MTM. Therefore the process has been described in detail. Each process step is assigned a time. The sum of all times is the total time [2, 3]. Table 1 lists the times for picking an orderline (ol) and the basis time per bin and order (o) respectively. The basis time can be split into the process time for the receiving of a bin and the delivery of a bin after the picking process. In addition, the speed for the picker and the conveyor technique are listed.

Table 1: Process times and velocities

	unit	value
basis process (receiving)	[sec/o]	14,05
basis process (delivery)	[sec/o]	12,2
picking process	[sec/ol]	15,2
speed of pickers	[sec/m]	0,9
speed of conveyors	[sec/m]	0,5

WORKFLOW OF SIMULATION

All data needed for simulation is previously saved by PlanKom in a database. The Workflow of the simulation is oriented on figure 7.
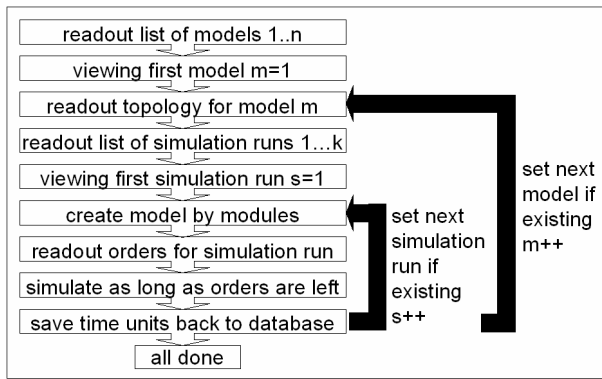
Figure 7: Workflow of simulation

First of all the list of models is read from database. Parameters are set to the first model and the topology and the simulation runs which have to be executed for this model are read. After this the model is created by the simulation automatically for the first simulation run by the information of topology and known modules. Also the order information is loaded and the simulation run is executed. After saving the time units to the database, the next simulation run for the actual model can be loaded and executed or if no more exist, the next model can be loaded and execute the simulation runs accordingly.

RESULTS AND INTERPRETATION OF SIMULATION OUTPUT

In this study the picking time and performance of zone picking systems are focused on especially. After that the costs per pick are shown in an example.

The time for receiving and delivery of an order (bin) in a zone will be called as basis time.

Generally the basis time is depending on the number of zones in which picking for an order has to be done. This time unit is always equal if there are only orders with one orderline because only one zone has to be reached to fulfill the order. If there is a great number of orderlines which have to be picked in a single zone, much time can be saved.

By increasing the number of zones, the basis time per orderline approaches the value of one orderline per order, because on average there will only be picked one orderline per zone. Thus, for almost every orderline of an order, a zone has to be reached.
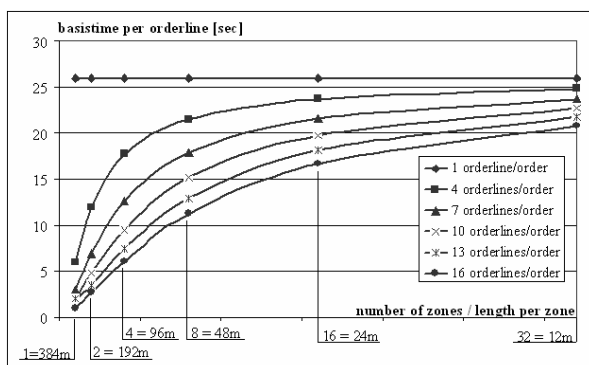
Figure 8: Basis time per orderline

By determining the bin location for every article by their access frequency, so that high-frequency articles are located near the basis, there is an influence on the walking time.

There is a big variation of the walking time in the values between the values with the same number of orderlines and the same number of zones as long as the number of zones is small. With a greater number of zones the values are almost equal between the different strategies and article structures.

By using an equal distribution of articles in the zone, the walking time is independent from the article structure.
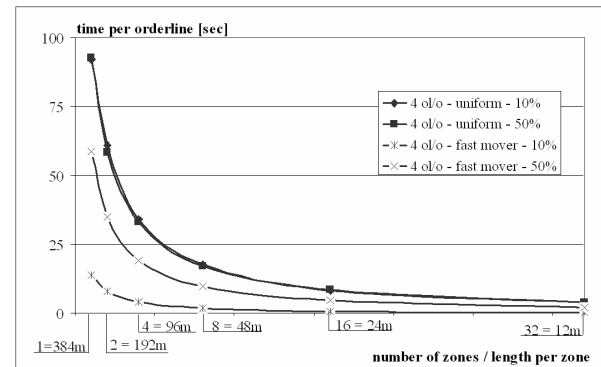

Figure 9: Influence of the article structure in combination with the storage strategy on the walking time

The total time per orderline is the sum of basis, walking and picking time.

There is no influence of the length of zones, article- or order structure on picking time. This will be the same time value for every zone. Because of that there will be no further view on the picking time.

The curve progression in figures 10-15 is caused by the basis and walking time.

Basis time increases by increasing the number of zones and decreasing length of zone respectively (except for one orderline per order as mentioned previously).

By decreasing length of zone, walking time per position decreases accordingly.

If there are only orders with single orderlines it seems to be optimal to choose a very short length of zone. At the same time every additional zone is associated with additional investments. In addition to that, the work load of the pickers will not be ensured any more, so they have to change zones from time to time. This decreases the efficiency.

If the order structure mainly contains orders with a great number of orderlines and at the same time there is a strong ABC distribution in the article structure there is a great savings potential for the total time per orderline (compare figure 5). This combination leads to an optimal length of zone which is relatively short.

If there are average 10 orderlines per order and an ABC distribution where 20% of articles lead to 80% of sales the optimal value of the length of a zone is about 30-50% of the complete required length.
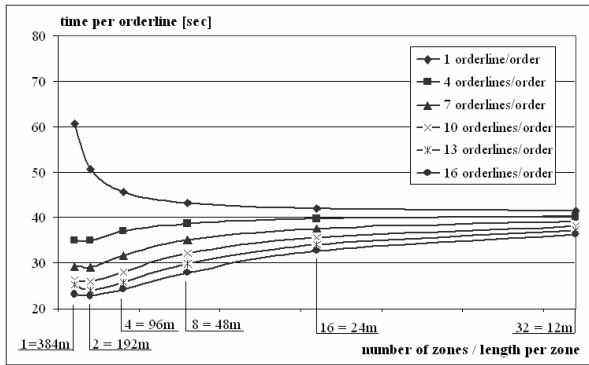
Figure 10: Picking time per orderline for the storage strategy "cfm" and the article structure 10% depending on the number of zones and the orderlines per order
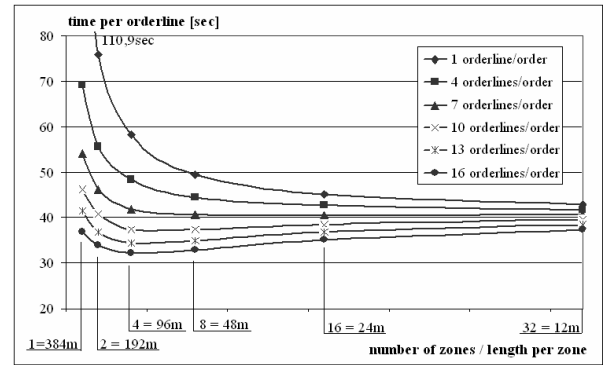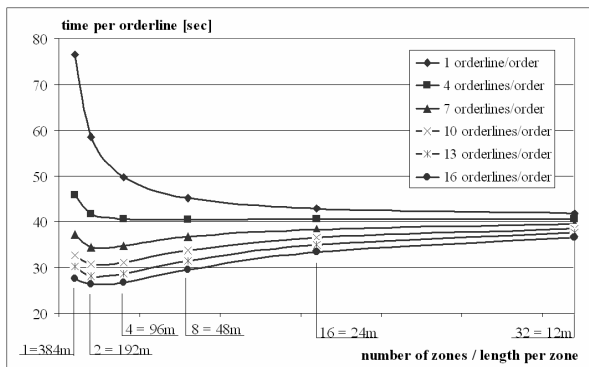

Figure 11: Picking time per orderline for the storage strategy "cfm" and the article structure 20% depending on the number of zones and the orderlines per order
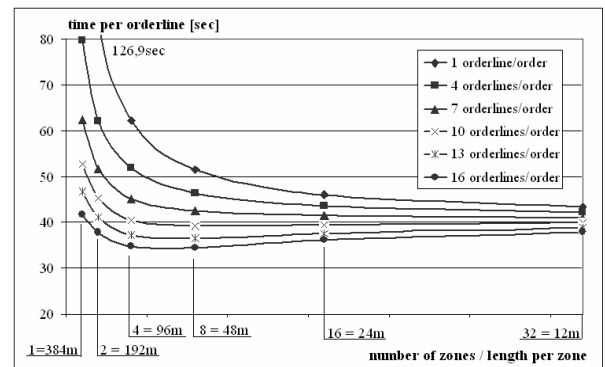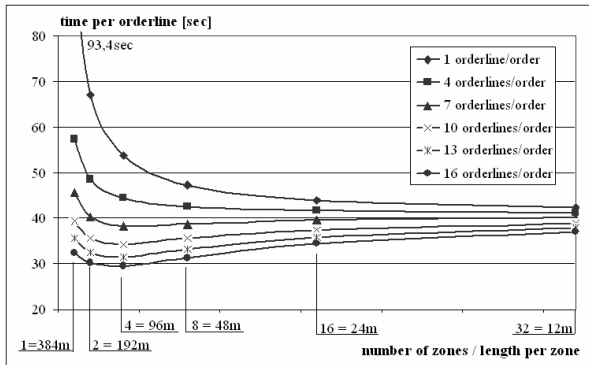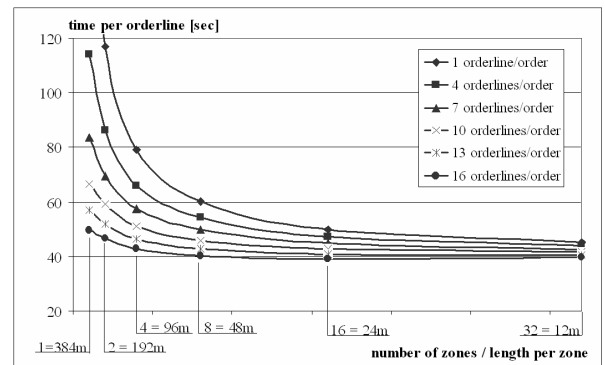

Figure 12: Picking time per orderline for the storage strategy "cfm" and the article structure 30% depending on the number of zones and the orderlines per order

Figure 13: Picking time per orderline for the storage strategy "cfm" and the article structure 40% depending on the number of zones and the orderlines per order
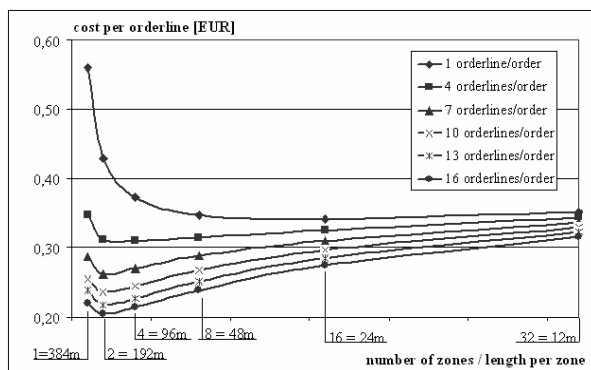

Figure 14: Picking time per orderline for the storage strategy "cfm" and the article structure 50% depending on the number of zones and the orderlines per order

By using a uniform distribution for the allocation of articles to bin locations, the total times of picking for all cases of study were inferior to using an ABC distribution. In the present basic conditions a uniform distribution for reposition strategy should not be chosen. If there is a necessity to use a uniform distribution, the optimal zone length could be determined only if investments and costs are considered, because the results of simulation show that a very short length of a zone is beneficial.

Figure 15: Picking time per orderline for the storage strategy "chaotic" and the article structure 20% depend from the number of zones and the orderliness per order

For determining the expense-optimum of zone length, the investments have to be considered. Based on the investments the accrual, imputed interests, maintenance and running costs could be calculated. The determined times are the basis for calculation of the personnel costs. The costs progression shown in figure 15 depends on the applied investments, personnel cost and the effort per year (business days/orderlines per day). These factors are project-specific. Because of that there is no conclusion to generality feasible. In the example shown in figure 16 the number of zones set to two (length of 192 metres) is cost-optimal in case, that the number of orderlines is greater than four. In case, there is only one orderline to pick for each order, it is advisable to set the number of zones to 16.

Figure 16: Costs per orderline in EUR for the storage strategy "cfm" and the article structure 20% to 80% depending on the number of zones and the orderlines per order

SHAPING NOTES FOR ZONE-PICKING-SYSTEMS

The storage strategy has an influence on performance. The results show that the concentration of fast moving parts is always better than the chaotic allocation of storage places. It is optimal to use the fast mover concentration. If it cannot be used, the article structure can be disregarded, because the results show that the article structure has no influence on the performance by chaotic allocation of storage places.

In contrast, the article structure influences the performance by using the concentration of fast moving parts. If the ABC distribution is great sharply increasing it is beneficial to dimension the picking system into a small number of zones (except for orders with one or two orderlines). This effect is enhanced by increasing the number of orderlines per order. It is beneficial to split the system in two zones if there is an order structure from more than 10 orderlines per order and a sharp increase of the ABC distribution (up to 20%). If the ABC distribution is even, it is better to split the system in three to six zones. For order structures with less orderlines per order it is better to have more zones. The investment must be considered for the decision, dimensioning the picking system in a great number of zones. The improvement of performance by an additional zone is small. This is shown into the flat rise

of the curve into the range from 16 zones and more. Figure 15 shows that an additional zone can drive up the costs, though the time per pick is smaller (compare figure 10). In this case a universal statement cannot be taken because the investment and personnel costs are project-specific parameters.

Through an analysis of actual data the order mix (proportion of number of orderlines per order) can be determined. This proportion can be used customizing the results. So it is possible to convert the simulation results for project-specific parameters. Figure 17 shows five examples for the customization of these simulation results.
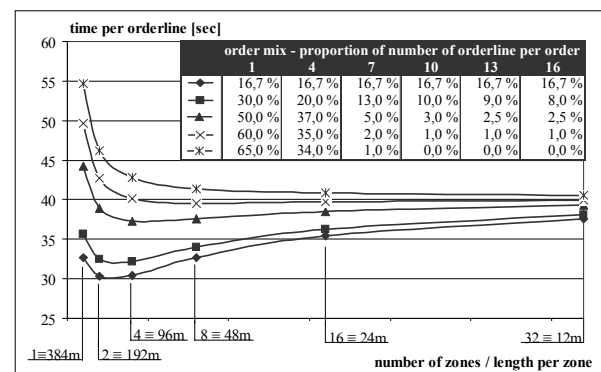

Figure 17: Time per orderline

OUTLOOK

Use of simulation in rough planning brings many advantages. For example:

- planning is based on a substantially more exact/broader database
- interactions between the individual ranges of an order-picking system are considered
- detailed investigation of several versions [13]

The standardized planning process with PlanKom software helps the planner structuring the planning task. The planner does not need extensive knowledge in using simulation software, because the simulation model can be designed by drag and drop with PlanKom software. The interface between PlanKom and the simulation is a relational database with 49 tables. The database is the core of the planning process with PlanKom. The data structure in the database allows the usage for all kinds of order picking systems.

At the moment PlanKom is being extended with functions for analytical methods. In future it is possible to choose between both - simulation or analytical methods. As part of the enlargement of PlanKom the results of simulation and analytical methods will be compared [5].

REFERENCES

1. Bode, Wolfgang: Welches System für welche Aufgabe. F+H 7-8 (2006), Vereinigte Fachverlage, Mainz
2. Deutsche MTM-Vereinigung e.V.: Schulungsunterlagen MTM-UAS (2007), Hamburg
3. Deutsche MTM-Vereinigung e.V.: Handbuch MTM-Logistik (2007), Hamburg
4. Dullinger, Karl-Heinz: Das richtige Kommissionier-Konzept: eine klare Notwendigkeit. Jahrbuch der Logistik (2005)
5. Galka, Stefan; Ulbrich, Alexander; Günthner, Willibald: Performance Calculation For Order Picking Systems By Analytical Methods And Simulation. 3. Deutsch-Russischer Logistik-Workshop (2008), Moskau
6. Goldscheid, Christian; Mackowiak, Jürgen: Ganzheitliche Betrachtung manueller Kommissioniersysteme. keit. Jahrbuch der Logistik (2006)
7. Gudehus, Timm: Logistik – Grundlagen, Strategien, Anwendungen. Springer-Verlag, Berlin Heidelberg, (2004)
8. Langen, Karl-Heinz: Strategien für die Kommissionierung mit Regalbediengeräten, Karussellägern und Sortern Jahrbuch der Logistik (2001)
9. Lüning, Ralf: Die Automatisierungsschere der Kommissionierung. TU-Ilmenau, Institut für rechnergestützte Produktion (2006)
10. Pieper-Musiol, R: Entscheidungshilfe zur Auswahl und Bewertung von Kommissioniersystemen. F+H "Marktbild Lager", Vereinigte Fachverlage (1982), Mainz
11. Sadowsky, Volker: Beitrag zur analytischen Leistungsermittlung von Kommissioniersystemen, Dissertation Universität Dortmund (2007)
12. ten Hompel, Michael; Schmidt, Thorsten: Warehouse Management. Springer-Verlag, Berlin Heidelberg New York London Paris Tokio (2003)
13. Ulbrich, Alexander; Galka, Stefan, Günthner; Willibald: Simulation Of Multi-Level Order-Picking Systems Within Rough Planning For Decision Making. 21st European Conference on Modelling and Simulation (2007), Prag
14. Verein Deutscher Ingenieure: VDI 3590 Blatt 1 – Kommissioniersysteme – Grundlagen. Beuth-Verlag (1994), Berlin

AUTHOR BIOGRAPHIES

Dipl.-Inf. Alexander Ulbrich was born in Munich, Germany and attended the University of Dortmund, where he studied applied computer science with mechanical engineering and obtained his degree in 2005. Now he works as a researcher at the Institute for Materials Handling, Material Flow and Logistics, Technische Universität München to obtain his conferral of a doctorate in engineering. His e-mail address is: ulbrich@fml.mw.tum.de and his web-page can be found at http://www.fml.mw.tum.de.

Dipl.-Wirtsch.-Ing. Stefan Galka was born in Magdeburg, Germany and attended the University of Magdeburg, where he studied business mechanical engineering and obtained his degree in 2005. From 2005 to 2006 he worked as a logistics planner for the DaimlerChrysler AG. Now he works as a researcher at the Institute for Materials Handling, Material Flow and Logistics, Technische Universität München to obtain his conferral of a doctorate in engineering. His e-mail address is: galka@fml.mw.tum.de and his web-page can be found at http://www.fml.mw.tum.de.

Prof. Dr.-Ing. Dipl.-Wirtsch.-Ing. W. A. Günthner is the head of the Institute for Material Handlings, Material Flow and Logistics, Technische Universität München. His e-mail address is: guenthner@fml.mw. tum.de and his web page can be found at http://www.fml.mw.tum.de.

SIMULATION EDUCATION IN LOGISTICS:
CASE STUDIES IN A VIRTUAL LEARNING ENVIRONMENT

Gaby Neumann
Logistics Knowledge Management Group
Otto-von-Guericke University of Magdeburg
39016 Magdeburg, Germany
E-mail: Gaby.Neumann@ovgu.de

KEYWORDS

Logistics Simulation, Simulation Education, Simulation Cases, E-Learning

ABSTRACT

To develop logistics simulation competence does not require just to advance knowledge on the simulation methodology in general and its applicability to logistics problems in particular, but even more to advance abilities and practical skills in creating and implementing a valid simulation model to solve a specific logistics problem. In traditional classroom-based learning this is well established via simulation exercises and labs. The challenge consists in incorporating those case studies in a virtual learning environment in such a way that both self-organised problem-solving at different levels of difficulty and automatic evaluation and assessment of learner-specific approaches and solutions become possible. The paper demonstrates how modelling and simulation might be linked with learning in a virtual logistics learning environment and introduces a virtual simulation lab to enable self-experienced learning-by-doing. With this, the paper sets up a framework for creating interactive exercises and case studies on-the-fly and contributes to improving simulation education in logistics.

INTRODUCTION

When discussing the link between computer simulation and learning in general we have to distinguish learning about simulation from learning by use of simulation (see Neumann et al. 2005):

- In *simulation-focussed learning*, computer simulation is the subject of learning, i.e. modelling concepts, simulation methodology and software tools for modelling and simulation are introduced, explained and practised to enable a student to apply computer simulation for solving practical problems either in general or specific to a particular application area such as logistics.
- In *simulation-based learning*, computer simulation (usually in combination with animation or any other visualisation technique) is used as a means of

supporting learning in a particular field such as logistics. Here, simulation models assist explanation and understanding of complex processes, illustrate abstract knowledge, demonstrate connections or interdependencies, and allow experiential learning or situated knowledge application.

This differentiation also applies when talking about cross-connections between simulation and e-learning, i.e. learning assisted by electronic media. According to Jonassen (2001) e-learning technology needs to be used for creating learning-by-doing and perceptual environments for problem-based learning, where you are immersed in making rapid-fire decisions, rushing to gain new information, utilizing the expertise of colleagues, and relying on your ability to create and store useful knowledge that will allow you to innovate and get your products to market way before your competitor. The role of simulation consists in supporting these learning processes:

- Computer simulations might be integral parts of interactive or even gaming-like environments. Students use them eventually being unconscious of simulation methodology and not necessarily having any theoretical simulation background.
- Simulation methodology can purposefully and consciously be applied to a particular context within specialized experimentation environments.
- Education and training of core simulation concepts themselves can make use of electronic technologies including the use of simulations for simulation learning.

E-learning and simulation both use the computer as their central medium, which makes their combination particularly convenient. No change of context or any additional tool is required. Even more, online simulations have been touted as the next big wave in e-learning (Chapman 2005): learners generally prefer them because they offer complexity, realism, and an opportunity to practise new skills in a risk-free environment. Administrators like them because they result in more motivated students and (perhaps most importantly) higher retention rates.

Consequently, simulation should also be used in a virtual logistics learning environment in both ways, as a means to support and intensify learning and as a particular subject of learning. The following sections will show how this might be achieved and discuss challenges to be mastered.

SIMULATION-BASED LOGISTICS E-LEARNING

Logistics – in its widest scope of understanding – is an extremely rapidly developing field of knowledge and practical applications. The competence profile of a logistics practitioner as requested by industry is a portfolio of wide-area knowledge in engineering and management, various abilities, such as problem finding and problem solving, systems' planning, design and operation, process planning, management and control or knowledge management and knowledge generation, and well-developed logistical and social skills.

Therefore, logistics education and training needs to be designed in such a way providing students with all of,

- *declarative* knowledge in the form of "knowing that…",
- *conceptual* knowledge as "knowing how" and
- *procedural* knowledge which is "know-how" in the domain.

As a matter of experience, students face particular problems in acquiring such kind of knowledge when it comes to:

- reducing real processes to the essentials;
- recognizing fundamental structures and processes;
- understanding parallel processes causing conflict;
- understanding cause-effect relations.

Here, simulation and animation are ideal tools to support knowledge acquisition through illustration, visualization and experimentation. Depending on the focus and purpose they might be integrated into e-learning in two ways:
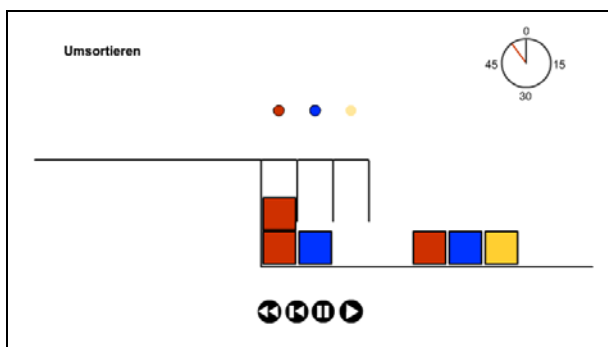

Figure 1: Flash animation to illustrate sequencing operations (in German)

Focus on illustration:
The content and appearance of animation sequences can be fixed in the form of Flash movies or pre-defined simulation-based trace files if animation is used only for

a comprehensible and vivid representation of knowledge to be imparted, for example, to illustrate processes and calculation algorithms or presenting alternative process variants for comparison (see Figure 1). In this case animation data are created or generated outside the e-learning environment in advance. The student just starts the animation sequence without paying any attention to how the animation has been developed nor to the means and methods used.

Focus on experimentation:
If knowledge acquisition may be supported by the independent planning, running and analysis of experiments in simulation-based exercises, animation is one of several possible forms for presenting simulation results produced by use of an existing simulation model. Students can change parameters within the range of freedom as defined by the model developer, but cannot create and implement a new simulation model (see Figure 2).
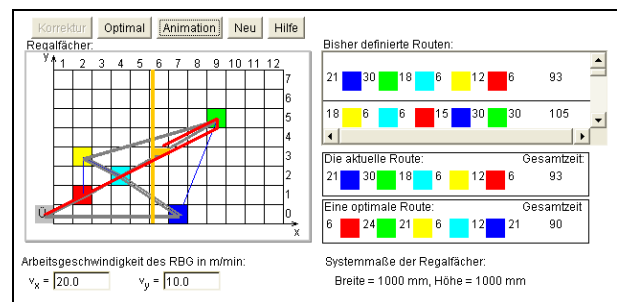

Figure 2: Java applet for solving routing problems (in German)

If experiments are based on a manageable quantity of pre-set parameters and/or model variations, respective simulations could even be run in advance and outside the e-learning environment. One trace file is generated and stored for each possible combination between a model and a set of experimental parameters and the student practically just selects the respective simulation results. Nevertheless and because of the psychology of learning it might be useful to let the student seemingly start a simulation run.

As soon as a user can vary parameter values according to individual intentions, a pre-simulation is no longer possible. Starting the simulation run really causes its execution inside the e-learning environment; simulation results are simultaneously collected for visualization. This requires a working simulation tool (or runtime version of the model) to be integrated in the e-learning environment, even though no model building functionality is accessible to the user.

Both scenarios for using simulation and animation in logistics learning have in common that no advanced simulation knowledge nor any specific model building competence is required by the students. They view or use models and should understand what is represented,

but do not need to know how those models are designed and implemented. Respectively, we cannot expect to develop any simulation competence with the student this way. To achieve this and provide the student with sufficient simulation knowledge, abilities and skills required for simulation-based problem-solving in a more complex logistics context, simulation-focused learning related to logistics applications has to be enabled and supported.

LOGISTICS SIMULATION-FOCUSSED E-LEARNING

Simulation is a method approved and established in logistics to support logistics planning, design and management. There is a high demand for logistics experts who at least know about simulation or even better are prepared to apply it to a particular problem. That is why simulation needs to be taught by transferring knowledge about the simulation methodology and its scientific background, but even more by developing and strengthening students' abilities and skills in purposefully applying it, efficiently and effectively organizing modelling and experimental phases as well as understanding, correctly interpreting and attractively presenting simulation results. This can be achieved best by implementing a constructivist approach of problem-based learning. Here, a specific problem to be solved by the student forms the starting point for initiating an individualized, self-organized, highly-motivated learning process. Whenever the student does not know or feels uncertain how to approach the problem, continue in the problem-solving process or apply a particular method/tool s/he might go to and through the respective learning module offered by the learning environment.

Consequently, simulation learning addresses two different goals: (i) *learning about* simulation methodology and applying this knowledge to practical problem solving, and (ii) *learning how* to use a particular simulation tool for model implementation and experimentation. The first requires some sophisticated learning material covering key aspects of logistics simulation; the latter needs to be designed in the form of an interactive tutorial specifically dedicated to understand and master particular simulation software. No matter which of these purposes the learning material serves, it needs to include a variety of interactive elements, support the gaining of self-experience and allow individual learning-by-doing.

Therefore, the virtual logistics learning environment currently comprises a number of modules belonging to the following main categories (see Table 1):

- *Description-oriented modules* summarize basic knowledge on simulation in general and fundamental knowledge on logistics simulation in particular. (What is …?)
- *Instruction-like modules* provide relevant procedural knowledge for identifying, specifying and solving logistics simulation problems or explain how a particular simulation package works via interactive tutorials. (How to …?)
- *Problem-based modules* enable individual knowledge application and evaluation within tests/quizzes at different levels of difficulty with extended feedback being provided to the user (assessments) or free knowledge application within an interactive supply chain scenario by running case studies at four levels of difficulty (case studies).

Table 1: Simulation learning modules in the virtual logistics learning environment

Module	Type	Learning Goals	Contents
Selected basics of simulation methodology	D	Know basic simulation terms and concepts Know mathematical foundations of simulation Understand the simulation methodology	What is simulation? How does simulation work? Introduction to stochastic theory and statistics
Introduction to logistics simulation	D	Know about the use of simulation in logistics problem-solving Understand requirements and challenges related to simulation application in logistics	Simulation application in logistics, incl. typical problems, aims, benefits, tools Logistics simulation database Challenges and procedures in a logistics simulation project
How to run a logistics simulation project	I	Know main phases and stages of a logistics simulation project Be able to run a logistics simulation project	Preconditions and constraints of a simulation project Problem analysis, model building, validation, experiments, presentation of results in a logistics simulation project
How to use DOSIMIS-3	I	Be able to implement simulation models, run simulations and analyse simulation results by use of the DOSIMIS-3 simulation package	Introduction to the simulation package and its functionality Instructions, guidelines and tutorials on how to implement a model, run simulations, view and analyse results
Assessment module	P	Be able to apply simulation knowledge in tests Be able to identify knowledge gaps	Questions and focused assignments on simulation terminology, methodology, procedure, application
Case study module	P	Be able to solve problems in authentic logistics cases	Case studies related to simulation application for logistics problem solving in a supply chain

Types of learning modules: D = description-oriented modules; I = instruction-like modules; P = problem-based modules

These modules are uploaded to a simulation course in a Moodle platform (see Figure 3), which is an Open Source course management system (http://moodle.org/) designed using sound pedagogical principles, to help educators create quality online courses. The software package is available in a wide variety of languages and thus used all over the world by universities, schools, companies and independent teachers. It provides various learning activities and communication functionality, such as forum and chat, glossary and wiki, workshop and assignment, lesson, quiz or survey.
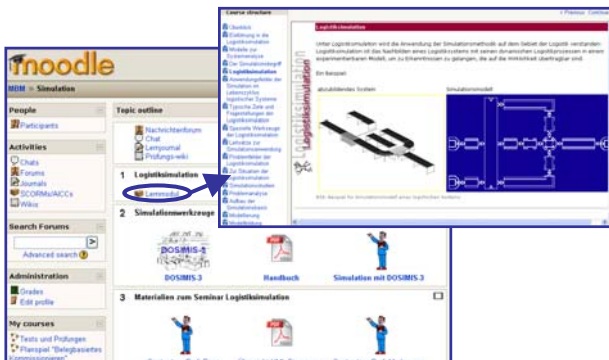
Figure 3: A logistics simulation course in Moodle

Furthermore, a commercial assessment package, Questionmark Perception, is linked into the Moodle platform enabling educators and trainers to author, schedule, deliver, and report on surveys, quizzes, tests and exams (http://www.questionmark.com). This way, a centralized pool of a wide variety of questions (ranging from multiple choice questions via drag-and-drop, hot spot, matching and knowledge matrix questions up to numeric questions – see Figure 4) has been created from which again and again new quizzes are automatically generated by combining on-the-fly questions of a varying level of difficulty and from different topic areas.
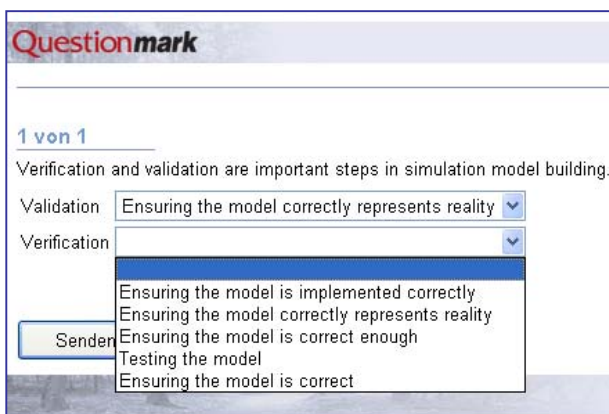
Figure 4: Exemplary question from the centralized pool

Once the student starts an assessment, a number of questions is selected at random from the available alternatives and put into the pre-defined quiz structure. This way, every time a student re-starts the same assessment a quiz new to him/her appears. Assuming

the pool of questions contains a sufficient number of questions, the own state-of-knowledge and conceptual understanding can repeatedly be tested without just reproducing what has been learned before.

As to the simulation package to be used, it was decided to integrate a professional tool from the real world rather than an academic toy. DOSIMIS-3 (http://www.sdz.de) is a simulation package 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. DOSIMIS-3 provides an extensive library of components from the material-flow and logistics world, enabling model-building by a few clicks on the basis of a well-structured conceptual model (see Figure 5). The simulation model then consists of the selected components specified by respective sets of technical, geometrical, topological and strategic parameters which are placed in a working area and logically linked to each other by so-called nodes, i.e. directed arrows free of any further information.
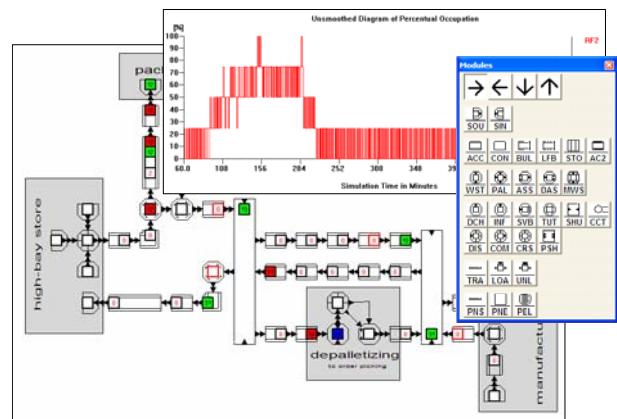

Figure 5: Logistics simulation using DOSIMIS-3

With this, model building and simulation is brought closer to the experts in the application area – in our case material flows and logistics enabling them to implement and use a simulation model themselves. It is very much self-explaining and therefore the perfect tool for logistics simulation education. DOSIMIS-3 runs on PC requiring Windows 95 or higher operating system. The reason for not looking for a fully web-based simulation package was very simple: Students should be able to cut down their times of having to be online to reduce communication costs. Instead a free download was organized, but guaranteeing that copyrights are not violated with this (see Neumann 2003).

Within this educational infrastructure and based upon these kinds of modules, customization and individualisation of a student's learning process and learning path according to his/her needs (resulting from current skills and individual levels of knowledge) become possible. There is no fixed sequence of

modules, but just hyperlinks between modules if deeper knowledge and understanding of a certain concept is assumed. In case of insufficient prior knowledge the student can follow this hyperlink and learn about fundamentals first before continuing at more specialized or complex levels. This way of a modularly designed learning environment enables the student to purposefully select the suitable module(s), but at the same time puts more self-responsibility for a successful learning process on the student. In the end (and according to requirements from practice), it only counts whether a student is able to successfully and efficiently perform in an authentic case study.

SIMULATION CASE STUDIES IN A VIRTUAL LOGISTICS LEARNING ENVIRONMENT

In contrast to the assessment module where a student answers questions within quizzes to give proof of conceptual knowledge, the case study module enables free knowledge application within an activity-oriented scenario. Here, the student has to become familiar with new situations and eventually deal with different problems. Thus, s/he is expected to demonstrate competences by understanding situations, applying methods and finding appropriate solutions in an effective and efficient way. To provide an experience as authentic as possible a fictitious supply chain of three partners – raw material supplier, manufacturer of potato chips and retailer with distribution centre and a number of shops – has been created (see Figure 6) which is not just being used in the simulation course, but also serves as case study scenario in other fields of logistics knowledge and learning, like e. g. Efficient Consumer Response (see Neumann and Krzyzaniak 2007). This becomes possible due to the strict separation of information and data (the scenario) from problems to be solved or tasks to be worked on.
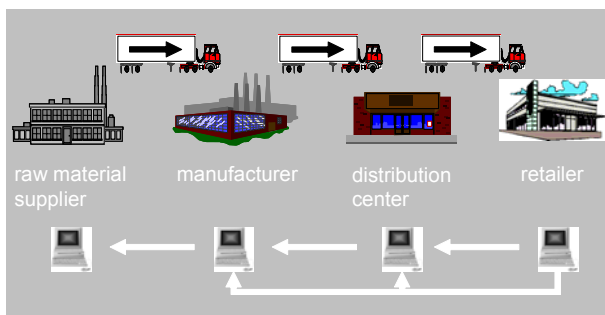

Figure 6: The supply chain scenario

Within the supply chain scenario each partner has a public website and a private Intranet including various company-specific information and data. They cover all aspects of the company from presenting its vision, mission, history, products and services (public information) to providing process descriptions, staff lists, resource data, performance data, customer/ purchase orders, stock levels etc. (private information).

Consequently, this scenario serves as case development framework; a specific case study is finally created by attaching problems or tasks to the supply chain as a whole or any of the partners individually. In the end, case studies with different targets and varying levels of difficulty result from this, which can aim at:

(i) *characterizing a situation by conceptual modelling* (The student describes the situation in the supply chain as a whole or any of the participating companies in particular e. g. by choosing from different descriptions presented. Furthermore, s/he matches this description against a collection of conceptual models selecting the most appropriate one for the subject of investigation. Feedback is returned directly on his/her decision.)

(ii) *identifying problems or potential improvements* (A list of problems and potential improvements is presented to the user from which s/he chooses the ones s/he identified. In addition to this, there are hot spots in the conceptual model on which students click to specify the problem area. Furthermore, students are asked to suggest an experimentation strategy for figuring out improvement potentials. Feedback is returned directly on his/her decision.)

(iii) *recommending actions to solve problems* (The student chooses from a set of actions the ones s/he recommends to use. Each of these possible choices is linked to a simulation model. The effects of his or her choices will be presented to him/her in the form of corresponding animation sequences and a set of simulation output data. Based upon this, the student might assess him/herself whether or not the chosen action has been completely solving the problem or contributed to problem solving at least. As consequence further actions might be suggested until the student comes to a final solution in his/her eyes. Feedback is returned directly on his/her submission.)

(iv) *applying methods to solve problems* (The user applies certain methods to solve the problems as identified and specified before by use of tools (EXCEL-based for input/output data analysis or simulation-based for output generation or others) and models provided to choose from and work with. Even more, the student might also develop and implement own models and run experiments using the simulation package provided. The results need to be self-assessed by the student e.g. according to an exemplary solution.)

It is obvious, that those different types of cases do require different methods to support the student (guidance-on-demand) and also different ways for gaining the student's results. Generally, the student has to work selectively with the material available and delivers his/her results by answering multiple-choice,

matching or questions of other types first. This ability to get into a situation, see problems and their causes, relate systems and process to models and link those findings to own knowledge about best practices or suggestions from theory as it is demonstrated by the student here forms the pre-condition for any purposeful and promising problem solving through experimentation and model building at a later stage.

Finally, this way of challenging students at increasing levels of difficulty also allows electronic assessment of at least parts of the complex problem solving processes related to simulation application in logistics. Introduction of e-assessment methods to individual simulation model building and experimentation still remains an unsolved problem, since there is neither just one correct solution to a logistics simulation problem nor exactly one successful problem solving procedure.

CONCLUSIONS

The symbiosis of e-learning and simulation brings many advantages to both fields. Computer simulations greatly help in the presentation of abstract ideas and processes, whose details and implications would otherwise be very hard to understand. On the other hand, e-learning also serves well in support of teaching simulation techniques. Since this is particularly true in logistics education, integration of animation and visualisation as well as simulation case studies into a virtual logistics learning environment was seen as a challenging, but very promising task.

In the end, within the presented logistics e-learning environment a number of learning modules form the basis for encouraging students' application of logistics simulation knowledge. To train students' ability to deal with certain aspects of a simulation project a pool of small examples and problems has been developed. They focus on typical methodological questions like e. g.:

- What are effects of certain control strategies?
- When to use simulation instead of analytical methods?
- How to cope with particular model building challenges?
- How do different validation technologies work and which technology should be applied by when?
- Which role simulation might play in solving optimisation problems?

In addition to this, a general framework, a number of specific problems and a set of dedicated e-assessment tools allow to run simulation case studies to provide a more authentic learning experience.

Further extension and enrichment of available challenges is planned to increase the level of attractiveness and even better achieve objectives of simulation education in logistics: to understand simulation methodology; to gain experiences in logistics model building and validation; to understand logistics phenomena through systematic experiments; to get familiar with a specific (professional) simulation package and practise its use.

REFERENCES

Blumstengel, A. 1998. *Entwicklung hypermedialer Lernsysteme*. Berlin, Wissenschaftlicher Verlag. (Developing hypermedia learning systems, in German)

Chapman, B. 2005. *Online Simulations 2005: A KnowledgeBase of 35+ Custom Developers, 300+ Off-the-Shelf Simulation Courses, and 40+ Simulation Authoring Tools*. brandon-hall.com.

Jonassen, D. 2001. *E-Learning to Solve Problems*. Keynote of ED-MEDIA World Conference on Educational Multimedia and Hypermedia 2001 (Tampere, Finland, June 25-30, 2001)

Neumann, G. 2003. Simulation in a problem-based e-learning environment for logistics. In: Y. Merkuryev, A. Bruzzone, G. Merkuryeva, L. Novitsky, E. Williams (editors), *Proceedings of HMS 2003 - The International Workshop on Harbour, Maritime & Multimodal Logistics Modelling and Simulation*, pp. 268-276.

Neumann, G.; Page, B.; Kreutzer, W.; Kiesel, G.; Meyer, R. 2005. Simulation and E-Learning. In: B. Page, W. Kreutzer, *The Java Simulation Handbook – Simulating Discrete Event Systems in UML and Java*, pp. 401-433, Aachen: Shaker.

Neumann, G.; Krzyzaniak, S. (2007). On the way towards an e-coach for supporting implementation and realisation of the Efficient Consumer Response concept. In: *LogForum* 3(2007)1, No. 1

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. She also has been part-time consultant in logistics simulation since 1991. 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. She is author/co-author of three books, one educational multimedia module on warehousing and a series of e-learning modules in logistics as well as of 50 book chapters and journal publications and about 80 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 involved in a couple of respective projects. Her e-mail address is gaby.neumann@ovgu.de.

MICRO-SIMULATION STUDY OF BUS PRIORITY ON ROADS
CARRYING HIGHLY HETEROGENEOUS TRAFFIC

Prof. Dr. V. Thamizh Arasan
P. Vedagiri
Transportation Engineering Division, Department of Civil Engineering,
Indian Institute of Technology Madras, Chennai, 600 036, India.
E.mail: arasan@iitm.ac.in

KEY WORDS

Heterogeneous Traffic, Simulation, Exclusive Bus Lane, Impact Assessment, Level of Service.

ABSTRACT

The desirable goal in passenger transportation is moving more people in fewer vehicles. This goal, in respect of road transport, can be attained by encouraging public transport modes like buses by assigning priority. This paper is concerned with the conceptual framework of a micro simulation model of highly heterogeneous traffic flow and application of the model to study the impact of exclusive bus lanes introduced on urban arterials. The impact is measured in terms of the reduction in speed of other categories of motor vehicles, due to the consequent reduction in road space. The results of the study indicates that it is possible to introduce exclusive bus lanes on selected urban roads to enhance the level of service of the bus, without much adverse impact on the level of service of other modes of road transport.

INTRODUCTION

The road traffic in the cities of developing countries like India is highly heterogeneous comprising vehicles of wide-ranging static and dynamic characteristics. The different types of vehicles present in the traffic in Indian cities can be broadly grouped into eight different categories as follows: 1. Motorized two-wheelers, which include motor cycles, scooters and mopeds, 2. Motorized three-wheelers, which include Auto-rickshaws–three-wheeled motorized transit vehicles to carry a maximum of three passengers and tempos–three-wheeled motorized vehicles to carry small quantities of goods, 3.Cars including jeeps and small vans, 4. Light commercial vehicles comprising large passenger vans and small four wheeled goods vehicles, 5. Buses, 6. Trucks, 7. Bicycles and 8.Tricycles, which include cycle-rickshaws -three-wheeled pedal type transit vehicles to carry a maximum of two passengers and three wheeled pedal type vehicles to carry small amount of goods over short distances. All these categories of vehicles share the same road space without any physical segregation. By virtue of their wide-ranging static and dynamic characteristics, the vehicles occupy any lateral position on the road depending on the availability of road space at a given instant of time without any lane discipline. Under these heterogeneous traffic flow conditions, the buses, being relatively large vehicles, find it difficult to maneuver through the mixed traffic and are subjected to frequent acceleration and deceleration leading to lower speed and discomfort to both the driver and passengers. This also results in enormous delay and uncertainty to bus passengers and consequently, the level of service of buses gets reduced considerably making buses a less attractive mode of transport.

The road traffic in Indian cities has grown at a very steep rate in the recent past making the available transport infrastructure inadequate. As augmentation of urban transport infrastructure is expensive, there is a need to find alternative solutions to the problem. One way is to devise methods for optimal utilization of the available infrastructure (road space) in such a way that the carrying capacity of the roadway, in terms of number of persons transported, is enhanced. This may be achieved by providing priority for buses, which will facilitate faster movement of more people in fewer vehicles resulting in reduced congestion. This paper is concerned with the conceptual traffic simulation framework of highly heterogeneous traffic flow and application of the model to study the impact of provision of exclusive bus lanes on urban roads.

SCOPE

Introduction of exclusive bus lanes requires comprehensive study of the flow characteristics of the traffic as a whole, and this can be done by using appropriate modeling technique. This paper is concerned with validation of a recently developed micro simulation model of heterogeneous traffic flow (Arasan and Koshy, 2005) and application of the model to study the impact of provision of reserved bus lanes on urban roads.

THE SIMULATION MODEL

Simulation models may be classified as being static or dynamic, deterministic or stochastic, and discrete or continuous. A simulation model which does not require any random values as input is generally called *deterministic*, whereas a *stochastic* simulation model

has one or more random variables as inputs. Random inputs lead to random outputs and these can only be considered as estimates of the true characteristics of the system being modeled. Discrete and continuous models are defined in an analogous manner. The choice of whether to use a discrete or continuous simulation model is a function of the characteristics of the system and the objectives of the study (Banks et al. 2004). For this study, a dynamic stochastic type discrete event simulation is adopted in which the aspects of interest are analysed numerically with the aid of a computer program.

As this study pertains to the heterogeneous traffic conditions prevailing in India, the available traffic simulation models, which are based on homogeneous traffic conditions, where clear lane and queue discipline exists, are not applicable to study the heterogeneous traffic flow characteristics. Also, the research attempts made to model heterogeneous traffic flow (e.g- Katti and Ragavachari, 1986; Marwah, 1995; Kumar and Rao, 1996; Khan and Maini, 2000) are limited in scope and do not address all the aspects comprehensively. Hence, there was a need to develop appropriate models to simulate heterogeneous traffic flow. Accordingly, a model of heterogeneous traffic flow, named HETEROSIM was developed (Arasan and Koshy, 2005). The modeling framework is explained briefly here to provide the background for the study. For the purpose of simulation, the entire road space is considered as single unit and the vehicles are represented as rectangular blocks on the road space, the length and breadth of the blocks representing respectively, the overall length and the overall breadth of the vehicles. The front left corner of the rectangular block is taken as the reference point, and the position of vehicles on the road space is identified based on the coordinates of the reference point with respect to an origin chosen at a convenient location on the space. The simulation model uses the interval scanning technique with fixed increment of time. For the purpose of simulation, the length of road stretch as well as the road width can be varied as per user specification. The model was implemented in *C++* programming language with modular software design. The flow diagram illustrating the basic logical aspects involved in the program is shown as Figure 1. The simulation process consists of the following major sequential steps related to traffic flow on mid-block section of roads: (1) vehicle generation, (2) vehicle placement, and (3) vehicle movement.

Vehicle Generation

In a stochastic traffic simulation process, the vehicles arrive randomly, and they may have varying characteristics (e.g. speed and vehicle type). Traffic-simulation models therefore, require randomness to be incorporated to take care of the stochasticity. This is easily done by generating a sequence of random numbers. For generation of headways, free speed, etc.,

of vehicles, the model uses several random number streams, which are generated by specifying separate seed values. Whenever a vehicle is generated, the associated headway is added to the sum of all the previous headways generated to obtain the cumulative headway. The arrival of a generated vehicle occurs at the start of the warm-up road stretch when the cumulative headway equals the simulation clock time. At this point of time, after updating the positions of all the vehicles on the road stretch, the vehicle-placement logic is invoked.
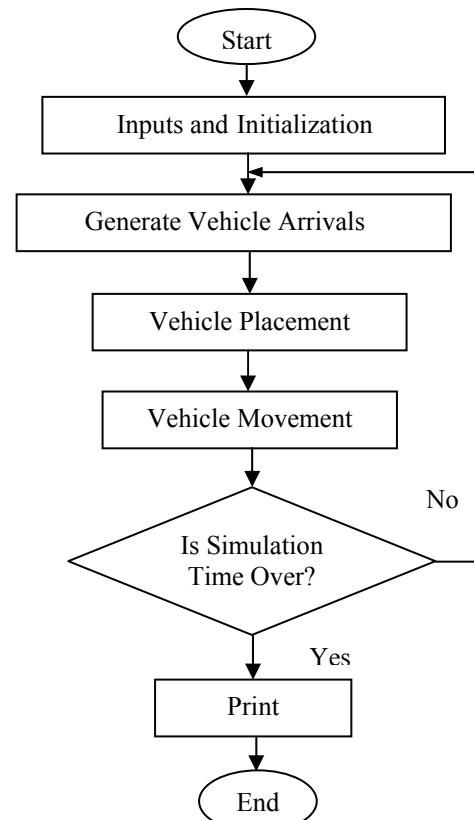

Figure 1: Flow Diagram of the Simulation Model

Vehicle Placement

Any generated vehicle is placed at the beginning of the simulation stretch, considering the safe headway (which is based on the free speed assigned to the entering vehicle), lateral gap and the overall width of the vehicle with lateral clearances. If the longitudinal gap in front is less than the minimum required safe gap, the entering vehicle is assigned the speed of the leading vehicle, and once again the check for safe gap is made. If the gap is still insufficient to match the reduced speed of the entering vehicle, it is kept as backlog, and its entry is shifted to the next scan interval. During every scan interval, the vehicles remaining in the backlog are admitted first, before allowing the entry of a newly generated vehicle.

Vehicle Movement

This module of the program deals with updating the positions of all the vehicles in the study road stretch

sequentially, beginning with the exit end, using the formulated movement logic. Each vehicle is assumed to accelerate to its free speed or to the speed limit specified for the road stretch, whichever is minimum, if there is no slow vehicle immediately ahead. If there is a slow vehicle in front, the possibility of overtaking the slow vehicle is explored. During this phase, the free longitudinal and transverse spacing available for the subject vehicle (fast moving vehicle), on the right and left sides of the vehicle in front (slow vehicle), are calculated. If the spacing is found to be adequate (at least equal to the movable distance of the vehicle intending to overtake plus the corresponding minimum spacing in the longitudinal direction and the minimum required lateral spacing in the transverse direction), an overtaking maneuver is performed. If overtaking is not possible, the fast vehicle decelerates to the speed of the slow vehicle in front and follows it. The model is also capable of displaying the animation of simulated traffic movements through mid block sections. The animation module of the simulation model displays the model's operational behavior graphically during the simulation runs. The snapshot of animation of traffic flow, obtained using the animation module of HETEROSIM, is shown in Figure 2. The model has been applied for a wide
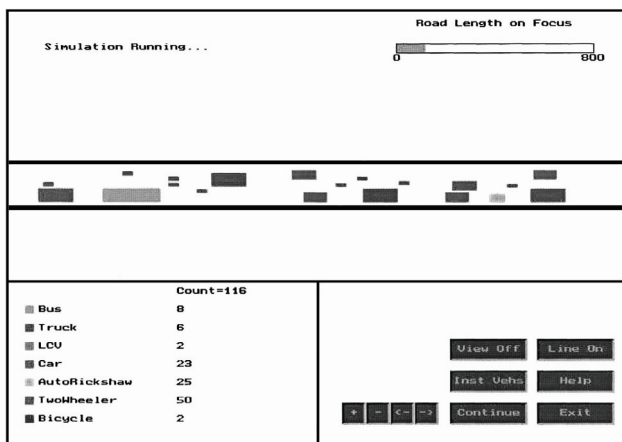
Figure 2: Snapshot of the Animation of Simulated Heterogeneous Traffic Flow

range of traffic conditions (free flow to congested flow conditions) and has been found to replicate the field observed traffic flow to a satisfactory extent through an earlier study (Arasan and Koshy, 2005).

DATA COLLECTION

Collection and analysis of data play a pivotal role in the development of successful simulation models. Field data should be gathered covering the ranges of anticipated roadway and traffic flow conditions. The required traffic data were collected by observing traffic flow on the Maraimalai Adigalar Bridge, near Saidapet, which falls in the southern part of the metropolitan area of Chennai, India. The bridge has a six-lane divided road with raised curbs on both sides and it is 250 m long. Since the study stretch is on the bridge, the road geometry is uniform and there is no interference to vehicular movement due to pedestrian traffic as the pedestrian walkway is segregated by a barricade. The traffic flow from Guindy side to Saidapet side was considered for the study. The width of the carriageway is 12m for the traffic stream considered. The traffic flow was recorded for one hour using a video camera mounted on the terrace of an adjacent building, which enabled recording of all the traffic flow characteristics at the same time. The video data were then transferred to computer for further processing. The inputs required for the model to simulate the heterogeneous traffic flow are: road geometry, traffic volume, and composition, vehicle dimensions, minimum and maximum lateral spacing between vehicles, minimum longitudinal spacing between vehicles, free speeds of different types of vehicles, acceleration and deceleration characteristics of vehicles, the type of headway distribution and the simulation period. The relevant characteristics of the vehicles are given in Table 1. The composition of the measured traffic volume on the study stretch is as depicted in Figure 3.

Table 1: Characteristics of Vehicles of the Heterogeneous Traffic

Vehicle Type	Dimensions in m		Lateral - Clearance Allowance in m		Free Speed in km/h	
(1)	Length (2)	Breadth (3)	Minimum (4)	Maximum (5)	Mean (6)	Standard Deviation. (7)
Bus	10.3	2.5	0.3	0.6	67	7
Truck	7.5	2.5	0.3	0.6	62	9
LCV	5.0	2.0	0.3	0.5	61	7
Car	4.0	1.6	0.3	0.5	72	7
M.Th.W.	2.6	1.4	0.2	0.4	48	8
M.T.W.	1.8	0.6	0.1	0.3	61	10
Bicycle	1.9	0.5	0.1	0.3	15	2

LCV- Light Commercial Vehicle, M.Th.W – Motorised Three Wheelers, M.T.W - Motorised Two Wheelers
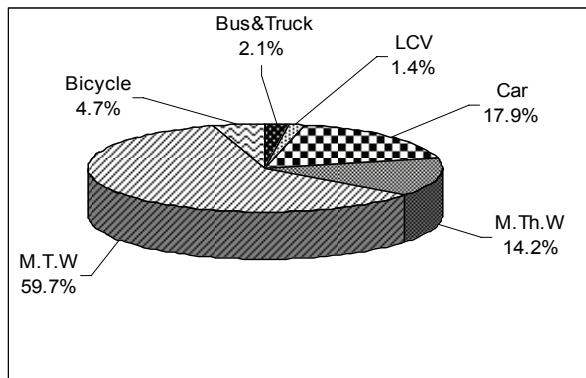
LCV- Light Commercial Vehicles, M.Th.W. – Motorised Three-Wheelers, M.T.W. - Motorised Two-Wheelers

Figure 3: Traffic Composition at the Study Road Stretch

The overall dimensions of all categories of vehicles, adopted from literature (Arasan and Koshy, 2005) are shown in columns (2) and (3) of table 1. The Minimum Clearance value pertaining to zero speed condition and the maximum Clearance corresponding to a speed of 60 km/h and more adopted from literature (Arasan and Koshy, 2005) are shown respectively in columns (4) and (5) of table 1. Any vehicle moving in a traffic stream has to maintain sufficient transverse clearances on both sides with respect to other vehicles/curb/ median to avoid side friction. The clearance value is assumed to vary linearly from minimum to maximum depending upon the speed of Vehicles. Lateral clearance allowance is the clearance share pertaining to a vehicle type. For example, if a bus and Motorised Three Wheeler are placed side by side, the minimum lateral clearance between the two vehicles will be 0.3 + 0.2 = 0.5 m. Knowledge of speed characteristics of various categories of vehicles is essential for the calibration and validation of simulation models. Free speeds of different types of vehicles are important input parameters for any traffic flow simulation model.

The free speeds of the different categories of vehicles were also noted by estimating the time taken by the vehicles to travel a trap length of 30 m on the study stretch of the road during lean traffic periods when the movement of vehicles are not hindered by the presence of other vehicles. The observed mean free speeds of various types of vehicles and the respective standard deviations are shown respectively, in columns (6) and (7) of table 1. The observed traffic volume and composition were given as input to the simulation process. The simulation runs were made with different random number seeds and the averages of the values were taken as the final model output. The model output includes the number of each category of vehicle generated, values of all the associated headways generated, number of vehicles present over a given road length at any point of time, number of overtaking maneuvers made by each vehicle, speed profile of vehicles, etc.

MODEL VALIDATION

For validating the simulation model, the traffic flow through a length of 1400 m of the study stretch was simulated. The observed roadway condition, traffic volume and composition were given as input to the simulation process. The inter-arrival time (headway) of vehicles was found to fit into negative exponential distribution and the free speeds of different categories of vehicles, based on the results of an earlier study, (Arasan and Koshy, 2005) were assumed to follow a Normal distribution. These distributions, then, formed the basis for input of the two parameters for the purpose of simulation. For the purpose of model validation, the field observed and simulated mean speeds of each of the categories of vehicles were compared. A comparison of the observed and simulated average speeds of the different types of vehicles is shown in Figure 4. It can be seen that the simulated speed values significantly replicate the field observed speeds for all vehicle types. Also, a paired t-test of null hypothesis of no mean difference was performed to check for the match between simulated and observed average speeds of vehicles. The calculated value of t (t_0) is 0.39 against the critical value (from 't' table) of 2.57. It was found that the observed and simulated average speeds agreed at a 5% level of significance (95% confidence limit).
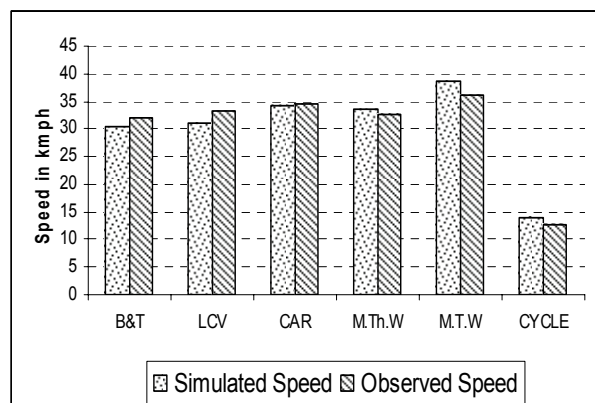
LCV- Light Commercial Vehicles, M.Th.W. – Motorised Three-Wheelers, M.T.W. - Motorised Two-Wheelers

Figure 4: Observed and Simulated Speeds

MODEL APPLICATION

The 'HETEROSIM' model can be applied to study a host of heterogeneous traffic scenarios on urban road links. Here, the application of the model is specific to study the impact of provision of an exclusive bus lane. For this purpose, a traffic composition representing the mean composition of traffic on the major roads of Chennai, India was considered (Figure 5).

The roadway width for the simulation was fixed as 11 m (3 lanes) in each direction (most common type of urban arterial in Indian cities). Out of the total width of 11 m,
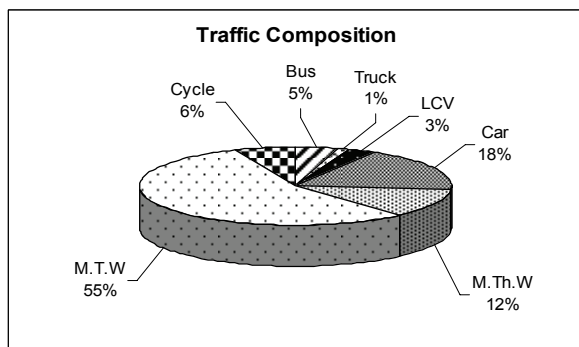
Traffic Composition

Figure 5: Representative Traffic Composition

a 1.5 m wide road space, adjacent to the curb, was reserved for bicycles (as is the normal practice in Indian cities). First, the traffic flow on the assumed arterial, without bus lane, was simulated. The simulation was run with volumes varying from a low level to the capacity flow condition. The speed flow relationship developed, based on the results of the simulation runs, is depicted in Figure 6.
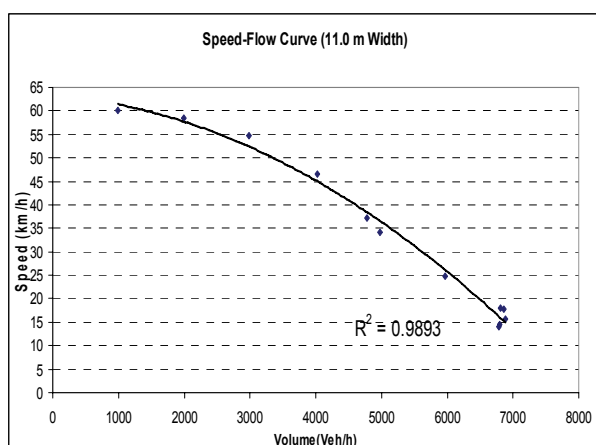

Figure 6: Speed Flow Curve for 11m Wide Urban Road

During validation of the model, it was found that three simulation runs (with three different random seeds) were sufficient to get consistent simulation output to replicate the field observed traffic flow. Hence, for model application also, the simulation runs were made with three random number seeds and the averages of the three values were taken as the final model output.

It can be inferred from the plot (figure 6) that the capacity of 11 m wide road space, when there is no exclusive bus lane (all vehicles mixed), is about 6900 vehicles per hour and the corresponding stream speed is about 16 km/h. As per the Indian Roads Congress-a statutory body responsible for development of codes and standards for road transport in India, guidelines (IRC, 106-1990), the acceptable level of service for urban roads is 'C' and the volume of traffic corresponding to this level of service can be taken as 0.7 times the capacity. Accordingly, here, the volume of traffic corresponding to level of service C is 0.7*6900=4830, say 4800 vehicles per hour.

To study the impact of provision of exclusive bus lane under the assumed road condition, for the purpose of simulation, an exclusive bus lane was introduced by the side of the median on the stretch of road, which will be used by all the buses and this roadway condition was given as the input to the model by holding the traffic volume and composition to be the same as for the previous case. The assumed layout of the road stretch with the proposed bus lane is shown in Figure 7. The simulation runs, after introducing the bus lane, were made similar to the previous case. For these (with bus lane) simulation runs, the traffic volume on the study stretch was varied, as in the case without bus lane, from 1000 to 7000 vehicles/h. A simulation run was also made with traffic volume corresponding to level of service C, namely, 4800 vehicles/h. The speeds maintained by the different types of vehicles for the different simulated traffic volume levels are shown in Table 2.
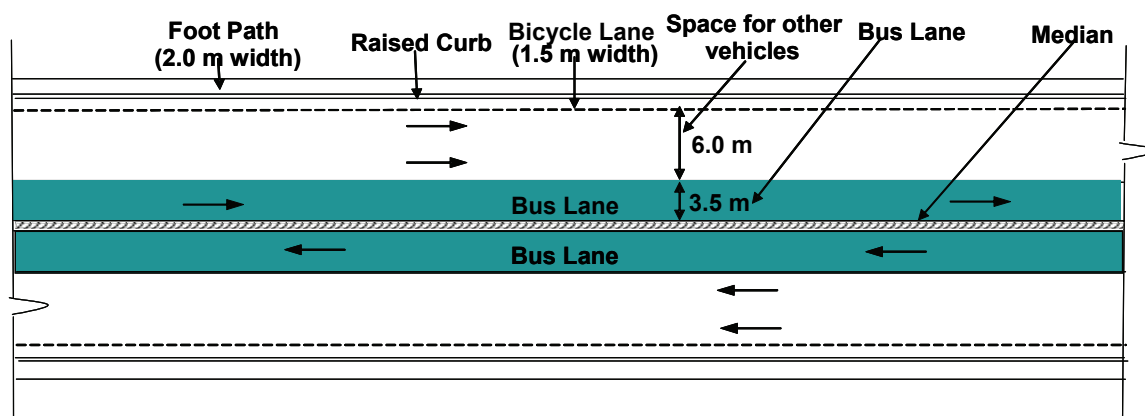

Figure 7: Schematic Layout of the Road Stretch with Exclusive Bus Lane

Table 2: Speeds of Different Categories of Vehicles on the Roadway with and without Bus Lanes

Traffic Volume (Vehicles/h) (1)	Road-way Condition (2)	Speed Maintained by Vehicles in Km/h						
		Bus (3)	Truck (4)	LCV (5)	Car (6)	M.Th.W (7)	M.T.W (8)	Bicycle (9)
1000	WoBL	63.5	55.6	60.8	72.8	48.9	62.9	14.5
	WBL	66.2	52.9	58.5	70.4	48.3	62.9	13.9
2000	WoBL	58.8	53.7	58.4	68.9	48.0	62.2	14.5
	WBL	65.9	48.4	53.9	61.9	46.6	60.6	13.3
3000	WoBL	51.2	45.8	51.1	60.9	46.1	59.4	14.5
	WBL	65.5	41.4	45.3	49.9	42.7	53.3	13.1
4000	WoBL	40.3	37.6	41.2	47.6	41.6	51.9	14.5
	WBL	65.2	29.3	30.4	33.6	32.0	37.3	12.9
4800	WoBL	30.3	28.2	31.7	35.3	34.2	41.8	14.4
	WBL	65.1	19.9	19.5	20.4	20.9	22.9	12.9
5000	WoBL	28.3	26.8	28.9	32.7	32.0	37.9	14.4
	WBL	65.1	16.7	16.3	16.9	17.2	19.3	12.8
6000	WoBL	20.9	20.5	21.6	23.3	23.7	27.4	13.9
	WBL	65.4	14.3	13.9	13.9	14.5	15.5	12.2
7150	WoBL	14.7	14.6	14.9	15.2	15.3	16.5	12.5
	WBL	64.5	13.7	13.5	13.4	13.5	14.9	12.2

WoBL: Without Bus Lane, WBL:With Bus Lane
LCV-Light Commercial Vehicles,   M.Th.W.-Motorised Three-Wheelers, M.T.W.-Motorised Two-Wheelers

From table 2, it can be seen that there is increase in the speed of bus due to provision of exclusive bus lane, at all volume levels. It can be noted that at lower volume levels (1000 & 2000 vehicles/h), due to provision of bus lane, there is marginal increase in bus speeds and marginal speed reduction to other vehicles. This is mainly because of the near-free-flow condition enjoyed by all categories of vehicles at low volume levels. Also, it can be noted that at higher volume levels (3000 vehicles/h and above), there is a significant speed improvement for bus and a steep decline in the speeds of other categories of vehicles. This implies that at higher volumes, there is a complex interaction among the different categories of vehicles and this creates a negative impact of the bus lane on the flow of all the other categories of vehicles. In the case of the other categories of motorised vehicles, the exact value of

speed reduction varies between vehicle types. It is important, while providing exclusive bus lanes, to see that the levels of service enjoyed by the other categories of vehicles do not deteriorate beyond the acceptable limit. In this context, it is reasonable to ensure level of service C (recommended as acceptable level of service on urban roads by Indian Roads Congress (IRC)) for the other categories of motor vehicles while providing an exclusive bus lane. Hence, there is a need to have information on the trend of speed variation of the stream of motorised traffic, excluding the buses, for roadway conditions, with and without bus lanes. Hence, two plots, on the same set of axes, depicting the variation of the stream speed, over volume to capacity ratio, for the two conditions of the road, were made as shown in Figure 8.

$R^2 = 0.9975$

$R^2 = 0.9938$
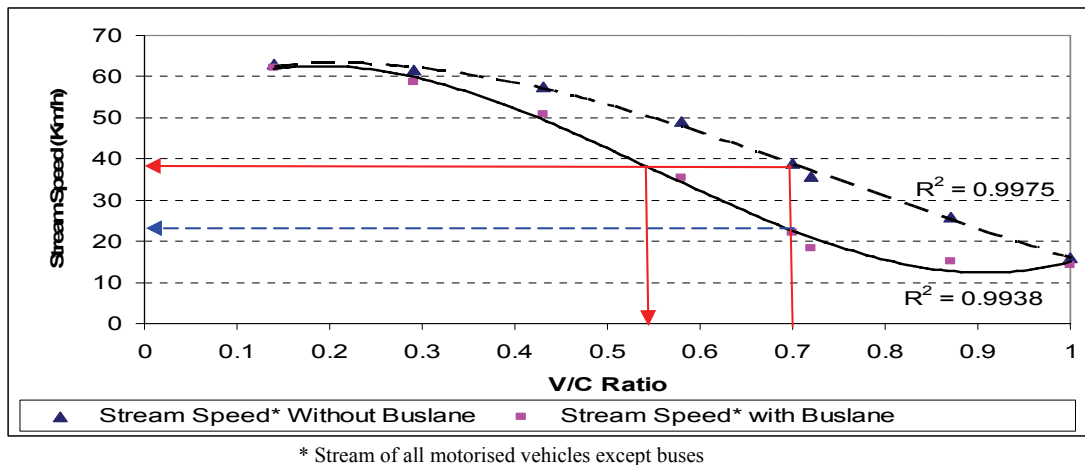
* Stream of all motorised vehicles except buses

Figure 8: Traffic Stream Speed on the Roadway with and without Bus Lane

It can be seen that the speed of the stream involving the other motorised vehicles, when no bus lane is provided, at level of service C (corresponding volume/capacity ratio of 0.7) is 39 km/h and the speed reduces to 22 km/h when a bus lane is provided. If it is desired to provide bus lanes without adversely affecting the level of service of the other categories of motor vehicles, then, the volume of traffic that will ensure the same speed for the other categories of motor vehicles corresponds to a V/C ratio of 0.53 as depicted in the figure. Thus, for the assumed road geometry and traffic composition, provision of exclusive bus lane may not adversely impact the minimum level of service required for the other categories of vehicles up to a traffic volume to capacity ratio of 0.53. The two plots also enable understanding of the traffic flow conditions, in terms of, speed over a range of volume-to-capacity ratios for the two roadway conditions (with and without bus lane).

CONCLUSIONS

The simulation model of heterogeneous traffic flow named, HETEROSIM is found to be valid for simulating heterogeneous traffic flow for the specific purpose of this study. It has been found through the study that for the assumed traffic composition, without any exclusive bus lane, the capacity of a 11 m wide road with 1.5 m wide bicycle track (included in the total width of 11 m), for one way movement of traffic, is about 6900 vehicles per hour.

If an exclusive bus lane is provided under the assumed roadway and traffic conditions, then, the maximum permissible volume to capacity ratio that will ensure a level of service of C for the traffic stream comprising all the other motor vehicles (except the buses), is about 0.53. When an exclusive bus lane is provided, the mean running speed of buses can be up to 65 km/h (depending on the bus stop spacing and the dwelling times, the corresponding mean journey speed may work out to be about 40 km/h) and the mean running speed of the stream of traffic comprising all the other motor vehicles (other than buses) enjoying level of service C, will be 39 km/h.

REFERENCES

Arasan, V.T. and R.Z.Koshy. 2005. "Methodology for Modeling Highly Heterogeneous Traffic Flow". *Journal of Transportation Engineering*, Vol. 131, No. 7, 544-551.

Indian Road Congress (IRC). 1990. "Guidelines for Capacity of Urban Roads in Plain Areas". IRC :106, New Delhi, India.

Jerry Banks; John S.Carson; Barry L.Nelson; and David M.Nicol. 2004. *Discrete-Event System Simulation.* Pearson Education, Singapore, Third Edition, 12-14.

Katti, V.K. and S.Raghavachari. 1986. "Modeling of Mixed Traffic with Speed Data as Inputs for the Traffic Simulation Models". *Highway Research Bulletin*, 28, Indian Roads Congress, 35-48.

Khan, S.I. and P. Maini. 2000. "Modeling Heterogeneous Traffic Flow". *Transportation Research Record*, No. 1678, 234- 241.

Kumar, V.M. and S.K. Rao. 1996. "Simulation Modeling of Traffic Operations on Two Lane Highways". *Highway Research Bulletin*, No. 54, Indian Roads Congress, 211-237.

Marwah, B.R. 1995. "Simulation and Animation of Traffic Flow on Highways with Intersections". *International Conference on New Horizons in Road and Road Transport*, Roorkee, 989-999.

AUTHOR BIOGRAPHIES

Prof. Dr. V. Thamizh Arasan is currently a full Professor in the Transportation Engineering Division of the Department of Civil Engineering of Indian Institute of Technology Madras, Chennai, India, which is one of the seven national level higher technological institutions in the country. He has professional experience of about 30 years in teaching research and consultancy in the area of Transportation Engineering. Travel demand modeling and traffic flow modeling are his areas of research interest. He has guided a number of doctoral degree students and has published more than eighty research papers in international and national journals and conference proceedings. Three of his papers published in journals have received awards for excellence in research. Prof. Arasan has successfully completed several sponsored research projects both at national and international levels. The international projects are: (i) on Development of Transportation Planning Techniques for Indian conditions in collaboration with the Technical University of Braunschweig, Germany and (ii) on Enhancing the Level of Safety at Traffic Signals in collaboration with the Technical University of Darmstadt, Germany. Prof. Arasan is member of several professional bodies and Technical committees of different government departments.

Mr. P.Vedagiri is a Ph.D. Scholar in Transportation Engineering Division, Department of Civil Engineering, Indian Institute of Technology Madras, Chennai, India. His doctoral research work is in the area of 'Heterogeneous Traffic Flow Modeling'. Mr. Vedagiri obtained his undergraduate degree in the area of Civil Engineering in the year 2002 and post graduate degree in the area of Urban Engineering in the year 2004 from Anna University, Chennai, India.

Bond Graph
Modelling and Simulation

ON THE ROLE OF ESSENTIAL ORDERS ON FEEDBACK DECOUPLING AND MODEL INVERSION: BOND GRAPH APPROACH

Mariem El Feki, Michael Di Loreto, Eric Bideaux, Daniel Thomasset, Wilfrid Marquis-Favre
Laboratoire Ampère (UMR 5005), INSA Lyon, France
Bât. St Exupéry
25, Avenue Jean capelle, Villeurbanne, 69621
E-mail: mariem.el-feki@ec-lyon.fr

KEYWORDS

Bond Graph, structure analysis, essential order, inverse model, decoupling, dynamic extension

ABSTRACT

The essential orders have an important role in the study of the systems decouplability as well as in the inverse model characterization. The aim of this paper is first to define essential orders on the bond graph model. Secondly, static and dynamic decoupling by bond graph approach is discussed and the dynamic extension order is defined. Finally, the dynamic compensation is physically located on the bond graph model and an approach to synthesize a model statically decouplable is suggested in order to define an adequate structure to the control requirements.

INTRODUCTION

Since its apparition the bond graph has been an efficient graphic support modelling for complex and multidisciplinary systems (Rosenberg et al. 1996; Karnopp et al. 2000; Gawthrop and Bevan 2007). However, modelling is not the only contribution of Bond graphs. Several works were achieved to develop system analysis using causal bond graphs, like stability analysis (Margolis 1984) or flatness study (Junco et al. 2005). The structure analysis by bond graph approach of linear time-invariant systems was introduced by (Rahmani et al. 1996; Dauphin-Tanguy, et al., 1999; Karim et al. 2003) to simplify decouplability study of these systems and without making any calculation. So, static decoupling conditions were defined on the bond graph model. The structural properties were generally used to the control problems (Loiseau 1986). Nevertheless some structural properties like essential orders (Commault et al 1986) have a very important significance when studying inverse model and consequently when modelling the systems to be controlled.

The purpose of this paper is to expand structure analysis by bond graph approach to define essential orders in order to interpret these orders on the inverse model and to discuss the system decouplability and the dynamic compensation if the system is invertible but not decouplable by static feedback.

In a first section, we recall preliminary results on the power lines, causal paths and model inversion. In second part, brief results on the infinite structure analysis are given and the extension of this analysis by bond graph approach is presented. In the next section, decoupling conditions are defined. Then, an approach to synthesize a model statically decouplable is discussed on the bond graph model. A mechanical example is established and treated in the different sections of the paper. The conclusion is given in the last section.

MODEL INVERSION: PRELIMINARY RESULTS

The inversion of bond graph model uses the bicausality concept which was introduced formally by Gawthrop (Gawthrop 1995 and 1997). This concept established new rules for causal assignment and it has contributed at the study of inverse problems (Gawthrop 2000; Ngwompo et al. 2005; Bideaux et al. 2006) and at design and sizing problems (Ngwompo et al. 1999 and 2001). The bicausality propagation is done according to SCAPI procedure (Ngwompo et al. 2005). This procedure uses the concept of independent power lines and disjoint Input-Output (I/O) causal paths.

Definition1 (Wu and Youcef-Toumi 1995)
A power line between two components is a series of power bonds and junction structure elements connecting these two components.

Definition2 (Ngwompo et al. 2001)
A causal path between two variables is a set of variables successively connected according to the causality assignment.

Definition3 (Sueur and Dauphin-Tanguy 1989; Sueur and Dauphin-Tanguy 1991)
An I/O causal path (u_j, y_i) is a path starting from a modulated command element (MS_e, MS_f, MR,..) and going to a detector y_i (D_e or D_f). Note that the power lines are an acausal concept while causal paths are defined for causal bond graphs.

Definition4 (Ngwompo et al. 2005)
Two I/O power lines are independent if they do not share a common variable: no effort or flow variable in common.

Definition5 (Ngwompo 1997)
Two I/O causal paths are said to be disjoint if they have no variable in common. Thus, "independent" is a property of power lines while "disjoint" is a property of causal paths.

Definition6 (Rahmani et al. 1992).
The length of a causal path between an output y_i and an input u_j is equal to the number of dynamic elements in integral causality met along this path in the bond graph in preferential integral causality (BGI).

The system is not structurally invertible if the bond graph model contains no set of independent I/O power lines. If the bond graph model contains a unique set of independent I/O power lines then a necessary and not sufficient condition of the structurally invertibility is verified (Ngwompo et al. 2005). If the bond graph model contains multiple sets of independent I/O power lines, then the necessary condition of the structurally invertibility consists of finding a minimal-length set of disjoint I/O causal paths. If a non-solvable causal cycle appears after the propagation of the O/I bicausality along the minimal length set of I/O causal paths, then it must choose an alternative minimal- length or a longer set of I/O causal paths and the bicausality propagation repeated. If a non-solvable causal cycle remains in the model then we can not conclude on the system invertibility. (Ngwompo et al. 2001). In this case, a modification of the system structure can be an adequate solution to synthesize a system with a bond graph model verifying the invertibility condition (For example, adding an action chain). Otherwise, by applying the SCAP (Karnopp et al, 1990) or eventually the MSCAP (Van Dijk, 1990), the causality assignment must be completed to the remaining acausal part of the bicausal bond graph model.

INFINITE STRUCTURE ANALYSIS

In this part, we recall some results on infinite structure analysis. From some works done on structural analysis by bond graph approach, we define the essential orders on bond graph model.

Let us consider the square linear time-invariant system described by state equations (1).

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx \end{cases} \qquad (1)$$

With $x \in R^n$ denotes the state vector, $u \in R^p$ denotes the control vector, and $y \in R^p$ denotes the output vector. The relative degree of the ith output is defined by:

$$n_i' = \inf \{k \in N \, / \, c_i A^{k-1} B \neq 0, \; k=1,\dots, n\} \qquad (2)$$

where c_i is the ith row of C.

The relative degree n_i' is equal, on a bond graph model, to the length l_i of the shortest causal path between the ith output (D_e or D_f) and all the inputs (S_e or S_f) (Rahmani et al. 1996). The order n_i' represents the minimal and necessary number of derivations of this output to make appear explicitly at least in one of the inputs, see (Bertrand et al. 1997; Dauphin-Tanguy et al. 2000).

Let us suppose that the system (1) is invertible with a transfer matrix $T(s)$ strictly proper, of rank p and defined by:

$$T(s) = C(sI - A)^{-1} B \qquad (3)$$

The Smith-McMillan form at infinity of $T(s)$ permits the determination of the infinite zero orders, it is given by:

$$T(s) = B_1(s)\Lambda(s)B_2(s) \qquad (4)$$

Where $B_1(s)$ and $B_2(s)$ are biproper (proper with proper inverse): $\det\left(\lim_{s\to\infty} B_i(s)\right) \neq 0$, i=1,2 and $\Lambda = \mathrm{diag}(s^{-n_1},\dots,s^{-n_p})$. Thus, the transfer matrix $T(s)$ has only zeros whose orders are $n_1,\dots,n_p$.

On the bond graph model, the number of the infinite zeros of (A, B, C) is equal to the number of disjoint I/O causal paths and their orders are computed as in equation (5), where L_k is the lowest sum of the lengths of the k I/O disjoint causal paths (Dauphin-Tanguy, et al., 1999 and 2000).

$$\begin{cases} n_1 = L_1 \\ \vdots \\ n_p = L_p - L_{p-1} \end{cases} \qquad (5)$$

Thus, the following property is deduced:

$$L_p = \sum_{i=1}^{p} n_i \qquad (6)$$

The essential orders n_{ie} were defined by (Commault et al. 1986) for solving row by row decoupling problems. They are determined from the Toeplitz matrices Γ_μ (for $\mu \geq 1$) defined by:

$$\Gamma_\mu = \begin{bmatrix} CB & 0 & 0 & \cdots & 0 \\ CAB & CB & 0 & & 0 \\ \vdots & & & \ddots & \vdots \\ \vdots & & & & 0 \\ CA^{\mu-1}B & \cdots & \cdots & CAB & CB \end{bmatrix}$$

Let us recall that a row w_i of a given matrix W is said to be essential if it is not linearly dependent of other rows of W (Cremer 1971).

For a right-invertible system (A, B, C), with n_{sup} its supremal order of infinite zeros (n_{sup}= sup{n_i}, i=1,…,p), the essential order of the ith output is defined by (Commault et al. 1986):

$$n_{ie} = \inf \{k \in \text{N} / [c_iA^{k-1}B|…|c_iB|0…0]$$
$$\text{is essential in } \Gamma_{n_{sup}} \} \qquad (7)$$

Then, the following proprieties are deduced:

(i) $n_{ie} \geq n_i' \quad \forall$ i=1,…,p

(ii) $n_{ie} \leq n_{sup} \quad \forall$ i=1,…,p

(iii) $\sup\{n_{ie}\} = n_{sup} \quad \forall$ i=1,…,p

For any right-invertible system, the essential orders can be expressed as follows (Commault et al. 1986):

$$n_{ie} = \sum_{j=1}^{p} n_j - \sum_{k=1,k\neq i}^{p} n_k' \quad \forall\, i \in \{1,…p\} \qquad (8)$$

The essential orders have an important signification on the inverse model. Let us recall that the transfer matrix $T(s)$ has only infinite zeros. Thus we can deduce that the transfer matrix of the inverse system $T^{-1}(s)$ has only infinite poles. From this fact, Commault, et al. (1986) showed that the essential order n_{ie} corresponds to the order of the pole at infinity of the ith column of $T^{-1}(s)$. Thus, we can deduce that the essential order n_{ie} is defined as the highest derivation order of the ith output appearing in the inverse model

In order to simplify the determination of essential orders, a new method is established to determine these orders directly from the bond graph model and without making any computation.

Property1
On the bond graph model, the essential order of the ith output is determined as follows:

$$n_{ie} = L_p - \sum_{j=1,j\neq i}^{p} l_j \quad \forall\, i \in \{1,…p\} \qquad (9)$$

On the bicausal bond graph model, the essential order of the ith output can be determined directly:

$$n_{ie} = O_h(O_i / I_j) \quad \forall\, j \in \{1,…p\} \quad (10)$$

$O_h(O_i / I_j)$ corresponds to the highest order of the causal paths between the ith output and all the inputs on the bicausal bond graph. The order of a causal path on

bicausal bond graph corresponds to the difference between the number of the dynamic elements (I or C) in derivative causality and the number of the dynamic elements in integral causality met on this path.

The property (10) is a conjectural and enables the essential orders, directly from the bicausal bond graph, to be deduced. Its efficiency has been proved on several examples but a demonstration proving its generality will be the topic of a coming work.

Let us consider the mechanical system illustrated by Figure 1. The state, input and output vectors are given by: $x = [p_1\ p_2\ p_3\ q_1\ q_2]^T$; $u = [F_1\ F_2]^T$ and $y = [V_1\ V_2]^T$.
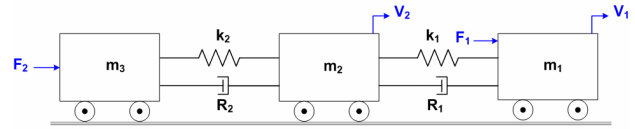

Figure 1: Mechanical system

Let us consider that the shock absorber R_2 has a negligible effect on the system. The bond graph model in preferential integral causality without considering this element is shown on Figure 2.
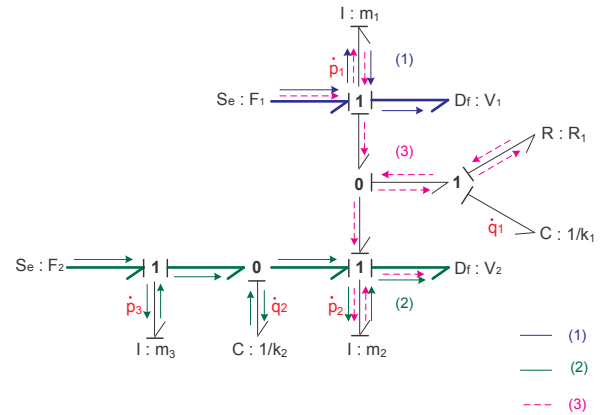

Figure 2: Bond graph model in preferential integral causality

This system has two independent I/O power lines: (F_1, V_1) and (F_2, V_2), then necessary condition of the invertibility is satisfied. The disjoint I/O causal paths are (F_1, V_1): (1) and (F_2, V_2): (2). This lead to L_2 =1+3=4 as length of the disjoint I/O causal paths. The relative degrees of the outputs are $l_1 = n_1' = 1$; $l_2 = n_2' = 2$, the essential orders are: $n_{1e} = L_2 - l_2 = 2$; $n_{2e} = L_2 - l_1 = 3$ and the orders of the infinite zeros are: $n_1 = L_1 = 1$; $n_2 = L_2 - L_1 = 3$.

Let us define the bicausal bond graph model on Figure 3 by replacing each input, respectively each output, by a double detector, respectively a double source, and by propagating the bicausality along the O/I power lines.

Figure 3: Bicausal bond graph model

The causal path (V_1, F_2): (1) is the causal path of the highest order connecting the output V_1 to an input, then $n_{1e} = O_h(V_1, F_2) = 2$. In the same way, The causal path (V_2, F_2): (2) is the causal path of the highest order connecting the output V_2 to an input, then $n_{2e} = O_h(V_2, F_2) = 3$. Thus, we can conclude that defining the essential orders on the bicausal bond graph model is easier than defining these orders on the bond graph model in preferential integral causality. The inverse model is directly computed from the bicausal bond graph model (Figure 3).

$$\begin{cases} \dfrac{d}{dt}(q_1) = V_1 - V_2 \\[2mm] F_1 = m_1 \dfrac{d}{dt}(V_1) + R_1(V_1 - V_2) + k_1 q_1 \\[2mm] F_2 = \dfrac{m_2 m_3}{k_2} \dfrac{d^3}{dt^3}(V_2) + \dfrac{m_3 R_1}{k_2}(\dfrac{d^2}{dt^2}(V_2) - \dfrac{d^2}{dt^2}(V_1)) \quad (18) \\[2mm] \quad - \dfrac{m_3 k_1}{k_2} \dfrac{d}{dt}(V_1) + \left(m_2 + m_3 \dfrac{(k_1 + k_2)}{k_2}\right) \dfrac{d}{dt} V_2 \\[2mm] \quad + R_1(V_2 - V_1) - k_1 q_1 \end{cases}$$

Let us denote by d_i i =1,2 the necessary number of derivations of outputs in the inverse model, then $d_1 = 2$ and $d_2 = 3$. We remark that $(d_1 = n_{1e}) \geq n_1'$ and $(d_2 = n_{2e}) \geq n_2'$, so the highest derivation order of each output in the inverse model is different from the relative degree. This number corresponds to the essential order n_{ie}.

FEEDBACK DECOUPLING ANALYSIS

In this section, static decoupling conditions will be presented by algebraic and bond graph approaches and the case of dynamic extension will be treated.

Let us recall that the invertibility implies the decoupling of the square linear system (1). Thus, after verifying the system invertibility, we can be interested by feedback decoupling.

The square system $(p \times p)$ defined by the triplet (A, B, C) is decouplable by a static feedback of the form $u = Kx + Lv$ with L a non-singular matrix if and only if one of these three equivalent conditions is satisfied:

(i) The decoupling matrix B^* is non-singular (Falb and Wolovich 1967):

$$B^* = \begin{pmatrix} c_1 A^{n_1'-1} B \\ \vdots \\ c_i A^{n_i'-1} B \\ \vdots \\ c_p A^{n_p'-1} B \end{pmatrix} \quad (11)$$

(ii) (Dion and Commault 1993):

$$\sum_{i=1}^{p} n_i = \sum_{i=1}^{p} n_i' \quad (12)$$

(iii) (Commault et al. 1986):

$$n_{ie} = n_i' \quad \forall\, i \in \{1,..,p\} \quad (13)$$

Gilbert (1969) showed that there is a class of invertible systems not decouplable by static feedback but which require a dynamic extension to achieve decoupling by feedback. In this case, the essential order n_{ie} of the ith output is strictly superior to its relative degree n_i'.

An invertible system represented by a bond graph model is decouplable by a static feedback if the following equivalent conditions are verified (Rahmani et al. 1996):

(i) $$L_p = \sum_{i=1}^{p} l_i \quad (14)$$

(ii) $$\{n_i\} = \{n_i'\} \quad (15)$$

Then an invertible model is decouplable by a static feedback if the lowest sum of the lengths of the p I/O disjoint causal paths is equal to the sum of the lengths of the shortest causal paths between every output i and all the inputs.

Property2
The bond graph model is decouplable by a static feedback if the following condition is satisfied:

$$l_i = L_p - \sum_{j=1, j \neq i}^{p} l_j \quad \forall\, i \in \{1,...p\} \quad (16)$$

If this decoupling condition is not satisfied, it is always possible to find the inverse model. In this case, a natural dynamic extension occurs during the construction of the inverse model.

If the system is not statically decouplable then the dynamic extension order is calculated on the model bond graph as follows:

$$n_{de} = L_p - \sum_{j=1}^{p} l_j \qquad (17)$$

Let us study the decouplability of the mechanical system (Figure1). In fact, we remark that $n_{1e} \neq n_1'$ and $n_{2e} \neq n_2'$, then we can deduce that the system is not decouplable by static feedback. A dynamic compensation is necessary to decouple the model and its order is computed as follows: $n_{de} = L_2 - l_1 - l_2 = 4 - 1 - 2 = 1$.

FEEDBACK DECOUPLING SYNTHESIS

In this section, a method will be given to synthesize a statically decouplable system by bond graph approach.

Obviously, if a system is not statically decouplable, then a dynamic compensation is necessary. The dynamic compensation consists of adding dynamics in the model. The classic method of dynamic compensation consists on the insertion of additional integrators in order to delay the effect of the fastest entry. This mathematical method enables automatically to increase the model order and is made independently from the physical phenomena of the system. The dynamic extension orders will be used here to determine what phenomena has to be considered or added in the initial model to obtain a new system that is statically decouplable. A new design can be applied to simplify the command without using mathematical methods. The synthesis of the new design is closely linked to the structure and the nature of the system under study. This enables the system to be reviewed and considers the physical phenomena that have been neglected or adds physical phenomena having insignificant effects on the system (e.g. adding a low inductance coil in an electric network and which does not affect the system behaviour). This approach is an aid to design systems statically decouplable (that will be easier to control) and it does not require any calculation.

From the bond graph model, we can deduce a number of informations that will help to analyze the initial system and identify the bond graph structure that make it not statically decouplable. In fact, if the system is not statically decouplable then for one or several outputs there is an I/O causal path shorter than the disjoint I/O causal path associated to this output. In order to identify these outputs, the I/O causal paths (u_j, y_i) which have served to compute the relative degrees (the shortest I/O causal paths associated to the outputs) must be located. For each output, the length of the shortest path (u_j, y_i) must be compared at the length of the disjoint I/O causal path (u_k, y_i) associated to this output; if the length of the I/O causal path (u_j, y_i) is the inferior one, then one of

this two causal paths must be modified in order to delay the fastest input. According to the modelling hypotheses, the possible modifications of the system consist in the addition or the deletion of a bond graph structure with only one set of junctions, dissipative and dynamic elements in order to increase the length of the shortest I/O causal path or to reduce the length of the disjoint I/O causal path. The order of this modification corresponds to the dynamic extension order (17).

Let us synthesize a decouplable model by a static feedback from the initial model (Figure 2). This model is not decouplable because the I/O causal path (F_1, V_2) is shorter than the disjoint I/O causal path (F_2, V_2). So the length of the causal path (F_1, V_2) must be increased or the length of the causal path (F_2, V_2) must be shortened. The consideration of the shock absorber R_2 (that has been neglected in the beginning) in the model renders the causal path (F_2, V_2) shorter, then the associated bond graph model in preferential integral causality is defined by Figure 4 and the associated bicausal bond graph model is defined by Figure 5.
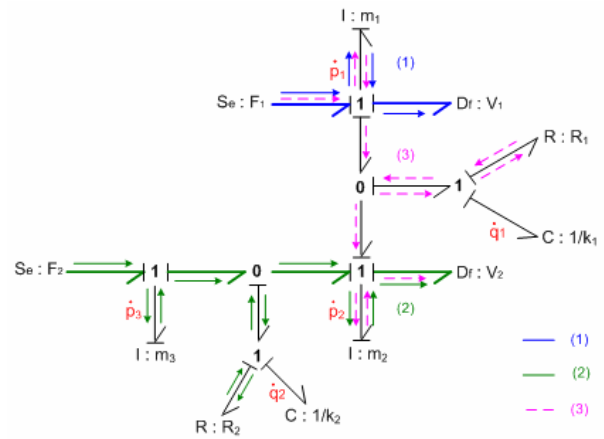
Figure 4: Bond graph model with the consideration of the shock absorber R_2
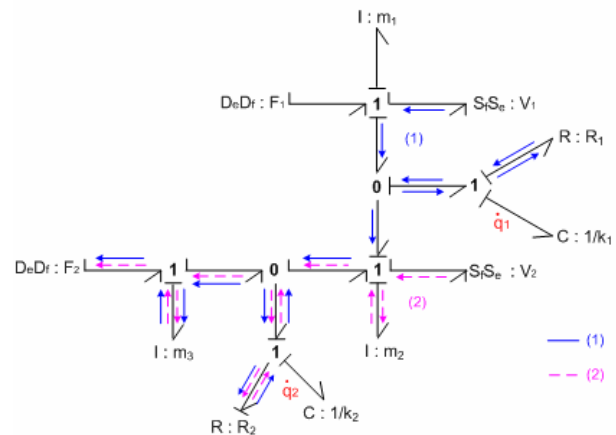
Figure 5: The associated bicausal bond graph model

The model is still structurally invertible and the unique set of disjoint I/O causal paths is $\{(F_1, V_1); (F_2, V_2)\}$, its length is then $L_2 = 1+2 = 3$. The structural properties are: $l_1 = n_1' = 1$; $l_2 = n_2' = 2$; $n_{1e} = 1$; $n_{2e} = 2$; $n_1 = L_1 = 1$; $n_2 = L_2 - L_1 = 2$. Note that $n_{1e} = n_1'$ and $n_{2e} = n_2'$, then the model is decouplable by static feedback.

The inverse model is the following:

$$
\begin{cases}
\dot{q}_1 = V_1 - V_2 \\
\dot{q}_2 = \dfrac{m_2}{R_2}\dot{V}_2 + \dfrac{R_1}{R_2}(V_2 - V_1) - \dfrac{k_1}{R_2}q_1 - \dfrac{k_2}{R_2}q_2 \\
F_1 = m_1\dot{V}_1 + R_1(V_1 - V_2) + k_1 q_1 \\
F_2 = \dfrac{m_2 m_3}{R_2}\ddot{V}_2 + \left(m_2 + m_3 + \dfrac{m_3 R_1}{R_2} - \dfrac{m_2 m_3 k_2}{R_2^2} \right)\dot{V}_2 \quad (19) \\
\qquad - \dfrac{m_3 R_1}{R_2}\dot{V}_1 + \left(R_1 + \dfrac{m_3 k_1}{R_2} - \dfrac{m_3 R_1 k_2}{R_2^2} \right)(V_2 - V_1) \\
\qquad + \left(\dfrac{m_3 k_1 k_2}{R_2^2} - k_1 \right)q_1 - \dfrac{m_3 k_2^2}{R_2^2}q_2
\end{cases}
$$

The highest derivation order of each output corresponds now to the relative degree. The model order has not been increased. In fact, it was enough to consider a simple dissipative element to decouple the model.

It shows that modifying the modelling hypotheses may be a possible way to change the structural properties of the system and enables the static decoupling of the system (It is the case of the shock absorber R_2 in the proposed example).

CONCLUSION

It has been shown that the essential orders of each output system can be graphically obtained from the bond graph model. This approach enables to deduce, directly after the bond graph modelling, relevant information on the design and the control of the system.

A new condition for decoupling by static feedback (using essential orders) has been defined on the bond graph model. The decoupling by dynamic extension was discussed. The dynamic extension order can be directly deduced from the bond graph model. Finally, an approach enabling the synthesis a model statically decouplable has been defined. This approach is linked to the structure and the behaviour of the system; It enables the system reviewing: Analyze all the neglected physical phenomena and add others elements to the system if it is necessary. The contribution of this approach appears at the design step where it is easy to modify the model and consequently the system in order to define an adequate structure to the command requirements.

REFERENCES

Bertrand, J.M.; C. Sueur and G. Dauphin-Tanguy. 1997. "On the finite and infinite structures of bond graph models." *Proceedings of the 1997 IEEE International Conference on Systems Man and Cybern*etics (Orlando USA, Oct), vol.3, 2472-2477.

Bideaux, E.; J. Laffite; A. Derkaoui; W. Marquis-Favre; S. Scavarda and F. Guillemard. 2006. "Design of a Hybrid Vehicle Powertrain using an Inverse Methodology." *Journal Européen des Systèmes Automatisés*, vol.40, pp.279-290.

Commault, C.; J. Descusse; J.M. Dion; J.F. Lafay and M. Malabre. 1986. "New decoupling invariants: the essential orders." *International Journal of Control*, vol.44, No.3, 689-700.

Cremer, M. 1971. "A precompensator of minimal order for decoupling a linear multivariable system." *International Journal of Control*, vol.14, No.6, 1089-1103.

Dauphin-Tanguy, G.; A. Rahmani and C. Sueur. 1999. "Bond graph aided design of controlled systems." *Simulation Practice and Theory*, vol.7, No.5-6, 493-512.

Dauphin-Tanguy, G. et al. 2000. *Les bond graphs*, 383p, Hermes Science, Paris.

Dion, J.M. and C. Commault.1993. "Feedback decoupling of structured systems." *IEEE Transactions on Automatic Control*, vol.38, No.7, 1132-1135.

Falb, P.L. and W.A. Wolovich. 1967. "Decoupling in the Design and Synthesis of Multivariable Control Systems." *IEEE Transactions on Automatic Control*, vol.AC-12, No.6, 651-659.

Gawthrop, P.J. 1995. "Bicausal bond graphs." *Proceedings of International Conference on Bond Graph Modelling and Simulation: ICGBM'95*(Las Vegas), 83-88.

Gawthrop, P.J. 1997. "Control system configuration: Inversion and bicausal bond graphs." *Proceedings of International Conference on Bond Graph Modelling and Simulation: ICGBM'97*(Phoenix), 97-102.

Gawthrop, P.J. 2000. "Physical Interpretation of Inverse Dynamics Using Bicausal Bond Graphs." *Journal of the Franklin Institute*, vol.337, 743-769.

Gawthrop, P.J; and G.P. Bevan. 2007. "Bond graph modeling." *IEEE Control Systems Magazine, vol.27, 24-45.*

Gilbert, E.G. 1969. "Decoupling of multivariable systems by state feedback." *S.I.A.M. Journal of Control*, vol.7, No.1, pp50-63.

Junco, S.; C. Sueur and G. Dauphin-Tanguy. 2005. "Nonlinear control of a series direct current motor via flatness and decomposition in the bond graph domain." *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol.219, No.3, 215-230.

Karim, A.; C. Sueur and G.Dauphin-Tanguy.2003. "Non-regular static state feedback for linear bond graph models." *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol.217, No.2, 61-71.

Karnopp, D. C.; D. L. Margolis and R.C. Rosenberg. 1990. *"System Dynamics: A Unified Approach."* Ed. John Wiley, New York.

Karnopp, D.C.; D.L. Margolis and R.C. Rosenberg. 2000. "*System dynamics: modeling and simulation of mechatronic systems.*" 507p., John Wiley and sons, New-York.

Loiseau, J.J. 1986. *"Sur la modification de la structure à l'infini par retour d'état statique."*PhD thesis, Nantes University.

Margolis, D.L. 1984. "Bond graphs for vehicle stability analysis." *International Journal of Vehicle Design*, vol.5, No.4, 427-437.

Ngwompo, R.F. 1997. "Contribution au dimensionnement des systèmes sur des critères dynamiques et énergétiques : approche bond graph." PhD Thesis, Labiratory : Automatique Industrielle, Institut National des Sciences Appliquées, Lyon-France, 216p.

Ngwompo, R.; S. Scavarda and D. Thomasset. 1999. "Inversion of linear time invariant SISO systems modeled by bond graph." *Journal of the Franklin Institute*, vol.333B, No.2, 157-174.

Ngwompo, R.F.; S. Scavarda and D. Thomasset. 2001. "Physical model-based inversion in control systems design using bond graph representation, Part 1: theory." *Proceedings of the Institution of Mechanical Engineers. Part I: Journal of Systems and Control Engineering*, vol.215, No.2, 95-103.

Ngwompo, R.F.; E. Bideaux and S. Scavarda. 2005. "On the role of power lines and causal paths in bond graph – based model inversion." *Proceedings of International Conference on Bond Graph Modelling and Simulation: ICGBM'05*, vol.37(1), 5-10.

Rahmani, A.; C. Sueur and G. Dauphin-Tanguy. 1992. "Formal determination of controllability/observability matrices for multivariable systems modelled by bond graph." *Proceeding of IMACS/SICE Int. Symposium of Robotics, Machatronics and Manufacturing System'92*, 573-580.

Rahmani, A.; C. Sueur and G. Dauphin-Tanguy .1996. "On the infinite structure of systems modelling by bond graph: feedback decoupling." *Proceedings of the 1996 IEEE International Conference on Systems Man and Cybern*etics(Beijing China,Oct), vol.3, 1617-1622.

Rosenberg, R.C; M.K. Hales and M.A. Minor. 1996. "Engineering icons for multidisciplinary systems." *American Society of Mechanical Engineers Dynamic Systems and Control Division* (New-York, Nov), vol.98, 665-672.

Sueur, C. and G. Dauphin-Tanguy. 1989. "Structural controllability/observability of linear systems represented by bond graphs." *Journal of the Franklin Institute.*, vol.326, No.6, pp869-883.

Sueur, C. and G. Dauphin-Tanguy. 1991. "Bond graph approach for structural analysis of MIMO linear systems." *Journal of the Franklin Institute*, vol.328, No.1, 55-70.

Van Dijk, J. 1994. "On the role of bond graph causality in modelling mechatronic systems." PhD thesis, Dept of Electrical engineering: University of Twente, Enschede, Netherlands, 220p.

Wu, S.-T. and K. Youcef-Toumi. 1995. "On the relative degrees and zero dynamics from physical system modelling." *Journal of Dynamic Systems, Measurement and Control Transactions of the American Society of Mechanical Engineers "ASME" (New-York USA)*, vol.117, No.2, 205-217.

AUTHOR BIOGRAPHIES

MARIEM EL FEKI was born in Ariana, Tunisia and went to the National Institute of Applied Sciences and Technology (Tunis, Tunisia). She obtained engineering diploma in industrial data processing and automatic (2007). In the same year, she obtained the Master of Research degree in automation system engineering from National Institute of Applied Sciences, Lyon (France) .She is now a Ph.D. student at "Ecole Centrale de Lyon" (France) and her research interests include structural analysis and sizing of Mecatronics systems by bond graph approach. Her e-mail address is: `mariem.el-feki@ec-lyon.fr`

MICHAEL DI LORETO received the M.S. degree from Ecole Centrale de Nantes, Nantes, France, in 2003, in automatic and computer engineering. He received the Ph.D. degree in automatic control from Ecole Centrale de Nantes, Nantes, France, in 2006. From 2006, he joined the INSA Lyon and Laboratoire Ampere, Lyon, France, where he is actually associated professor. His research interests include linear and nonlinear control theory, infinite dimensional systems, with particular emphasis in structural analysis. His e-mail address is: `michael.di-loreto@insa-lyon.fr`

ERIC BIDEAUX obtained his Engineering degree in Mechanics in 1991 (Nantes, France) and his Ph.D. degree in Control and Industrial Automation in 1995 from Franche-comté University (France). He joined The National Institute of Applied Sciences (Lyon, France) and Ampere Laboratory in 1996. He is Professor since 2006 and his research interests include modelisation, fluid power, control and sizing of mecatronics systems. His e-mail address is: `eric.bideaux@insa-lyon.fr`

DANIEL THOMASSET, Engineer (1972) in Electrical Engineering (option Control), PhD (1978) and "Doctorat d'Etat français" (1987). He is currently Professor at the National Institute of Applied Sciences, Lyon, France. His research interests include control and sizing of Mecatronics systems especially Fluid Power systems. He obtain (2001), with R. Fotsu Ngwompo and S. Scavarda, the Donald Julius Price of the Institution of Mechanical Engineers, U. K., for their work on sizing described in the paper "Physical model-based inversion in control systems design using Bond Graph representation", Proceedings of ImechE, Part I: Journal of Systems and Control Engineering, vol 15, n°12, 2001. His e-mail address is: `daniel.thomasset@insa-lyon.fr`

WILFRID MARQUIS-FAVRE is Assistant Professor in the Laboratory Ampère since 1998. He obtained his Engineering degree in Mechanics in 1992, his Master of Research degree in Industrial Automation in 1994, and his Ph. D. degree in Industrial Automation in 1997 in the Laboratory of Industrial Automation at the National Institute of Applied Sciences of Lyon (France). His main areas of research are modelling, simulation, analysis and design of mechatronic systems; inversion and optimization in bond graph language for the system sizing. His e-mail address is: `wilfrid.marquis-favre@insa-lyon.fr`

SENSITIVITY ANALYSIS OF LINEAR INVERSE MODELS
TO PARAMETER UNCERTAINTIES

Geneviève Dauphin -Tanguy
Ecole Centrale de Lille
LAGIS UMR 8146
59651 Villeneuve d'Ascq Cedex
E-mail:genevieve.dauphin-tanguy@@ec-lille.fr

KEYWORDS
Bond graph, parametric uncertainties, inverse model, sensitivity

ABSTRACT

The paper deals with the problem of sensitivity analysis of linear inverse models to parameter uncertainties. A new procedure is proposed to calculate symbolically the sensitivity matrix of a MIMO model using the sensitivity bond graph, bicausality and the Mason's rule. An implementation on a mechanical model shows the interest of the method.

INTRODUCTION

The sensitivity study was initiated by (Tomovic and Vukobratovic, 1972) and (Franck, 1978). Sensitivity analysis allows the quantification of the influence on a model of a change in any of its components (Marshal 1999, Middleton 1990). It may be of high importance for designing and controlling systems (Hayward and Cruz-Hernandez, 1998). Some papers deal with such a problem by using a bond graph approach (Gawthrop 2000, Kam and Dauphin-Tanguy, 2001, Borutzky et al, 2003).

The input-output relation in the linear case is written using the transfer matrix $M(s, \theta_i (i=1...p))$ (it will be denoted by $F(s, \theta_i (i=1...p))$ in the SISO case) as $Y(s) = M(s, \theta_i (i=1...p))U(s)$ where y and u respectively denote the output and input vectors supposed of same dimension m, and $\theta_i (i=1...p)$ are parameters. When the transfer matrix is deduced from a bond graph, the $\theta_i (i=1...p)$ are parameters associated with bond graph elements R, C, I, TF, GY, with a physical insight. Some of them are known with a good precision, but some can be subject to uncertainties (the mass of a car for example depends on the number of passengers and the amount of luggage).

Input u being independent of parameter uncertainties in open loop, it is straightforwardly written

$$\frac{\partial Y(s)}{\partial \theta_i} = \frac{\partial M}{\partial \theta_i} U(s) \qquad (1)$$

$S_{\theta_i}^M = \frac{\partial M}{\partial \theta_i}$ defined in the frequency domain is the sensitivity matrix of $M(s, \theta_i (i=1...p))$ with respect to the parameter θ_i.

[Canabellas et al, 1995] and [Gawthrop 2000] defined the sensitivity bond graph, denoted by SBG, obtained by duplicating the initial bond graph (BG) and introducing the SBG of the uncertain element. The association BG+SBG can be drawn in a BG related software and simulated. For symbolic manipulation purpose, or for analysis with a non BG simulation software, a method based on the Mason's rule (Brown 1972) was proposed in (Kam and Dauphin-Tanguy, 2001) to derive directly in the SISO case $S_{\theta_i}^F$ from the BG+SBG.

The inverse model $U(s) = M^{-1}(s, \theta_i (i=1...p))Y(s)$ is used for the designing of control laws for disturbance rejection and trajectory tracking as shown in figure 1, where d and y_{SP} denote respectively the disturbance and the reference trajectory.
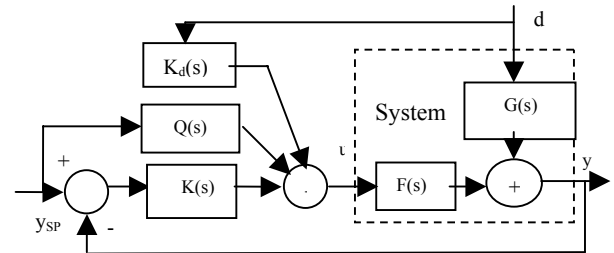
Figure 1: Closed loop control for disturbance rejection and trajectory tracking

In closed loop, equation (2) can be written as

$$Y(s) = (I + FK)^{-1}[(FQ + FK)Y_{SP}(s) + (G + FK_d)D(s)] \qquad (2)$$

Disturbance rejection and trajectory tracking are obtained in closed loop by setting

$$G + FK_d = 0 \text{ thus } K_d = -F^{-1}G$$

$$FQ + FK = I + FK \quad \text{thus } Q = F^{-1}$$

The performance of the control will depend on the validity of the model, and on the robustness of the control law with respect to parameter uncertainties. An other application of the inverse model is for the dimensioning problem in system design (Nwompo and Scavarda, 1999).

The study of the sensitivity of the inverse transfer matrix to parameter uncertainties is then a crucial problem. No paper in the literature at our knowledge until now deals with such a study.

The objective of the paper is to show how a BG model can help for deriving $S_{\theta_i}^{M^{-1}}$ under a symbolic form, by applying a graphical procedure based on the Mason's rule.

In the first section, we recall on a simple example how to derive the sensitivity function of the direct model from the sensitivity BG.

Then we propose a new procedure, based on the bicausality applied to the coupling BG+SBG to obtain the sensitivity function of the inverse model. It is implemented on a 2 input-2 output mechanical system, with two different sensor placements.

SENSITIVITY BOND GRAPH

Consider the DC motor given figure 2, and its bond graph model with a preferred derivative causality (BGD).
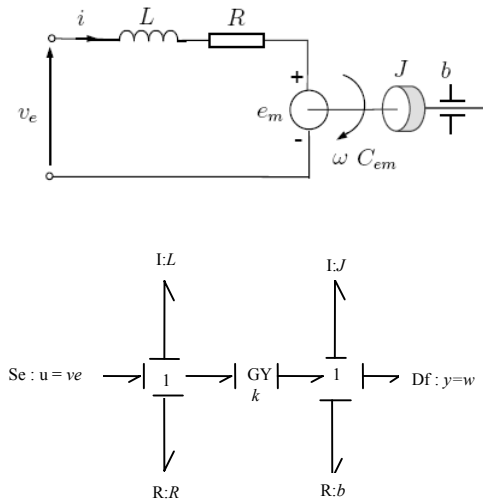
Figure 2 DC motor and its BGD

The transfer function $Y(s)/U(s)$, derived from the BGD using the Mason's rule, is

$$F(s) = \frac{N(s)}{D(s)} = \frac{1/k}{1 + \frac{1}{k^2} Rb + \frac{1}{k^2} RJs + \frac{1}{k^2} bLs + \frac{1}{k^2} LJs^2} \quad (3)$$

with

$$D(s) = 1 - \sum_i \underbrace{B_i}_{\substack{\text{causal loop} \\ \text{gains}}} + \sum_{i \neq j} \underbrace{B_i B_j}_{\substack{\text{disjoint causal} \\ \text{loop gains 2 by 2}}} -$$

$$- \sum_{i \neq j \neq k} \underbrace{B_i B_j B_k}_{\substack{\text{disjoint causal} \\ \text{loop gains 3 by 3}}} + ...$$

$$N(s) = \sum_i \underbrace{T_i(s)}_{\substack{\text{ith I/O causal} \\ \text{path gain}}} \underbrace{D_i(s)}_{\substack{\text{D without elements} \\ \text{included in Ti}}} \quad (4)$$

The sensitivity BG, denoted by SBG, has the same structure as the initial one, but all its variables are partial derivatives with respect to the uncertain parameter θ_i. The coupling between the BGD and the corresponding sensitivity SBGD is done by means of a detector (De or Df), a modulated source MSe or MSf and a block (or two for TF and GY-elements) corresponding to the partial derivative versus the uncertain parameter θ_i of the characteristic law $g(\theta_i)$ of the uncertain element. Figure 3 shows the BGD+SBGD structure in the general case.
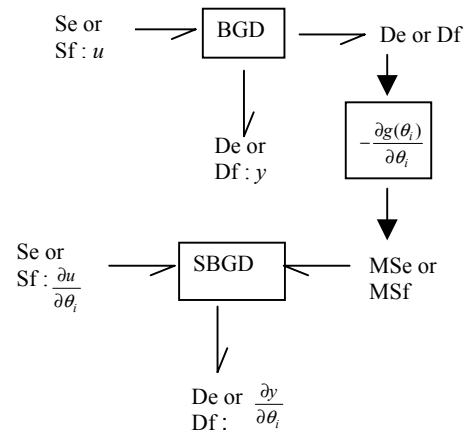
Figure 3 Structure of the BGD+SBGD

Suppose for example that the parameter b is uncertain in the DC motor model. The constitutive relationship for the R-element in resistance causality is $e_R = g(R)f_R$ with $g(R) = b$, which leads to

$$\underbrace{\frac{\partial e_R}{\partial b}}_{\text{effort}} = \frac{\partial g(R)}{\partial b} f_R + g(R) \underbrace{\frac{\partial f_R}{\partial b}}_{\text{flow}} \quad \text{with} \quad \frac{\partial g(R)}{\partial b} = 1 .$$

The corresponding BGD+SBGD is given in figure 4.
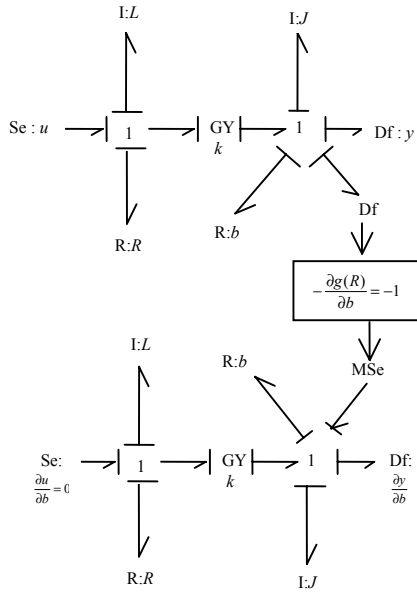
Figure 4 BGD+SBGD of DC motor with b uncertain

The determination of $S_b^F = \dfrac{\partial F}{\partial b} = \dfrac{\partial Y(s)/\partial b}{U(s)}$ using the Mason's rule on BGD+SBGD leads to

$$S_b^F = \frac{\partial F}{\partial b} = \frac{(-1)^3 \dfrac{1}{k^3}(Ls + R)}{\left(1 + \dfrac{1}{k^2}Rb + \dfrac{1}{k^2}RJs + \dfrac{1}{k^2}bLs + \dfrac{1}{k^2}LJs^2\right)^2} \quad (5)$$

The causal loops are the same in the BGD and in the SBGD, they are disjoint, which involves the expected square power at the denominator .

The (BGD+SBGD) can be simulated directly using a BG related software or with equation (5) using a mathematical simulation software as Matlab-Simulink. Using 20-Sim software, the Bode plot of S_b^F drawn with the numerical values of Table 1 and given in figure 5 shows that the sensitivity of F to parameter b uncertainty is decreasing for frequency higher than 10 Hz excitations.

Table 1: DC Motor Parameters

	Designation	Values
R	Armature resistance	2.4 Ω
L	Armature inductance	40 10^{-3} H
J	Moment of Inertia	0.84 10^{-3} kg.m^2
b	Friction coefficient	0.001 Nm/s
k	Torque coefficient	0.139 Nm/A

From equation (5), it comes that for low frequencies, $S_b^F \cong -\dfrac{R/k^3}{\left(1 + Rb/k^2\right)^2}$ depending on b value, and for

high frequencies, $S_b^F \cong 0$, which corresponds to the asymptotic behaviour of the S_b^F Bode plot.
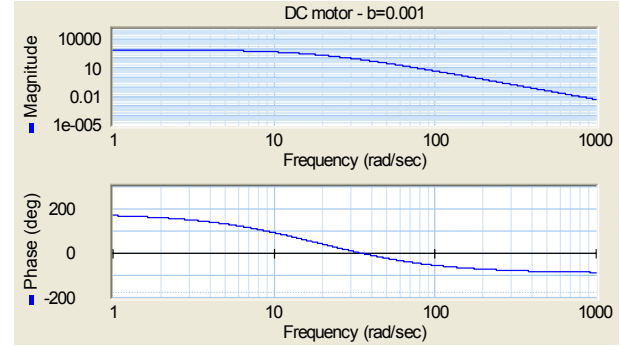
Figure 5 Bode plot of S_b^F for the DC motor

INVERSE MODEL

Let us denote by BG* the BG of the DC motor with a bicausality assignment (Gawthrop, 1995) as shown in figure 6. In the SISO case, the model is always invertible. In the MIMO case, (Nwompo et al, 1997) defined criteria, based on graphical procedures, to test the invertibility of the BG.
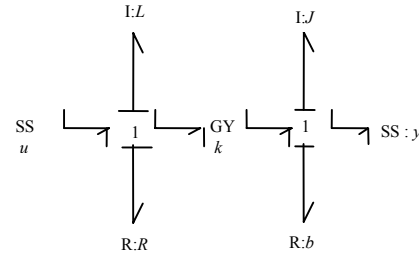
Figure 6 BG* bicausal BG of the DC motor

Using the Mason's rule on the BG* leads to the transfer function

$$\frac{U(s)}{Y(s)} = F^*(s) = F^{-1}(s) = \frac{k + 1/k\,(b + Js)(R + Ls)}{1} \quad (6)$$

INVERSE MODEL SENSITIVITY BOND GRAPH

Writing the partial derivative of the inverse model $U(s) = M^{-1}(s)Y(s)$ with respect to parameter θ_i gives

$$\frac{\partial U(s)}{\partial \theta_i} = \frac{\partial M^{-1}(s)}{\partial \theta_i}Y(s) + M^{-1}(s)\frac{\partial Y(s)}{\partial \theta_i} \quad (7)$$

$M^{-1}(s)$ is obtained from the BG* as shown previously.

To get $\dfrac{\partial M^{-1}(s)}{\partial \theta_i}$, we propose the following procedure using the SBG and bicausality.

Procedure:

- *in the MIMO case, test the inversibility of the model by determining the m I/O disjoint causal paths*
- *if it is invertible, draw the BG+SBG as shown previously*
- *assign the bicausality on the disjoint I/O causal paths on the BG part and on the SBG part separatively, and the classical (preferred derivative) causality elsewhere. It leads to the BG*+SBG* as shown in figure 7*
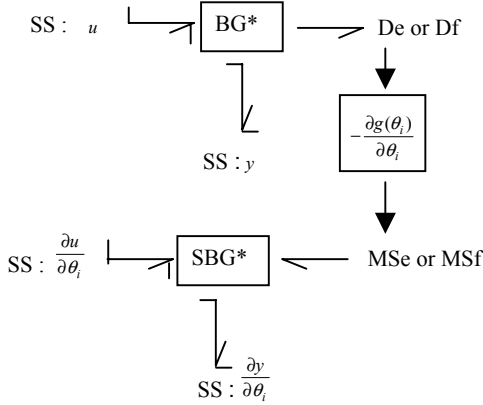
Figure 7 Structure of BG*+SBG*

- *calculate the different entries* $[S_b^{M^{-1}}]_{jk}$ *of* $S_b^{M^{-1}}$ *by applying the Mason's rule, using the causal loop gains and the gains of the causal paths from* y_k, *k=1...m, in the BG* to* $\dfrac{\partial u_j}{\partial \theta_i}$, *j=1...m, in the SBG*.*

Elements of justification:
Let us consider, for a sake of clarity, the SISO case. The transfer function is $F(s) = N(s)/D(s)$, its inverse $F^{-1}(s) = D(s)/N(s)$ will be denoted by $F^*(s) = N^*(s)/D^*(s)$ when calculated by means of the bicausal BG*. Calculating mathematically the sensitivity function of the inverse model

$$\frac{\partial F^{-1}(s)}{\partial \theta_i} = \frac{1}{N^2(s)}[N\frac{\partial D(s)}{\partial \theta_i} - D\frac{\partial N(s)}{\partial \theta_i}] \text{ is equivalent}$$

to determining

$$\frac{\partial F^*(s)}{\partial \theta_i} = \frac{1}{D^{*2}(s)}[D^*(s)\frac{\partial N^*(s)}{\partial \theta_i} - N^*(s)\frac{\partial D^*(s)}{\partial \theta_i}] \quad (8)$$

Let us denote by $B_i^*(\theta_i) = a_i g(\theta_i)$ the causal loops involving the element with uncertain parameter θ_i and B_j^* the others. Thus the denominator of F^* has the general form

$$D^*(\theta_i, s) = 1 - \sum B_j^* - \sum B_i^*(\theta_i) + \underbrace{\sum B_i^*(\theta_i)B_j^*}_{\substack{\text{disjoint causal} \\ \text{loop gains 2 by 2}}} +$$

$$+ \underbrace{\sum B_j^* B_k^*}_{\substack{\text{disjoint causal} \\ \text{loop gains 2 by 2}}} - .... \quad (9)$$

which gives

$$\frac{\partial D^*(\theta_i, s)}{\partial \theta_i} = -\sum \frac{\partial g(\theta_i)}{\partial \theta_i} a_i + \sum \frac{\partial g(\theta_i)}{\partial \theta_i} a_i B_j^* - .... \quad (10)$$

or $\dfrac{\partial D^*(\theta_i, s)}{\partial \theta_i} = \dfrac{\partial g(\theta_i)}{\partial \theta_i}[-\sum a_i + \sum a_i B_j^* - ....] \quad (11)$

The numerator $N^*(s)$ can be composed of only three different types of terms depending whether the uncertain parameter belongs to the I/O causal path or to a causal loop:

$$N^*(\theta_i, s) = T_1^*(s)D_1^*(s) + T_2^*(\theta_i, s)D_2^*(s) + T_3^*(s)D_3^*(\theta_i, s)$$

A term as $T_4^*(\theta_i, s)D_4^*(\theta_i, s)$ cannot exist because the uncertain element with parameter θ_i in the forward path cannot be in the loops considered in $D_4^*(\theta_i, s)$ as expressed in equation (4). It gives

$$\frac{\partial N^*(\theta_i, s)}{\partial \theta_i} = \frac{\partial g(\theta_i)}{\partial \theta_i} a_i \overline{T}_2^*(s)D_2^*(s) +$$

$$+ T_3^*(s)[-\sum \frac{\partial g(\theta_i)}{\partial \theta_i} a_i + \sum \frac{\partial g(\theta_i)}{\partial \theta_i} a_i B_j^* - ....]_3 \quad (12)$$

(the subscript 3 after the square bracket denotes that the terms inside the brackets are derived from the loops considered in $D_3^*(\theta_i, s)$)

or $\dfrac{\partial N^*(\theta_i, s)}{\partial \theta_i} = \dfrac{\partial g(\theta_i)}{\partial \theta_i}[a_i \overline{T}_2^*(s)D_2^*(s) +$

$$+ T_3^*(s)[-\sum a_i + \sum a_i B_j^* - ....]_3] \quad (13)$$

Regrouping the different terms in $\dfrac{\partial F^*(s)}{\partial \theta_i}$ leads to

$$\frac{\partial F^*(s)}{\partial \theta_i} = \frac{1}{D^{*2}(s)}\frac{\partial g(\theta_i)}{\partial \theta_i}\sum \hat{T}_i^*(s)\hat{D}_i^*(s) \quad \text{which can}$$

be interpreted in terms of causal path and causal loop gains in BG*+SBG*. The term $\dfrac{\partial g(\theta_i)}{\partial \theta_i}$ connects the two parts and appears in the gain of the causal path from y to $\dfrac{\partial u}{\partial \theta_i}$. The causal loops are the same in BG* and SBG*, which justifies the square term in the denominator. The term $\sum \hat{T}_i^*(s)\hat{D}_i^*(s)$ is always independent of θ_i.

To implement the procedure, consider the mechanical system given in figure 8 (a) and 10 (a), with two different placements of sensors. The uncertain element is the damper parameter *b*.

First case: *the uncertain element belongs to the u_2-y_2 I/O causal path in the BGI (figure 8(b))*

(a)

(b)

(c)
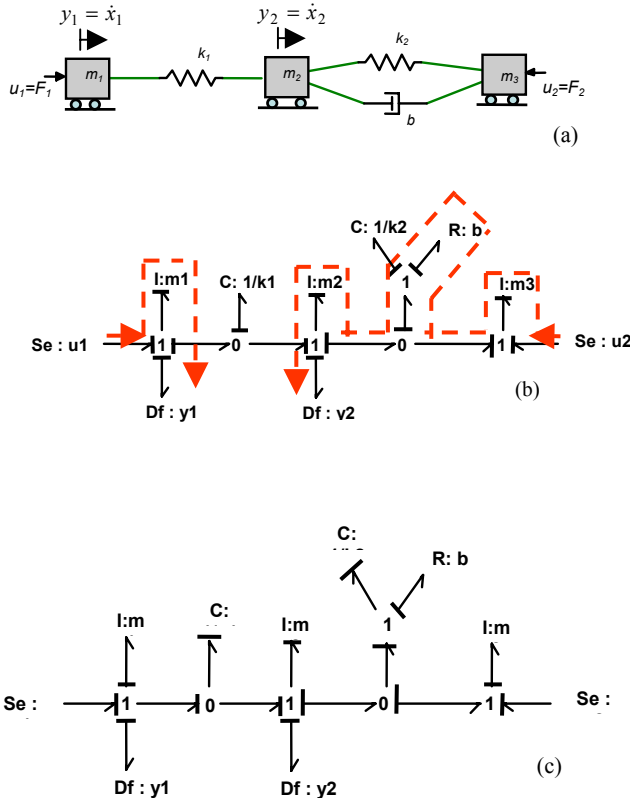
Figure 8 (a) Physical system- (b) its BGI- (c) its BGD

The model of the system given in figure 8 has the following structural properties:
- its order is 5 (from its BGI), and the rank of the state matrix is 4 (from the BGD where I:m_2 remains in integral causality), which corresponds to a null pole in the denominator
- it is structurally observable by the detectors, because the dualisation of one of the Dfs in the BGD allows the change of integral to derivative causality for I: m_2 (Sueur and Dauphin-Tanguy 1989)
- it is structurally invertible (from the BGI) because there is a unique choice of two disjoint I/O causal paths.

From figure 8(c), using the Mason's rule leads to:

$$\underbrace{\begin{pmatrix} Y_1(s) \\ Y_2(s) \end{pmatrix}}_{Y_{12}(s)} = \underbrace{\frac{1}{D(s)}\begin{bmatrix} N_{11} & N_{12} \\ N_{21} & N_{22} \end{bmatrix}}_{M_1(s)} \underbrace{\begin{pmatrix} U_1(s) \\ U_2(s) \end{pmatrix}}_{U(s)} \quad (14)$$

with

$$D(s) = s[\frac{m_1 m_3}{k_1 k_2}s^4 + (\frac{m_1 b}{k_1 k_2} + \frac{m_1 m_3 b}{k_1 m_2 k_2})s^3 + (\frac{m_1}{k_1} + \frac{m_3}{k_2} + \frac{m_1 m_3}{k_1 m_2})s^2$$
$$+ (\frac{b}{k_2} + \frac{bm_1}{k_2 m_2} + \frac{bm_3}{k_2 m_2})s + (\frac{m_1}{m_2} + \frac{m_3}{m_2} + 1)]$$

$$N_{11}(s) = \frac{s^2}{k_1}[\frac{m_3}{k_2}s^2 + (\frac{b}{k_2} + \frac{m_3 b}{m_2 k_2})s + \frac{m_3}{m_2} + 1] +$$
$$+ \frac{1}{m_2}[\frac{m_3}{k_2}s^2 + \frac{b}{k_2}s + 1]$$

$$N_{12}(s) = \frac{1}{m_2}[\frac{b}{k_2}s + 1], \quad N_{21}(s) = \frac{1}{m_2}[\frac{m_3}{k_2}s^2 + \frac{b}{k_2}s + 1]$$

$$N_{22}(s) = \frac{1}{m_2}[\frac{m_1 b}{k_1 k_2}s^3 + \frac{m_1}{k_1}s^2 + \frac{b}{k_2}s + 1]$$

Applying the proposed procedure gives BG*+SBG* as shown in figure 9. It can be seen that the bicausality assignment affects only the disjoint I/O causal paths in BG* and in SBG*. The other bonds are in classical causality with a preferred derivative one for the I and C-elements, including I:m_2 which had to remain in integral causality in the BGD, but except for C:$1/k_1$.
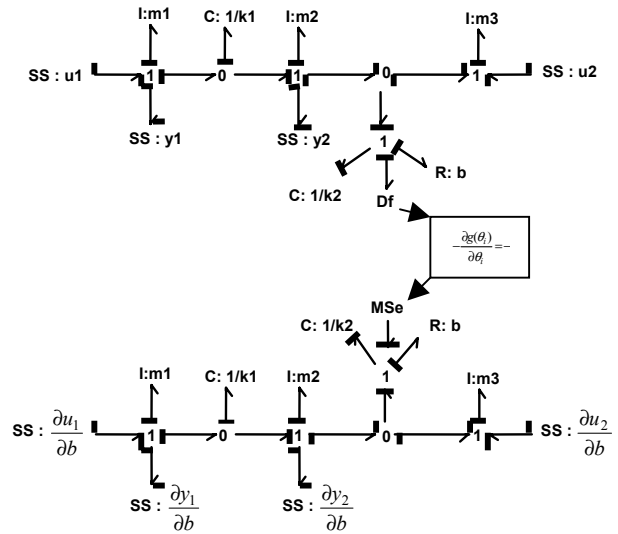
Figure 9 BG*+SBG* of system Fig 8

The inverse model can be deduced from the BG* figure 9 (upper part) as

$$\underbrace{\begin{pmatrix} U_1(s) \\ U_2(s) \end{pmatrix}}_{U(s)} = \underbrace{\frac{1}{D^*(s)}\begin{bmatrix} N_{11}^* & N_{12}^* \\ N_{21}^* & N_{22}^* \end{bmatrix}}_{M_1^{-1} = M_1^*} \underbrace{\begin{pmatrix} Y_1(s) \\ Y_2(s) \end{pmatrix}}_{Y_{12}(s)} \quad (15)$$

with $D^*(s) = 1 + \frac{b}{k_2}s$ and

$$N_{11}^*(s) = m_1 s(1 + \frac{b}{k_2}s), \quad N_{12}^*(s) = -\frac{k_1}{s}(1 + \frac{b}{k_2}s)$$

$$N_{21}^*(s) = \frac{k_1}{s}[-(1+\frac{b}{k_2}s)+\frac{m_3}{k_2}s^2],$$

$$N_{22}^*(s) = m_2 s(1+\frac{b}{k_2}s)+\frac{m_3}{k_2}s^2$$

The sensitivity matrix $S_b^{M_1^{-1}} = \frac{\partial M_1^{-1}(s)}{\partial b}$ is obtained from the BG*+SBG* by applying the Mason's rule, using the causal paths from y_k to $\frac{\partial u_j}{\partial b}$, j,k=1,2. It gives

$$S_b^{M_1^{-1}} = \begin{bmatrix} [S_b^{M_1^{-1}}]_{11} & [S_b^{M_1^{-1}}]_{12} \\ [S_b^{M_1^{-1}}]_{21} & [S_b^{M_1^{-1}}]_{22} \end{bmatrix} \qquad (16)$$

such that

$$S_b^{M_1^{-1}} = \frac{1}{(1+\frac{b}{k_2}s)^2}\begin{bmatrix} 0 & 0 \\ -\frac{k_1 m_3}{k_2^2}s^2 & -\frac{m_3}{k_2^2}s^3 \end{bmatrix} \qquad (17)$$

Physical interpretation:
- Two entries are null because there is no causal path from y_1 and y_2 to $\partial u_1/\partial b$. It can be concluded that the two terms $[S_b^{M_1^{-1}}]_{11}$ and $[S_b^{M_1^{-1}}]_{12}$ are not parameter b-sensitive.
- The two other entries correspond to the gain of the causal paths from y_1 and y_2 to $\partial u_2/\partial b$ respectively. There is no causal loop disjoint of the causal paths. It can be concluded that $[S_b^{M_1^{-1}}]_{21} \cong 0$ and $[S_b^{M_1^{-1}}]_{22} \cong 0$ for low frequencies and $[S_b^{M_1^{-1}}]_{21} \cong -\frac{k_1 m_3}{b^2}$ and $[S_b^{M_1^{-1}}]_{22} \to -\infty$ for very high frequencies.

Second case: *the uncertain element does not belong to any input-output causal path in the BGI (figure 10(b))*

Consider the same system as previously, but the flow detector is now on the 1-junction connected to mass m_3, as shown figure 10(a). From figure 10(b) and 10(c), it is concluded that the model is invertible and observable by one Df.

(a)

(b)

(c)
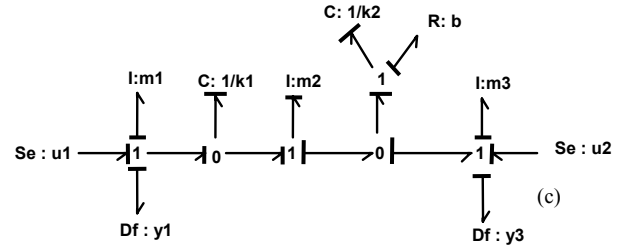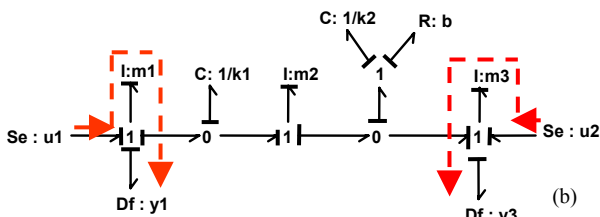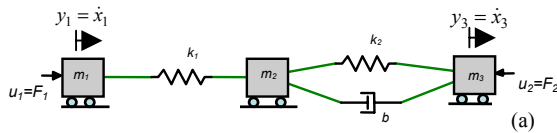
Figure 10 a) Physical system b) its BGI c) its BGD

The BG*+SBG* is given figure 11.
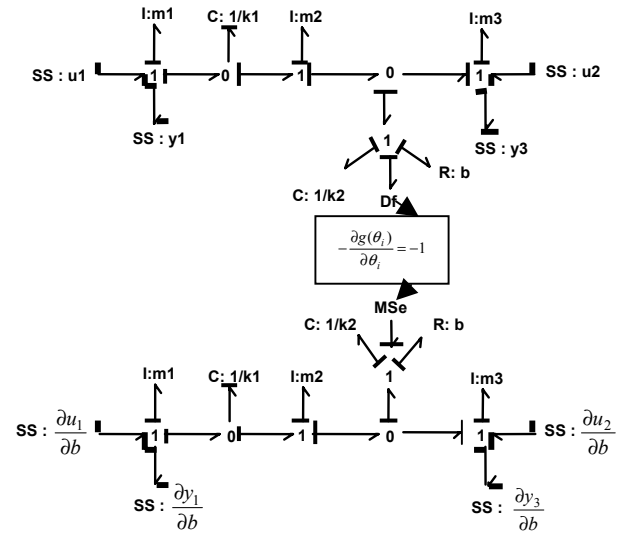
Figure 11 BG*+SBG* of figure 10

By using the proposed procedure on BG*+SBG* in figure 11, it gives:

$$S_b^{M_2^{-1}} = \frac{\partial}{\partial b}M_2^{-1} = \begin{bmatrix} [S_b^{M_2^{-1}}]_{11} & [S_b^{M_2^{-1}}]_{12} \\ [S_b^{M_2^{-1}}]_{21} & [S_b^{M_2^{-1}}]_{22} \end{bmatrix} \qquad (18)$$

such that

$$S_b^{M_2^{-1}} = \frac{1}{\Delta_2(s)}\begin{bmatrix} 1 & -(\frac{m_2}{k_1}s^2+1) \\ -(\frac{m_2}{k_1}s^2+1) & (\frac{m_2}{k_1}s^2+1)^2 \end{bmatrix} \qquad (19)$$

with $\Delta_2(s) = (\frac{m_2}{k_1}s^2 + \frac{b}{k_1}s + \frac{k_2}{k_1}+1)^2$

Physical interpretation:
From equation (19), it comes that $[S_b^{M_2^{-1}}]_{jk} \cong \frac{\pm 1}{1+k_2/k_1}$, j,k=1,2, for very low frequencies. $[S_b^{M_2^{-1}}]_{11} \cong 0$ for high frequencies,

$[S_b^{M_2^{-1}}]_{12} \cong 0$, $[S_b^{M_2^{-1}}]_{21} \cong 0$ and $[S_b^{M_2^{-1}}]_{22} \cong 1$ for very high frequencies.

All entries are independent of the uncertain parameter b.

CONCLUSION

The symbolic calculation of the sensitivity function of a transfer matrix is a heavy and unpleasant task, when done by hand, even for quite simple models. The complexity increases a lot when the aim is to study the sensitivity function of the inverse model, which involves matrix inversion. The calculation by means of symbolic software as Maple or Mathematica can be also long and even unsuccessful.

The bond graph, through its causal and graphical properties, may help the user to derive systematically and directly the desired sensitivity function. There is no need to firstly calculate the direct model, and to inverte it. Only graphical procedures are used.

The proposed approach was implemented on a 3 DF mass spring damper system of 5th order. For medium or large size BG models, it will need the use of software able to assign bicausality and to show up causal paths and causal loops, such as MS1 developed by F. Lorenz.

The interpretation of the entries of the sensitivity matrix in terms of causal path and causal loop gains can be an helpful mean for sensor placement analysis with the objective of finding the less sensitive to a parameter uncertainty sensor positions according to the frequency domain of interest. It is the direction we will take for future work.

REFERENCES

Borutzky W., Dauphin-Tanguy G. and Kam C., 2003 "Relation between two bond graphs approaches to sensitivity analysis and to robustness study", *Mathmod*, February 2003, Vienna., Austria.

Brown F.T. 1972, "Direct application of the loop rule to bond graph", *Journal of Dynamic systems, Measurement and Control*, pp 253-261, September 1972.

Canabellas J.M, J. Felez, and C. Vera, 1995 "A Formulation of the Sensitivity Analysis for Dynamic Systems Optimization Based on Pseudo Bond Graphs", *Proceedings of International Conference on Bond Graph Modeling and Simulation* (ICBGM'95) (F.E.Cellier and J.J.Granda, eds.), Las Vegas, SCS Publishing, Simulation Series, Vol.27, No.1, pp.135-144.

Franck P.M. 1978, "Introduction to system sensitivity theory", Academic Press. New York.

Gawthrop P. 1995 "Bicausal bond graphs", *Proceedings of the International Conference on Bond Graph Modelling,* ,

Las Vegas, SCS Publishing, Simulation Series, Vol.27, No.1, pp 207-213

Gawthrop P. 2000, "Estimation of mechatronic systems using sensitivity bond graphs", *IFAC 2000,* Darmstadt, Germany, pp 893-897.

Hayward V. and Cruz-Fernandez J.M. 1998 "Parameter sensitivity analysis for design and control of force transmission systems", *ASME J. of Dynamic Systems, Measurements. and Control*, 120(2), pp 241-249

Kam C. and Dauphin-Tanguy G., 2001 "Sensitivity function determination on a bond graph model". *WMC SCS, ESS'01*, October 2001, Marseille, France, pages 735-739.

Marshal, Harold E. 1999, "Sensitivity Analysis", Chapter 8.12, *The Technology management Handbook*, Editor-in-Chief, Richard C. Dorf (Boca Raton, FL: CRC Press, Inc.), pp.8/59-63.

Middleton R.H. and Goodwin G.C. 1990, "Digital control and estimation: a unified approach", by *Prentice-Hall*, Inc. A division of Simon & Schuster Englewood Cliffs, New Jersey 07632, 1990

Ngwompo R.F, Scavarda S., Thomasset D., 1997 "Structural invertibility and minimal inversion of multivariable linear systems- A bond graph approach", *Proceedings of the International Conference on Bond Graph Modelling,* Phoenix (Arizona), pp 109-114

Ngwompo R.F and Scavarda S., 1999 "Dimensioning problems in system design using bicausal bond graphs", *Simulation Practice and Theory,* vol7,issues 5-6 pp 577-587

Sueur C. and Dauphin-Tanguy G., 1989 "Structural controllability/observability of linear systems represented by bond graph", *Journal of Franklin Institute,* vol 326, issue 6, pp 869-883

Tomovic R., Vukobratovic M. 1972, "General sensitivity theory" *Modern Analytic and computational methods in science and mathematics* number 35, Elsevier. New York
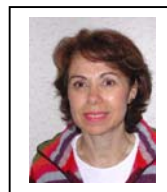
GENEVIEVE DAUPHIN-TANGUY (senior member IEEE), graduated from University of Lille, France (BSc Physics, BSc Mathematics, MSc Mechanics) and from the french Grande Ecole d'Ingénieurs, Ecole Centrale de Lille (State Engineering Diploma), received the Ph.D. and the Doctorate of Sciences both from University of Lille. She is presently Professor of Control Design at Ecole Centrale de Lille and Head of the research group "Bond graph models" in the Laboratoire d'Automatique Génie Informatique et Signal (LAGIS associated with the CNRS). She is expert for the French Ministry of Education and Research. She is mainly engaged in the research fields of Modelling, Analysis, Control and Monitoring of systems by using the bond-graph tool. Her application domains are mechatronics in car industry, thermofluid processes and power systems.

EFFICIENT SIMULATION OF HYBRID SYSTEMS: AN APPLICATION TO ELECTRICAL POWER DISTRIBUTION SYSTEMS

Indranil Roychoudhury, Matthew Daigle, Gautam Biswas, and Xenofon Koutsoukos
Institute for Software Integrated Systems
Department of Electrical Engineering and Computer Science
Vanderbilt University, Nashville, TN 37235, USA
Email: {indranil.roychoudhury, matthew.j.daigle, gautam.biswas, xenofon.koutsoukos}@vanderbilt.edu

KEYWORDS

Hybrid system simulation, hybrid bond graphs, component-based modeling, block diagrams

ABSTRACT

This paper presents an efficient simulation scheme for hybrid systems modeled as hybrid bond graphs (HBGs). Considerable computational savings are achieved when mode changes occur during simulation by identifying persistent causal assignments to bonds, and, consequently, fixed causal structures at HBG junctions when the simulation model is derived. Persistent causal assignments also reduce the possible computational structures across all mode changes, and this leads to an overall reduction in the complexity of the simulation models. We demonstrate the benefits of our approach for an electrical power distribution system that includes a fast switching inverter system.

INTRODUCTION

Accurate and efficient modeling and simulation approaches are essential for design, analysis, diagnosis, and prognosis of complex, embedded systems. To address these needs, we have developed component-oriented modeling techniques based on hybrid bond graphs (Manders et al., 2006), and a model-integrated design methodology for efficient simulation that facilitates diagnosis and prognosis experiments (Roychoudhury et al., 2007; Poll et al., 2007). The building of accurate and efficient simulation models for hybrid, nonlinear systems is not trivial, especially since the simulation must deal with the computational issues that arise from nonlinear behaviors, as well as, accommodate system reconfigurations that produce discrete behavior changes. For systems where the reconfigurations occur at high frequencies, such as modern electronics-based electrical power systems (Biel et al., 2004), it is especially important to maintain accuracy in the generated behaviors without sacrificing simulation efficiency.

Bond graphs (BGs) are well-suited for modeling electrical power systems. The BG modeling language allows for multi-domain, topological, lumped-parameter modeling of physical processes by capturing their energy exchange mechanisms (Karnopp et al., 2000). The continu-
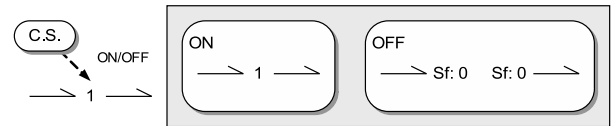
Figure 1: Semantics of a Switching 1-Junction

ous BG representation has been extended to model hybrid systems by several researchers (Buisson et al., 2002; Magos et al., 2003; van Dijk, 1994; Borutzky, 1995). Our approach, hybrid bond graphs (HBGs), introduces discrete mode changes into continuous BG models through idealized *switching junctions* (Mosterman and Biswas, 1998). A two state (*on* and *off*) finite state machine implements the junction *control specification* (CSPEC). Transitions between states may be functions of system variables and/or system inputs. When a switching junction is on, it behaves like a conventional junction. When off, all bonds incident on a 1-junction (0-junction) are deactivated by enforcing 0 flows (efforts) on all bonds incident on that junction (see Fig. 1). The system mode at any given time is determined by composing the modes of the individual switching junctions.

In earlier work, we produced an efficient simulation method for hybrid systems by constructing reconfigurable block diagram (BD) models from HBGs (Roychoudhury et al., 2007; Daigle et al., 2007). We adopted the BD formalism for simulating HBGs for three main reasons: *(i)* the BD formalism is a widely used computational scheme, *(ii)* the input-output formulation of each block in a BD can be determined using the causality information present in HBGs, and *(iii)* BD models preserve the component structure of the model, which facilitates introduction of faults into components for simulation-based diagnosis and prognosis studies. Our BD-based simulation models include switching elements that enable the online reconfiguration of the BD components to account for different causality assignments in different system modes. Every time a mode change occurs, causalities are incrementally reassigned from the previous mode, and the effort and flow links are rerouted by the switching elements to ensure that the computational model corresponds to the new causality assignment. This approach produces acceptable results when mode changes are infrequent. For fast switching systems
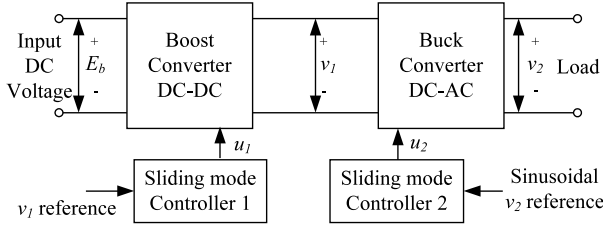
Figure 2: Block Diagram of a Boost-Buck AC Inverter

with frequent mode changes, e.g., electrical power conversion and distribution systems, invoking the procedure for reassigning causality at *every* mode change may produce unnecessary computational overhead, which leads to significant increases in the simulation execution times. Also, the BD models may include extraneous switching elements and signal connections, which account for causality assignments that never occur during the simulation.

This paper extends our earlier work by identifying bonds whose causal assignments persist across mode changes in the HBG model, and restrict possible reconfigurations that can occur in the simulation model when mode changes occur. Confining the effects of mode changes to small parts of the simulation model substantially reduces the computational effort required to execute mode changes during simulation. Further, persistent causality assignments limit the number of switches needed for each BD component, as well as, the number of possible signal connections. Therefore, simulation efficiency is increased, which produces significant gains for fast switching systems. We demonstrate the effectiveness of our approach by applying it to a power conversion and distribution testbed developed for diagnosis and prognosis studies at NASA Ames (Poll et al., 2007).

MOTIVATING EXAMPLE: AC INVERTER

The Advanced Diagnostics and Prognostics Testbed (ADAPT) models aircraft and spacecraft power distribution systems (Poll et al., 2007). It includes battery units for power storage, inverters for DC to AC power conversion, a power distribution network made up of a number of circuit breakers and relays, and DC and AC loads. In this paper, we focus on the AC subsystem, and develop the fast-switching inverter model to motivate our approach.

The inverter, a two-stage boost-buck DC-AC converter (Biel et al., 2004), consists of a cascade connection between a boost DC-DC converter with a full-bridge buck DC-AC converter to achieve a transformerless DC-AC step-up conversion (see Fig. 2). The boost converter boosts the input DC voltage to a higher value (190 V, in our case), and the buck converter stage generates the sinusoidal AC voltage. The fast switching in the boost and buck converters are governed by two sliding mode controllers, one for each stage of the inverter.

The equivalent circuit model of the boost-buck DC-
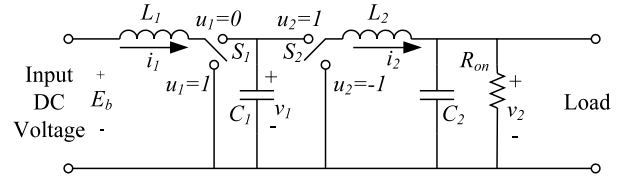
Figure 3: Circuit Model of a Boost-Buck AC Inverter
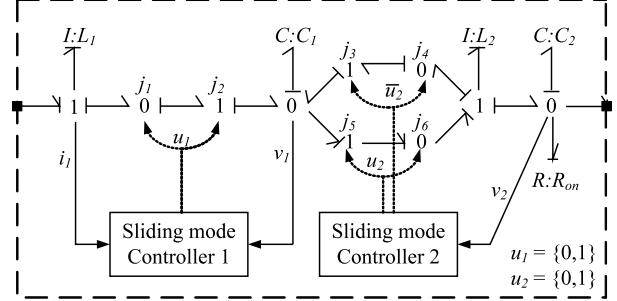
Figure 4: Inverter HBG Component Model

AC inverter is shown in Fig. 3, where S_1 is a conventional power switch, and S_2 corresponds to a full bridge switch. The control signals for S_1 and S_2 are u_1 and u_2, respectively. The differential equations for the system can be found in (Biel et al., 2004), and the model parameters are listed in Table 1. The internal resistance R_{on} accounts for the current that the inverter draws from the battery when it is disconnected from its loads.

The HBG model of the inverter, derived from its circuit model, is shown in Fig. 4. Switch S_1 is represented by the synchronous switching junctions j_1 and j_2, i.e., they share the same CSPEC function. Switch S_2, is represented by the switching junctions $j_3 - j_6$, with j_3 and j_5 having the same CSPEC as junctions j_4 and j_6, respectively. The switching conditions for junctions j_3 and j_4 are logical negations of those for junctions j_5 and j_6.

The sliding mode controllers are also modeled using HBGs. They generate signals that switch the inverter junctions at kilohertz rates. Our previous simulation algorithm invokes the `Hybrid SCAP` causal assignment procedure (Roychoudhury et al., 2007) at every mode change to compute the updated model configurations before the continuous simulation is resumed. However,

Table 1: Inverter Model Parameter Values

Inductances (H)	$L_1 = 0.0022, L_2 = 0.075$
Capacitances (F)	$C_1 = 0.0069, C_2 = 6 \times 10^{-6}$
Resitances (Ω)	$R_{on} = 489.49$
Sliding mode controller 1 parameters	$\alpha = 0.8, \beta = 4.3649$ $\delta = 111.375, K = 829.3347$
Sliding mode controller 2 parameters	$a_1 = 15.915,$ $a_2 = 0.0048$
Reference Voltages (V)	$v_{1Ref} = 190,$ $v_{2Ref} = 120\sqrt{2}\sin(120\pi)$
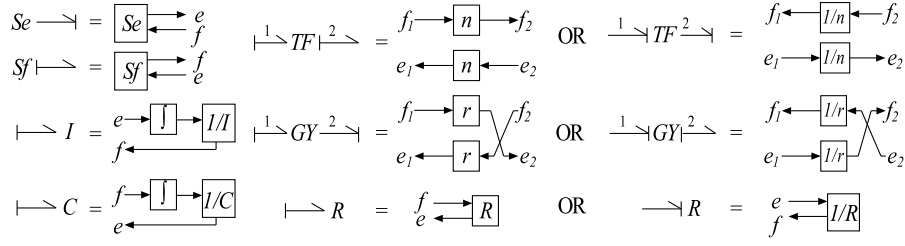
Figure 5: Computational Structures for 1- and 2-Port Bond Graph Elements

careful inspection of the inverter HBG model shows that the causality assignments at the switching junctions remain the same when the junctions are on. When the junctions are off, the causal changes do not propagate to adjacent junctions (see the next section for details). Therefore, the external calls to `Hybrid SCAP` at every mode change are not necessary, and considerably slows the simulation. If we can identify the causality assignments that do not change when reconfigurations occur, the number of calls to `Hybrid SCAP` can be reduced, and the model reconfiguration task can be simplified. This approach is likely to produce more efficient simulation models than our previous approach.

EFFICIENT SIMULATION OF HYBRID BOND GRAPH MODELS

Efficient simulation models for hybrid systems should meet two primary requirements: *(i)* avoid pre-enumeration of all system modes, especially for systems with a large number of modes, and *(ii)* minimize the amount of computations performed when mode changes occur. Reassignment of causality produces changes in the computational model. But, we can minimize the number of changes during reconfiguration by *(i)* recognizing causal assignments that persist across all modes, and *(ii)* not accounting for configurations that we can pre-determine will never occur. As a result, we can reduce the extent of causal propagation changes, and simplify the simulation models.

Converting Bond Graphs to Block Diagrams

In our work, we assume that energy storage elements (i.e., C and I) are in *integral causality*. Fig. 5 shows the BD structure for 1- and 2-port BG elements (Karnopp et al., 2000). The Sf, Se, and 1-port C and I elements, and the n-port I and C fields, each have a unique BD representation because their incident bonds have only one possible causal assignment. The BDs for the n-port I and C fields are simple functional and structural extensions to the 1-port I and C elements, respectively, and hence not shown in Fig. 5. The R, TF and GY elements allow two causal assignments each, and each assignment produces a different BD model.

Mapping a junction structure to its BD model is facilitated by the commonly used notion of the *determining*

bond, which captures the causal structure for the junction.

Definition 1 (Determining Bond) *The determining bond of a* 0- (1-) *junction is the bond that establishes the effort (flow) value of all other bonds incident on the junction.*

Fig. 6 shows the BD expansion for a 1- junction. For a 1-junction, all other bonds' flow values are equal to the determining bond's flow value, and the effort value of the determining bond is the algebraic sum of the effort values of the other bonds connected to this 1- junction. A junction with m incident bonds that does not switch can have m possible BD configurations.

There is a well-defined procedure for converting a causal BG structure to a BD model (Karnopp et al., 2000). First, each bond is replaced by two signals, i.e., the effort and flow variables for the bond. Next, each bond graph element is replaced by the computational structure corresponding to the assigned causality, and the signals connections are established to complete the model.

Converting Hybrid Bond Graphs to Block Diagrams

As is shown in Fig. 6, the reassignment of causality due to mode changes can alter a junction's determining bond, and, therefore, its underlying computational BD model. However, instead of rebuilding the entire BD model every time mode changes occur, we include switching elements in the BD components to reconfigure the computational model during simulation. For example, a switching junction with m incident bonds can switch between $m + 1$ possible computational configurations, m corresponding to each incident bond being a determining bond, and one corresponding to the junction being off, in which each outgoing signal is set to zero. Therefore, the physical connections between blocks are fixed, but the interpretation of the signal on the connection (effort or flow) changes depending on the causal assignment to the bonds.

In some cases, however, the causal assignment for a bond is invariant across *all* possible modes of system behavior, i.e., the causal assignment is *persistent*. For example, a C-element will always impose effort on a 1-junction through the connecting bond. In this case, the BD for the 1-junction does not need to include any
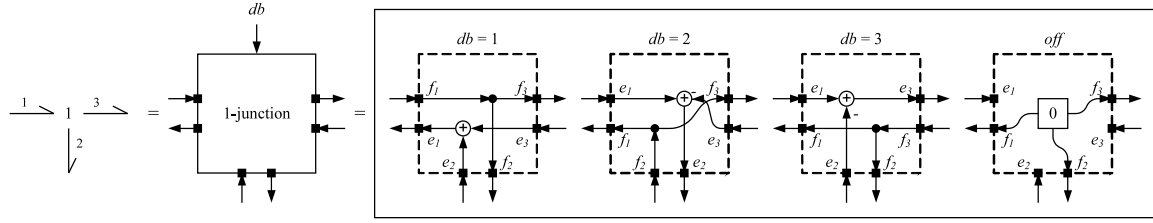
Figure 6: Block Diagram Expansion of a 1-Junction

switching mechanism to accommodate the possibility of this bond being its determining bond. If all bonds of a junction have persistent causal assignments, then its determining bond is fixed for all modes of the system in which it is on. In this case, the BD for the 0-junction does not need to include any switching mechanisms because it can assume a fixed structure when the junction is on. In previous work, we have termed a nonswitching junction with this property to have *fixed causality* (Roychoudhury et al., 2007). Switching junctions, by definition, change causality when they turn off, but this change may not affect adjacent junctions. Therefore, we extend our previous definition of fixed causality to also include switching junctions.

Definition 2 (Fixed Causality) *A junction that does not switch is in* fixed causality *if, for* all *modes of system operation, its determining bond does not change. A switching junction is in* fixed causality *if, for* all *modes in which the junction is on, its determining bond does not change, and for all modes where it is off, the inactivation of its incident bonds does not affect the determining bond of any of its adjacent junctions.*

Persistent causality of bonds, and fixed causality of junctions can be identified efficiently using a SCAP-like algorithm before we construct the BD model for simulation from the HBG. In this algorithm, the causality assignment is first performed at junctions connected to sources and energy storage elements, because the bonds connecting them to these junctions have persistent causality. A 0-(1-) junction is in fixed causality if it is connected to a Se (Sf) or a C (I) element. Otherwise, a junction is in fixed causality if *(i)* its determining bond connects it to a fixed causality junction (either directly, or through a TF or a GY element), or *(ii)* all incident bonds other than its determining bond are connected to fixed causality junctions. Once a junction is determined to be in fixed causality, we propagate this information to its adjacent junctions to check if they too are in fixed causality, or any of their bonds are in persistent causality.

Some additional analysis is required to determine whether a switching junction is in fixed causality. When the junction is on, all its bonds must have persistent causality. When the junction turns off the determining bond for its adjacent junctions should not change, i.e., none of its incident bonds can be a determining bond for its adjacent junctions. A special case occurs if

two switching junctions are adjacent and share the same CSPEC. The knowledge that they switch together can help determine if causal changes will propagate when they switch. When we visit a junction for the first time, all its adjacent junctions may not have been checked for fixed causality yet. Hence, the causality may need to be propagated from all adjacent junctions before it can be determined that the junction is in fixed causality. The computational complexity of our approach to identify persistent causality of bonds and fixed causality of junctions, is polynomial in the size of the HBG, as it is similar to the SCAP algorithm.

If a junction has incident bonds with persistent causality, or if the junction is in fixed causality, the computational model of the junction block can be reduced by eliminating switching elements and signal connections which account for causality assignments that will never occur during simulation. This is illustrated for a 3-port switching 1-junction in Fig. 6. If the junction is not in fixed causality, its implementation can, in general, switch between four possible configurations as mode changes occur. However, if the junction is in fixed causality, the BD representation for this junction has only one valid on configuration, in addition to the off configuration. If the junction is not in fixed causality, for example, if its bond 1 connected to a Se, its BD representation need not include a configuration with bond 1 as its determining bond. Switched junctions in fixed causality help minimize causality reassignment computations when mode changes occur. Therefore, when this 1-junction switches, we know exactly what the causality assignment at this junction is without having to call any external causality reassignment procedure.

When switching junctions not in fixed causality change modes, we have to make external calls to a causality reassignment procedure. In previous work, we have developed the Hybrid SCAP algorithm that reassigned causality incrementally, starting from the junction directly affected by the switching, and then propagating the changes only to those junctions whose causal assignments were affected by changes in the adjacent junctions (Daigle et al., 2007; Roychoudhury et al., 2007). In this work, we use the knowledge of junctions in fixed causality, and bonds with persistent causality, to reduce the search and propagation for the Hybrid SCAP algorithm. Causal changes are not propagated along bonds with persistent causality, or to junction in fixed causality.
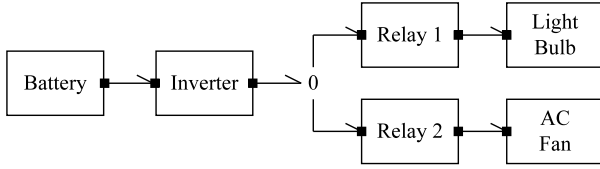
Figure 7: ADAPT Subsystem for Case Study

Consider the inverter HBG model (Fig. 4), where all nonswitching junctions are in fixed causality. The incident energy storage elements specify a unique determining bond for these junctions. All switching junctions are also fixed. Consider switching junctions j_1 and j_2. Since they always change modes simultaneously (because they share the same CSPEC), when on, j_1 always imposes flow on its adjacent junction j_2 which is also on. When they are off, the causality assignment of other active junctions are not affected. The case is similar for pairs j_3 and j_4, and j_5 and j_6. Since all junctions are in fixed causality, the mode switchings in the inverter do not require reassignment of causality, because the changes never propagate. Therefore, `Hybrid SCAP` is not invoked, and minimal changes have to be made to the computational model when mode changes occur, thus considerably speeding up the inverter simulation, as we illustrate later.

Simulating the Block Diagrams

Once a reconfigurable BD is generated for a new mode, using the extensions described above for a system model, the execution engine executes the BD in a continuous manner until the next mode change occurs. If the mode change is attributed to the switching of junctions that are not in fixed causality, the simulation is paused, the modified `Hybrid SCAP` algorithm is invoked to reassign causality, and the BD is reconfigured accordingly before the continuous simulation resumes. On the other hand, if a mode change is attributed to a switching junction with fixed causality, the BD reconfiguration is performed without invoking `Hybrid SCAP`.

CASE STUDY

We demonstrate our modeling and simulation framework using the ADAPT system at NASA Ames (Poll et al., 2007). In this paper, we focus on a subsystem of ADAPT shown in Fig. 7, i.e., a battery driving an inverter that is connected to two loads through two relays. One of these loads is a light bulb, while the other load is a large fan.

We build the system model using a component-based modeling approach (Manders et al., 2006; Poll et al., 2007). We use the HBG model of the inverter shown in Fig. 4 for our experiments. The HBG model of the lead-acid battery is based on an electrical equivalent circuit model, which captures the nonlinear battery behavior (Daigle et al., 2007). We omit a detailed description of the battery model due to space constraints. The re-
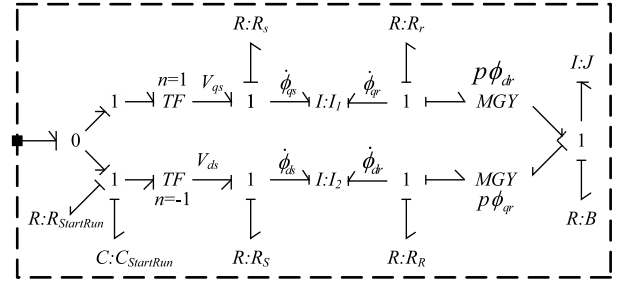
Figure 8: AC Fan Hybrid Bond Graph

Table 2: AC Fan Model Parameter Values

Inductances (in H)	$L_{ss} = 0.275$, $L_{SS} = 0.274$, $L'_{rr} = 0.272$, $L'_{RR} = 0.271$, $L_{ms} = 0.1772$, $L_{mS} = 0.2467$
Inertia (in kg m^2)	$J = 6.5 \times 10^{-4}$
Resistances (in Ω)	$R_s = 163.02$, $R_S = 168.14$, $R'_r = 145.12$, $R'_R = 145.12$, $R_{startRun} = 26$,
Friction (in kg m^2/s)	$B = 4.734 \times 10^{-4}$
Capacitances (in μF)	$C_{startRun} = 21.1$
Other Parameters	$N_S/N_s = 1.18$, $p = 2$

sistive light bulb load consists of a single power port, a 1-junction, and a R element. The R element parameter value is a constant $234.6966\ \Omega$.

The AC fan is modeled as a single phase, fixed capacitor induction motor (Thaler and Wilcox, 1966). We represent the system using the standard $d - q$ model, described in (Krause, 1986). We do not present the fan model equations due to space constraints. The model parameters are presented in Table 2. The HBG model of the fan, shown in Fig. 8, is adapted from that described in (Karnopp, 2003). Note that the AC fan has two I-fields with parameters,

$$I_1 = \left[\begin{array}{cc} L_{ss} & L_{ms} \\ L_{ms} & L'_{rr} \end{array} \right], \text{ and } I_2 = \left[\begin{array}{cc} L_{SS} & L_{ms} \\ L_{ms} & L'_{RR} \end{array} \right].$$

Experimental Results

For the experimental results, we automatically derived a Matlab$^®$ Simulink$^®$ model of the subsystem shown in Fig. 7, from the HBG model of the system using the implementation described in (Roychoudhury et al., 2007) along with the extensions presented in this paper. All experiments were performed on a 2.4 GHz Intel$^®$ Pentium Core^{TM}2 Duo CPU desktop, with 2 GB RAM. The model was simulated using a fixed-step simulation with a sample period of 7.5 μs.

Fig. 9 shows the results of the simulation. We plot the voltages and currents at the output of the battery and the inverter, as well as the rotational speed of the AC fan. The simulation model was run for 20 seconds of simulation time. In the first configuration, the light bulb is connected to the inverter from $2 - 5$ seconds, followed
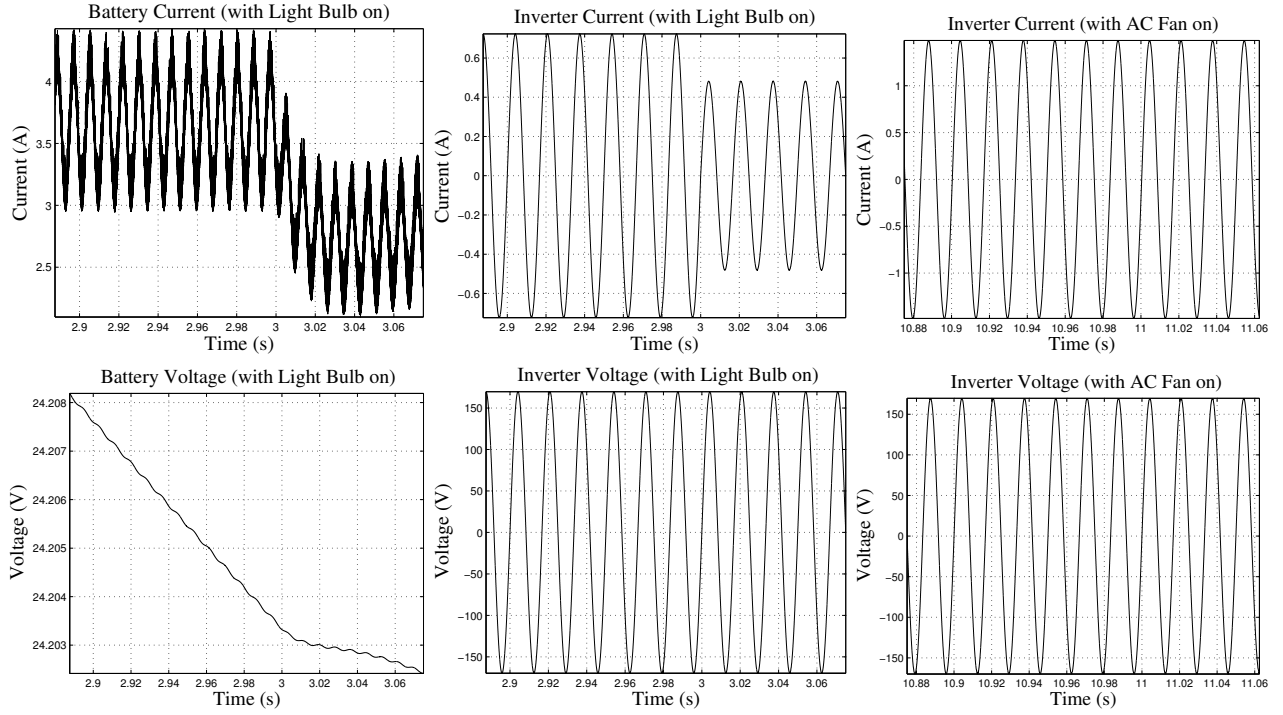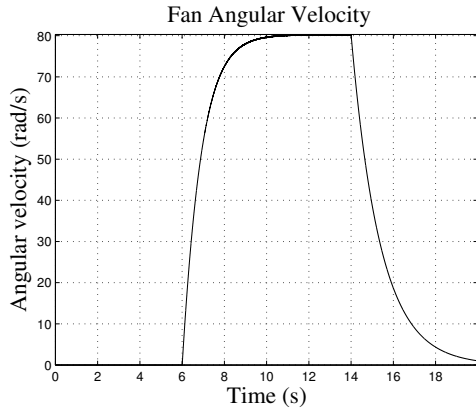
Figure 9: Simulation Results

Figure 10: Angular Velocity of AC Fan

Table 3: Duration (in s) for Simulating Model for 1s

SCAP called at every mode change	Hybrid SCAP called at every mode change	No causality reassignment procedure called
6054.3	6025.2	58.3

by a second configuration where the AC fan is turned on between $7 - 15$ seconds. An abrupt fault is injected in the light bulb resistance at 3 seconds to demonstrate the usefulness of the simulation approach for diagnosis applications. As we can see, the sliding mode controllers are robust to load changes, and generates true 120 V rms voltage for both the loads. However, the light bulb fault affects the inverter current, and, therefore, the battery current and voltage. The AC fan current shows a phase difference of 0.1346 rad. As can be seen in Fig. 10, when the AC fan is switched on, its speed of rotation increases until it reaches a steady state of about 78.5 rad/s. On turning off, the speed falls to zero.

Table 3 presents the result of an experiment to illustrate the efficiency gained by simplifying a recon-

figurable BD model by identifying bonds with persistent causal assignments and junctions in fixed causality, and avoiding the need for causal reassignment for these modes. For this experiment, we assume that the AC fan is the only load and is on for the duration of the experiment. Each column in Table 3 reports the real time taken to simulate 1 second of simulation time for different HBG simulation runs. In all runs, all junctions of the HBG model are in fixed causality. In the first run, we call SCAP every time an inverter mode change occurs. Next, we repeat the previous run, using Hybrid SCAP. Finally, in the third run, we simulate the HBG without requiring any external calls to Hybrid SCAP, since all switching junctions are fixed. As can be seen from Table 3, our enhanced simulation approach, implemented in the third run, is 103.85 times faster than the first run, and 103.35 times faster than the second run. Our simulation approach also resulted in considerable improvements in the efficiency of simulation of a number of other configurations, especially for large systems like the VIRTUAL ADAPT simulation testbed (Poll et al., 2007). Further increase in simulation efficiency can be obtained by running our simulation models in the Rapid Accelerator mode of Simulink.

CONCLUSIONS

In this paper, we have presented an improved framework for simulating hybrid systems. The crux of these improvements is the identification of persistent causality of bonds, which not only avoids unnecessary invocations of the external `Hybrid SCAP` algorithm, thereby gaining increase in simulation efficiency, but also improves the efficiency of the `Hybrid SCAP` algorithm, as well as, enables the simplification of the simulation models by removing parts that correspond to configurations that never occur during the simulation. In the future, we will extend our modeling approach and computational model generation schemes to handle situations of derivative causality, and systematically evaluate how our approach performs when applied to other real-world large hybrid systems.

ACKNOWLEDGEMENTS

REFERENCES

Biel, D., Guinjoan, F., Fossas, E., and Chavarria, J. (2004). Sliding-mode control design of a boost-buck switching converter for ac signal generation. *IEEE Transaction on Circuits and Systems - I*, 51(8):1539–1551.

Borutzky, W. (1995). Discontinuities in a bond graph framework. *Journal of the Franklin Institute*, 332(2):141–154.

Buisson, J., Cormerais, H., and Richard, P.-Y. (2002). Analysis of the bond graph model of hybrid physical systems with ideal switches. *Proc Instn Mech Engrs Vol 216 Part I: J Systems and Control Engineering*, pages 47–63.

Daigle, M., Roychoudhury, I., Biswas, G., and Koutsoukos, X. (2007). Efficient simulation of component-based hybrid models represented as hybrid bond graphs. In Bemporad, A., Bicchi, A., and Butazzo, G., editors, *HSCC 2007*, volume 4416 of *LNCS*, pages 680–683. Springer-Verlag.

Karnopp, D. (2003). Understanding induction motor state equations using bond graphs. In *Proceedings of the International Conference on Bond Graph Modeling and Simulation*, volume 35, pages 269 – 273.

Karnopp, D. C., Margolis, D. L., and Rosenberg, R. C. (2000). *Systems Dynamics: Modeling and Simulation of Mechatronic Systems*. John Wiley & Sons, Inc., New York, third edition.

Krause, P. C. (1986). *Analysis of Electric Machinery*. John Wiley & Sons, Inc.

Magos, M., Valentin, C., and Maschke, B. (2003). Physical switching systems: From a network graph to a hybrid port hamiltonian formulation. In *Proc IFAC conf Analysis and Design of Hybrid Systems*, Saint Malo, France.

Manders, E.-J., Biswas, G., Mahadevan, N., and Karsai, G. (2006). Component-oriented modeling of hybrid dynamic systems using the Generic Modeling Environment. In *Proc of the 4th Workshop on Model-Based Development of Computer Based Systems*, Potsdam, Germany. IEEE CS Press.

Mosterman, P. J. and Biswas, G. (1998). A theory of discontinuities in physical system models. *Journal of the Franklin Institute*, 335B(3):401–439.

Poll, S., Patterson-Hine, A., Camisa, J., Nishikawa, D., Spirkovska, L., Garcia, D., Hall, D., Neukom, C., Sweet, A., Yentus, S., Lee, C., Ossenfort, J., Roychoudhury, I., Daigle, M., Biswas, G., Koutsoukos, X., and Lutz, R. (2007). Evaluation, selection, and application of model-based diagnosis tools and approaches. In *AIAA Infotech@Aerospace 2007 Conference and Exhibit*.

Roychoudhury, I., Daigle, M., Biswas, G., Koutsoukos, X., and Mosterman, P. J. (2007). A method for efficient simulation of hybrid bond graphs. In *Proceedings of the International Conference of Bond Graph Modeling*, pages 177 – 184, San Diego, California.

Thaler, G. J. and Wilcox, M. L. (1966). *Electric Machines: Dynamics and Steady State*. John Wiley & Sons, Inc.

van Dijk, J. (1994). *On the role of bond graph causality in modeling mechatronics systems*. PhD thesis, University of Twente, Enschede, The Netherlands.

AUTHOR BIOGRAPHIES

INDRANIL ROYCHOUDHURY received the M.S. degree in computer science from Vanderbilt University, Nashville, TN, in 2006, where he is currently a Ph.D. Candidate. His web page is http://people.vanderbilt.edu/~indranil.roychoudhury/.

MATTHEW DAIGLE received the Ph.D. degree in computer science from Vanderbilt University, Nashville, TN, in 2008. His web page can be found at http://people.vanderbilt.edu/~matthew.j.daigle/.

GAUTAM BISWAS received the Ph.D. degree in computer science from Michigan State University, East Lansing. He is a Professor of Computer Science and Computer Engineering in the Department of Electrical Engineering and Computer Science, Vanderbilt University, Nashville, TN. His web page can be found at http://www.vuse.vanderbilt.edu/~biswas/.

XENOFON KOUTSOUKOS received the Ph.D. degree in electrical engineering from the University of Notre Dame, Notre Dame, IN, in 2000. He is an Assistant Professor in the Department of Electrical Engineering and Computer Science, Vanderbilt University, Nashville, TN. His web page is http://www.vuse.vanderbilt.edu/~koutsoxd/.

ON THE STABILITY OF A CLASS OF SWITCHED BOND GRAPHS

Sergio Junco and Alejandro Donaire [1]
Departamento de Electrónica, FCEIyA
Universidad Nacional de Rosario
[1] CONICET
Riobamba 245 bis, Rosario, Argentina
{sjunco, adonaire}@fceia.unr.edu.ar

Ahmed Rahmani and Hicham Hihi
Ecole Centrale de Lille
LAGIS UMR 8146
59651 Villeneuve d'Ascq Cedex
Nord-Pas-de-Calais, France
{ahmed.rahmani, hicham.hihi}@ec-lille.fr

KEYWORDS
Bond Graphs, Switched Systems, Lyapunov Stability.

ABSTRACT

This paper deals with the stability analysis of a class of switched bond graphs where all the storage components remain unchanged and in integral causality under mode switching. These properties assure the existence of a unique energy function common to all the switching modes, and the non occurrence of state jumps or discontinuities in the switched system trajectories. A result on stability of equilibria in the sense of Lyapunov is presented. The derivation of this result is done using a bond graph technique for Lyapunov stability analysis (previously developed by one of the authors of this paper) in association with results available in the literature related to the satisfaction by a candidate Lyapunov function of a sequence nonincreasing condition for the state trajectories of a general autonomous switched system. Because of its versatility, the Switched Power Junction formalism is chosen in this paper to model and simulate the commutations in the bond graph domain.

1. INTRODUCTION

Frequently in engineering problems, abrupt changes in physical systems are considered to occur instantly. This is mainly due to the facts that the behavior the engineer is interested in has a time scale much bigger than that of the abrupt change, and that the details inside the time window of this change are not relevant to the behavior under study. Thus, ignoring them results in saving time and effort. As this practice departs from the assumptions of continuity and smoothness underlying classical physics, it requires special modeling, simulation and analysis tools to handle the systems it yields, see (Mosterman and Biswas, 1998) for a detailed discussion of modeling and simulation issues related to this problem.

This paper deals with the stability analysis of switched physical systems modeled as switched bond graphs. The main tools employed to represent the commutations among the continuous bond graph (BG) models constituting the switched bond graph (SwBG) is the Switched Power Junction formalism, which has been introduced in (Umarikar and Umanand, 2005), and revisited in (Junco et al., 2007) in view of its

implementation in the simulation tool 20sim®. There are two kinds of SPJ, the 0_S and the 1_S, which are 0- and 1-junctions admitting more than one bond *graphically* imposing effort or flow on them, respectively. The rules of causality are not violated because one and only one of these bonds is behaviorally connected (transferring power and relating variables) at a given arbitrary time instant, the others being excluded from the model. The decision on what bond to connect is made by a control variable associated to the SPJ.

A big amount of research has been dedicated to the stability of switched and, more generally, hybrid systems, as stability is both, theoretic and practically important. This problem has been approached with varied mathematical tools, ranging from differential inclusions, variational principles, multiple Lyapunov functions, Lie algebra, etc. Some relevant surveys on this subject are (Antsaklis and Nerode, 1998), (Morse et al., 1999), (Antsaklis, 2000), (Liberzon, 2003), (Margaliot, 2006).

The stability analysis of (continuous) BG has been approached in several works. An approach based on Lyapunov's Second Method and extensions (LaSalle's Invariant Principle), which exploits the system's energy, interconnection and causality features directly on the BG domain, has been presented in (Junco, 1993 and 2001), and further applied to analyze stability of equilibria and input-output passivity, as well as to synthesize stabilizing controllers for electrical machines in (Junco, 1999 and 2000) and (Junco and Donaire, 2005). More results on stability on BGs are presented in (Wu and Youcef-Toumi, 1995) and related papers by the same research group, and in (Grujic and Dauphin-Tanguy, 2000; see also previous work referenced therein). The first group of researchers presents a (non-Lyapunov) technique taylored to analyze stability properties of zero dynamics on BG. The second introduces the concept and method of energetic stability, which is Lyapunov-related but, because explicitly using state equations, it is not a pure BG-technique. For the case of linear BGs, (Morvan et al., 2003) introduced an observability-related algebraic criterion. The same work addresses a condition for the asymptotic stability of the origin being considered as a common equilibrium point (EP) of a linear, sourceless *switched BG* (i.e., an autonomous linear switched system). This result makes use of the notion of state observability from the power

variables of the R-elements of each linear BG constituting the SwBG.

In this paper we approach the stability analysis of SwBG using the energy-based methods developed by Junco for continuous BG (see (Junco, 2004) for a summary of the above cited results) immersed in a framework for general switched and hybrid system analysis as presented in, for instance, (Branicky, 1998) and (DeCarlo et al., 2000). More specifically, a class of SwBG is considered, where all the storage components remain unchanged and in integral causality under mode switching, what assures the existence of a unique energy function common to all the switching modes. Assuming that it is definite positive wrt an EP common to all the switching modes, stability in the sense of Lyapunov is proven provided that the energy function satisfies a sequence nonincreasing condition for the state trajectories.

This research is a first step towards the dynamic analysis of hybrid systems on graphical models, in our case SwBGs augmented with the graphical representation (via block diagrams or flow graphs, etc.) of the subsystems controlling or determinig their commutation rules. As shown in (Junco et al., 2007) the SPJ-formalism allows for this simultaneous representation of the complete hybrid system. Moreover, some preliminary studies, which will be the object of a future paper, indicate the feasibility of analyzing the hybrid dynamics on this representation with graphical tools in the style of (Rahmani and Dauphin-Tanguy, 2006), (Hihi and Rahmani, 2007a, b) and (Reinschke, 1988).

The remaining of this paper is organized as follows: Section 2 presents the modeling of SwBG means the SPJ formalism, addresses some previous results on stability of continuous BGs, and summarizes the background on stability of switched systems. Section 3 presents the main result of the paper concerning the stability of SwBG. Section 4 illustrates some properties and results with simulation examples and, finally, Section 5 presents the conclusions of this research.

2. BACKGROUND ON STABILITY AND SWITCHED BOND GRAPHS.

This section summarizes some background results on stability of BGs and stability of switched systems that will be used in the sequel, and presents the SPJ-formalism as the modeling tool chosen to represent the switched bond graphs.

Lyapunov Stability on Bond Graphs

To fix ideas consider the (possibly) nonlinear dynamical system $\dot{x} = f(x)$ and, without loss of generality, suppose that the state space origin is an EP, i.e., $f(0) = 0$. Analyzing the stability of the origin using Lyapunov's second method basically implies choosing a scalar *positive definite function* (pdf) $V(x)$ (written $V(x) > 0$)

and studying the sign of $L_f V(x)$ (where $L_f V(x) \equiv dV(x(t))/dt$ or, for short, $\dot{V} \equiv \dot{V}(x(t))$, is the orbital or Lie-derivative of V i.e., its time-derivative along the orbits, or trajectories, or solution paths of the state equation system). This is usually done evaluating the scalar product between the gradient of $V(x)$ and the vector field $f(x)$, i.e., $\dot{V}(x(t)) = \nabla_x^T V(x) \cdot f(x)$. If $\dot{V}(x)$ is negative semidefinite (written $\dot{V}(x) \le 0$), then the EP is (at least) stable in the sense of Lyapunov; if $\dot{V}(x)$ is negative definite (written $\dot{V}(x) < 0$), then the EP is asymptotically stable (applies only to isolated EPs). In this latter case, $V(x) > 0$ *is* (*called*) *a Lyapunov function* for the EP. For details on Lyapunov's theory and its applications to control problems refer for instance to (Sontag, 1998).

When studying physical systems a natural candidate for a pdf $V(x)$ is the energy E stored in the system. If this assumption is satisfied, then $\dot{V} \equiv \dot{E}$ is the power flowing into the storages, and can be evaluated directly on the BG as the power flowing out of the sources minus the power into the dissipative elements. Thus, there is no need of computing the scalar product determining $\dot{V}$, nor the vector field $f(x)$, i.e., the state equations. This fact constitutes the rationale underlying the Internal Stability Analysis Procedure given as Proposition 2.1 in (Junco, 2001). Further results on the direct application on BGs of Lyapunov's Second Method are summarized next. Without loss of generality, the isolated EP considered is always the state-space origin. The results apply without restriction to BGs consisting of elements out of the basic nine-component set, but they also hold in many cases on BGs containing modulated components. **In all cases of the transcription below, the energy E is supposed to be a pdf of the states**, if not, energy-based pdfs can be considered, as formulated in (Junco, 1993).

Proposition 1 (Junco, 2004; Proposition 2.2 in [Junco, 2001[*]]).
If the energy stored in a BG is a Lyapunov function $V(x)$ (i.e., $V > 0$ and $\dot{V} < 0$), then each storage-element imposes causality on at least one R-element of the BG.

Proposition 2 (Junco, 2004; Proposition 2.3 in [Junco, 2001]).

A) The EP is asymptotically stable if the following conditions hold:

(i) all R-elements are truly dissipative, i.e., each *e-f* relationship goes through the origin and is completely contained in the first and third quadrants;

(ii) each storage in integral causality imposes causality on at least one R-element.

---

[*] In the uncorrected conference version of this paper the proposition is erroneously stated as "A necessary condition for asymptotic stability". In fact it is "A necessary condition for E being a Lyapunov function".

Condition *(ii)* guarantees that the power dissipated in the R-elements depends on all the state variables; together with condition *(i)* this ensures that this dependence is a *pdf* function of the state. As $\dot{V}$ equals minus the dissipated power, the sufficient condition for asymptotic stability is satisfied.

In fact, the following stronger property holds for the simplest class of BGs:

Proposition 3 (Junco, 2004).
The EP of any connected BG (i.e., a BG not composed of disjoint BGs) constructed with elements out of the basic set and including strictly dissipative R's is asymptotically stable.
This property follows from the assumed positive definiteness of the stored energy, the strictly dissipative features of the R's, and the pure power-connected structure of this class of BGs. It can be proved applying Lyapunov's Second Method and LaSalle's Invariance Principle on BGs, see (Junco, 2001), Proposition 2.4.

Lyapunov Stability of Switched Systems

Among the many results on stability of switched systems, we will use the version of the approach via multiple Lyapunov functions presented in (Branicky, 1998), reproduced next with slight simplifications for the sake of brevity.

Theorem 2.7 (Branicky, 1998). Suppose the candidate Lyapunov functions $V_i(x)$ for the switched dynamics defined by $\dot{x}(t) = f_i(x(t))$, $i \in \{1, 2, 3, …,N\}$ with $f_i(0)=0$. Let **S** be the set of all switching sequences associated with the system.
If for all *i* and each switching sequence S∈**S**, V_i is Lyapunov-like for f_i and the trajectory-segment in mode-*i*, and the V_i satisfy the sequence nonincreasing condition for the whole trajectory (encompassing all the modes of S), then the system is stable in the sense of Lyapunov.

Definition 2.6 (Branicky, 1998). The sequence non-increasing condition for the candidate Lyapunov functions $V_i(x)$ in Theorem 2.7 above means that

$$V_{i_{j+1}}(x(t_{j+1})) < V_{i_j}(x(t_j))$$

The ideas behind this theorem is that "these possibly multiple Lyapunov-like functions can be pieced together in some way to produce a global (nontraditional) Lyapunov function whose overall energy decreases to zero along the system state trajectories" (DeCarlo et al., 2000).

Modeling Switched BGs with the SPJ-formalism

The need to incorporate some tools to handle switching phenomena with the BG-formalism, originally conceived to handle continuous physics, was quickly recognized by the BG research community. Among the many ideas and techniques proposed to treat *idealized commutations* we have selected the Switched Power Junction or SPJ-formalism. This is because of the advantges of the SPJs in order to circumvent some associated modeling drawbacks like varying causality of switching-modeling components, hanging junctions, failure to disconnect subsystems, and other inconsistencies. Most important for this research is the already mentioned fact that SPJs allow for a complete graphical representation of a hybrid system, via the bidirectional coupling of the SwBG with any informational graph used to explicit the subsystems determinig the commutations of the former.

Switched Power Junctions have been introduced in (Umarikar and Umanand, 2005) as a generalization or extension of standard BG-junctions. The generalized models of both $\mathbf{0_S}$ and $\mathbf{1_S}$ (the notation for 0- and 1-switched power junctions, respectively) are shown in Fig. 1. If they were standard junctions there would be a causal conflict in each of both cases. In the new formalism the causal conflict is removed via imposing the constraint that one and only one of the effort- (flow-) deciding bonds imposes the effort (flow) to the $\mathbf{0_S}$ ($\mathbf{1_S}$) at any given time instant. This convention is graphically represented by the presence of the activated bond in the junction symbol, and mathematically formalized in (1) and (2) for the $\mathbf{0_S}$ and $\mathbf{1_S}$, respectively. One and only one of the "boolean" variables in the set $\{U_1, U_2, U_3, …, U_N\}$ is allowed to have the value 1 at a given time instant, the rest are zero. Looking at (1) and (2) it is seen that the boolean variables assuming the zero values annihilate both effort and flow in each of the associated bonds, and thus completely disconnect all subsystems at the bond-ends opposite to the junction.
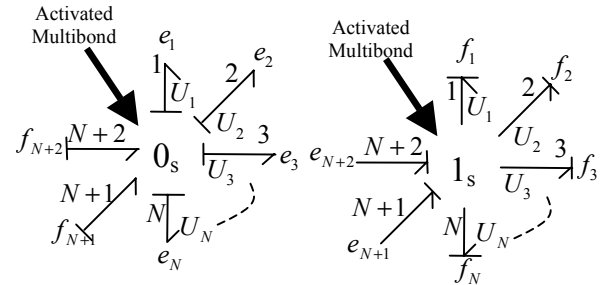
Fig. 1. Generalized $\mathbf{0_S}$- and $\mathbf{1_S}$-junctions.

Table 1. Constitutive relations of the SPJs

$\mathbf{0_S}$	*Junction effort* $= U_1 \cdot e_1 + U_2 \cdot e_2 + \cdots + U_N \cdot e_N$ $f_i = U_i \cdot (f_{N+1} + f_{N+2})$; $i = 1,...,N$	(1)
$\mathbf{1_S}$	*Junction flow* $= U_1 \cdot f_1 + U_2 \cdot f_2 + \cdots + U_N \cdot f_N$ $e_i = U_i \cdot (e_{N+1} + e_{N+2})$; $i = 1,...,N$	(2)

3. LYAPUNOV STABILITY OF SWITCHED BG

Some definitions are first introduced in order to characterize the class of SwBG this section deals with.

Definition 1. A Storage Invariant Integral SwBG (SI-I SwBG) is a switched BG where *i*) the storage elements are the same in all modes, and *ii*) all the storages are in integral causality in all modes.

Lemma 1. A SI-I SwBG has a unique energy function common to all its modes. This property follows immediately from the definition above and the fact that the energy function in a BG is fully determined by the state variables of the integral storages (the storages in integral causality).

Stability of Storage Invariant Integral SwBGs

Theorem 1. Consider a SI-I SwBG where the state space origin is an isolated EP common to all its modes. Further consider that the number of switchings in any finite time interval is finite, and that:

i) the energy function $E(p, q)$ is a (at least locally) *pdf* around the origin. Here p and q denote, respectively, the vectors collecting all generalized impulses and displacements of the SwBG.
ii) the BG_i associated to each mode is connected (i.e., it is not composed of disjoint BGs).
iii) all the R-elements are strictly dissipative.
iv) in each BG_i, each integral storage imposes causality on at least one R-element.

Then, the state space origin of the SwBG is asymptotically stable.

Proof. Because of assumption *i)* the energy function $E(p, q)$ is a candidate Lyapunov function in the sense of Theorem 2.7 of Branicky. This is a particular case where $V_i(x) \equiv E(p, q)$ is the same for all modes indexed by $i \in \{1, 2, 3, \dots, N\}$. Assumptions *ii)-iv)* imply that each mode, as considered per se, is asymptotically stable (Proposition 2, Junco 2004, see above), and thus, $\dot{V}_i(x)$ decreases with time. Piecing all this together shows that the conditions for the energy function $E(p, q)$ to satisfy the sequence nonincreasing condition (Definition 2.6 of Branicky) are satisfied. □

Some of the conditions of *Theorem 1* can be relaxed and asymptotic stability still retrieved, as commented next.

Remark 1. Assumption *iv)* implies that there is at least one R-element present in each mode. This condition can be removed, imposing instead the following condition to the switching sequences:

iv-bis) in each BG_i containing R-elements, each integral storage imposes causality on at least one of them. There is no switching sequence in S such that the system remains indefinitely in a (sequence of) mode(s) without R-elements.

Indeed, the sequence nonincreasing condition is recovered if we consider as a single mode the concatenation of the mode preceding the mode without dissipation with this latter one. □

Remark 2. Assumption *ii)* can be relaxed to handle non-connected or disjoint BG_i. In this case condition *iv)* should be imposed to each sub-BG composing a disjoint BG_i. If there are sub-BGs without R-elements, condition *iv)* can again be relaxed in the sense of condition *iv-bis).* □

Remark 3. Assumption *iv)* can be relaxed if the concerned BG_i's satisfy the conditions of Proposition 3 above. □

We remark that, rigorously speaking, the class of SwBG considered here does not strictly imply a multiple, but a single Lyapunov function, and thus, we handle a particular case of Branicky's theorem. Note however that a multiplicity is associated to this single energy function through the switching dynamics. Indeed, *any switching sequence provokes* a sequence *of negative (semi)definite orbital derivatives*, i.e., a multiple orbital derivative state-function $\dot{V}_i(x) = \dot{E}(p,q)|_{Mode\ i}$.

4. A MODELING AND SIMULATION EXAMPLE

Consider the mechanical sketch of Fig. 2. Two ideal mechanical couplers indicated $Sw_{1,2}$ can be noticed, which serve to couple the mass+two-dampers system in the middle to the mass-spring-damper systems on the sides. The following switching conditions are assumed for each coupler: *i)* switch closes on contact; *ii)* switch opens when $b_{1/2}$-damper compression force becomes zero. We identify the switch binary states as follows: $Sw_i=0$, switch open (disengaged); $Sw_i=1$, switch closed (engaged). The SwBG consists then of four switching modes, corresponding to the four binary states of the pair (Sw_1, Sw_2) $\{(0, 0), (1, 0), (0, 1), (1, 1)\}$. This is represented in Fig. 3 by four separated BGs. As inactive dampers do not play any role (dampers $b_{1,2}$ when related switch in the off-state), the associated R-elements have not been retained in the corresponding BGs.
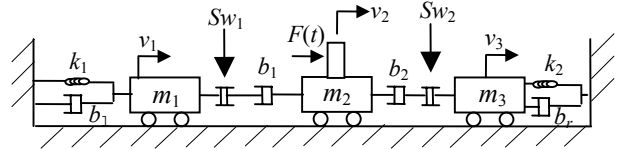
Fig. 2. Mechanical system

Sw-states	Physical system	Bond graph
(0, 0)		
(1, 0)		
(0, 1)		
(1, 1)		

Fig. 3. The four modes of the switched system

Switched Bond Graph Modeling

Fig. 4 is a SwBG representation of the example system using the (primitive) switch element or Sw-element (Strömberg, 1994, p. 86). Naturally, here all the BG-elements are present, even if they may not be active in some modes. As in the previous series of BGs, also here causality has been indicated: note that the causal stroke is pictured in the middle of a bond when this changes causality in dependence of the switch state, i.e., causality is undecided unless the switch state is specified. It means that the switched system is only specified up-to causality when using this formalism to construct the SwBG.
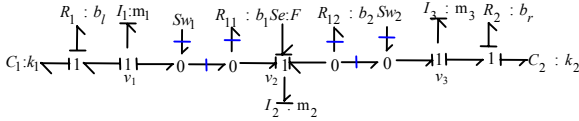
Fig. 4. The SwBG model using the ideal Sw-element.

Fig. 5 is a representation of the SwBG using the SPJ-formalism. The 0_S-SPJs are controlled by the signals u_i, which correspond to the states Sw_i as follows: $u_i = 1$ means Sw_i closed; $u_i = 0$ means Sw_i open. Consider for instance the Sw_1: when $u_1 = 1$ the 0_S ignores the effort $e_{C1}=0$ on the lower bond (associated to the auxiliary S_e with null effort), chooses the effort on the upper bond (associated to the R-element), and transmits it to the right and left bonds. Simultaneously, a net flow equal to the flow on the left bond minus the flow on the right bond is transmitted to the upper bond, and zero flow to the lower bond. This corresponds to a situation where the coupler is engaged. The opposite effort selection happens when $u_1 = 0$, meaning that zero effort is transmitted to left and right (coupler disengaged). Simultaneously, zero flow is transmitted to the R-element, what determines its inactivity during this mode. The behaviour just described is what Eq. (1) means in this particular case (see Table 1) (along with the rules on the switching conditions). Clearly, the SwBG is fully specified when using the SPJ-formalism for its representation; see (Junco et al., 2007) for other examples, including changing causality in storages and R-elements.
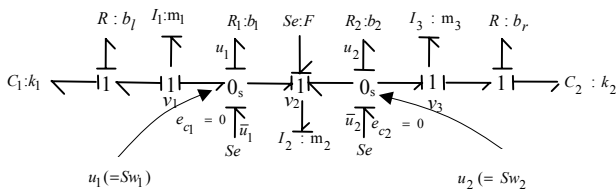
Fig. 5. SPJ- representation of the SI-I SwBG model associated to the example of Fig.2.

Stability Analysis of Switched Bond Graph

As the storages remain invariant and in integral causality under switching (only the two R-elements with coefficients $b_{1,2}$ become inactive here), the SPJ-BG of Fig.5 is a SI-I SwBG. The state vector is the same in all modes $x^T = [q_1, p_1, p_2, p_3, q_2]$, and so is the energy function. Clearly, all the BG_i share the state space origin as an isolated EP, i.e., it is a EP common to all the modes. Observe that the position of mass m_2 is not a BG state, if it were the EP would not be isolated. Naturally, the inertias are linear, but even if so depicted in the figure, the springs do not need to be linear. We do not write them down, but simply assume possibly nonlinear constitutive laws of the C's such that the energy function is a pdf of the states, so that assumption *(i)* of Theorem 1 is satisfied. Considering pure dissipative (also not necessarily linear) R-elements, assumption *(iii)* is also satisfied. Assumption *(ii)* is not satisfied because $BG_{(0,0)}$ (the BG of mode (0,0)) is disjoint. Assumption *(iv)* is not satisfied because *(a)*, none of the two C's imposes causality in a R-element, and *(b)*, in mode $BG_{(0,0)}$ the mass m_2 is isolated without R. The first situation is solved by Remark 3 and, regarding the second, we are in the situation foreseen in *Remark 2* with a condition similar to *iv-bis*, because, when out of equilibrium, mass m_2 will eventually contact either damper b_1 or b_2 and this mode will be abandoned in finite time.

All the above means that, by virtue of Theorem 1 and the modifications foreseen in Remarks 1-3, the state space origin of the switched system given in Fig. 2 is an asymptotically stable equilibrium point.

We calculate next the orbital derivatives of the energy function in the different modes. The energy function itself is:

$$E(p,q) = E_1(q_1) + E_2(q_2) + p_1^2/2m_1 + p_2^2/2m_2 + p_3^2/2m_3$$

To get a simpler formula for the orbital derivative it is better to perform an obvious change of state variables and consequently write the energy function as:

$$V(v,q) = E_1(q_1) + E_2(q_2) + m_1 v_1^2/2 + m_2 v_2^2/2 + m_3 v_3^2/2$$

Clearly, with the assumptions made on the constitutive laws of the C-elements, the energy function is a pdf of the states. Its orbital derivative in each of the modes is as follows:

$$\dot{V}_{(0,0)}(v,q) = -b_l v_1^2 - b_r v_3^2$$

$$\dot{V}_{(1,0)}(v,q) = -b_l v_1^2 - b_r v_3^2 - b_1(v_1 - v_2)^2$$

$$\dot{V}_{(0,1)}(v,q) = -b_l v_1^2 - b_r v_3^2 - b_2(v_2 - v_3)^2$$

$$\dot{V}_{(1,1)}(v,q) = -b_l v_1^2 - b_r v_3^2 - b_1(v_1 - v_2)^2 - b_2(v_2 - v_3)^2$$

It is clearly seen that *a multiple orbital derivative is obtained*. It is a sequence of negative semidefinite functions of the state, as none of them depend on the q-variables in any of the modes, because none of the C-elements imposes causality to any of the R-elements in any of the BG_i.

Simulation Results

The SPJ-model of Fig. 5 has been implemented in the 20sim™ simulator, and a simulation experiment has been conducted with the example system parameterized

as follows: b_1 =1; b_2 =1; b_l =0.1; b_r =0.5; k_1 =0.1; k_2 =2; m_1 =1; m_2 =4 and m_3 =1. The initial conditions of the experiment are: $x_1(0)$=0.2; $x_2(0)$=0.8; $x_3(0)$=1.1; where x_i is the position of the i-th mass. The left end of damper b_1 is at a distance $z_1(0)$=0.15 to the left of mass m_2; the right end of damper b_2 is at a distance $z_2(0)$=0.05 to the right of mass m_2; $v_1(0)$= – 0.025. SI-units are used for all magnitudes –but are not given here.

Figs. 6 and 7 plot the positions and the speeds of the three masses. With masses 1 and 3 at rest and mass 2 moving (at constant speed v_1= – 0.025) to the left, the system is initially in mode (0,0). In this mode the energy function remains constant and no power is dissipated, cf. the expression of $\dot{V}_{(0,0)}$ above (where v_3 and v_1 are zero), as well as Fig. 8 below. This latter figure also shows that the switches are open (U_1=Sw_1=0, U_2=Sw_2=0). The system commutes to mode (1,0) when the left end of damper b_1 gets in contact with mass 1, which initiates an oscillatory movement, while mass 2 decelerates first and inverts its motion later. Most of the energy originally stored in the system gets dissipated (in dampers b_l and b_1) during this mode, cf. $\dot{V}_{(1,0)}$ (where only v_3 is zero) and Fig. 8, which also shows that the control signal of switch 1 is high, i.e., U_1=Sw_1=1.
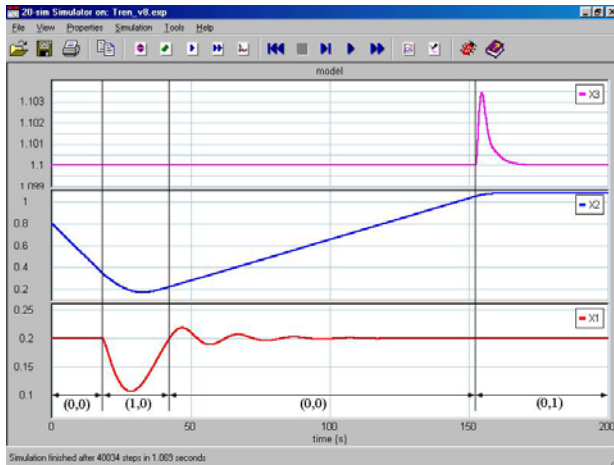
Fig. 6. Time trajectories of the mass positions.
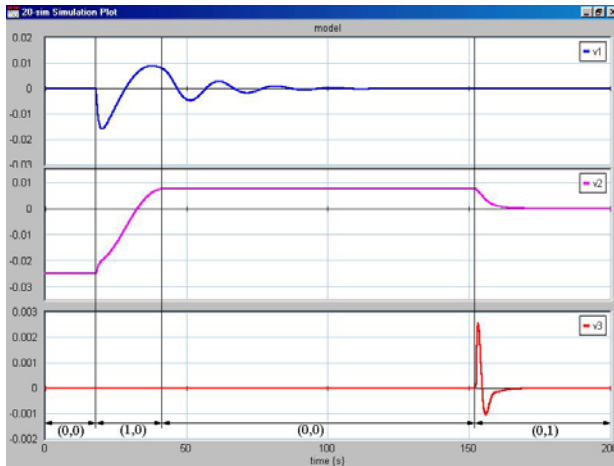
Fig. 7. Time trajectories of the mass velocities.

The back oscillation of mass 1 pushes mass 2 to the right and brings the switch 1 to its disconnection condition, so that the system commutes back to mode (0,0), but with a different energy configuration: while mass 2 moves at (positive) constant speed, the left mass-spring-damper subsystem is (slightly) excited, what means that (a small amount of) energy continues to get dissipated, cf. $\dot{V}_{(0,0)}$ (v_3 is zero, but not v_1). After a while, the right end of damper b_2 contacts mass 3, the switch 2 goes in the on-state (see U_2 in Fig. 8), and the system commutes to mode (0,1). With the movement of the left subsystem already faded (v_1=0), energy is only dissipated in the coupled subsystem on the right, cf. $\dot{V}_{(0,1)}$, where only v_1=0, and Fig. 8.
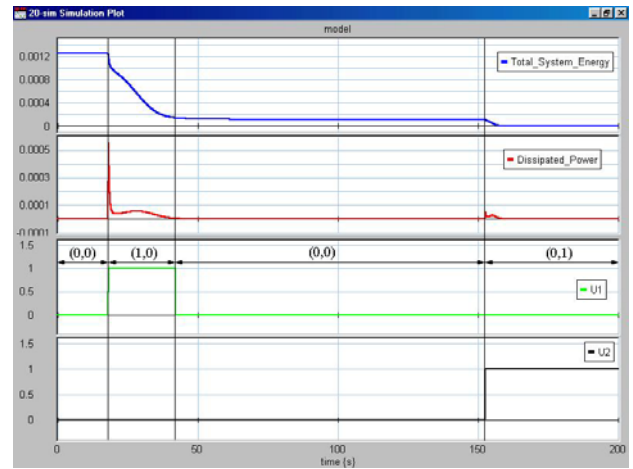
Fig. 8. The energy function, the power dissipated in the system, and the control signals of the switches (U_i=Sw_i)

The first plot of Fig. 8 shows the nonincreasing nature of the energy, which is the candidate Lyapunov function in this problem. It converges to zero, what means that the state converges to the origin, what in turn proves the asymptotic stability of this EP. This convergence of the state variables can be partially seen on the speeds (Fig. 7). The complementary conditions, namely that the deformations of the springs also go to zero, are also verified, even if not shown on the plots.

5. CONCLUSIONS

The stability of a particular class of switched bond graphs has been analyzed with Lyapunov-like tools applied directly in the BG-domain. The study is restricted to BG where the storages remain invariant under switching, both in their constitution and causality. Further research will focus on switched bond graphs admitting derivative causality when switching among modes. This implies the lost of a unique energy function common to all modes of the BG, the possible variation of the system order when switching, and the appearance of discontinuities in the state trajectories.

REFERENCES

Antsaklis, P. and A. Nerode (Eds.) 1998. Special Issue on Hybrid Control Systems, IEEE Tr. AC, Vol. 43, No. 4.

Antsaklis, P. (Ed.) 2000. Special Issue on Hybrid Systems, Proc. IEEE, Vol. 88, No. 7 (July).

Branicky, M. S (1998). Multiple Lyapunov functions and other analysis tools for switched and hybrid systems. IEEE Trans. Automat. Contr., vol. 43, pp. 475-482, Apr. 1998.

Controllab Products B.V., 20-sim™, http://www.20sim.com

DeCarlo, R.; M. Branicky; S. Pettersson; and B. Lennartson. 2000. "Perspectives and results on the stability and stabilizability of hybrid systems." Proc. IEEE, Vol. 88, No. 7 (July), 1069-1082.

Grujic, S. and G. Dauphin-Tanguy. 2000. "Stabilité Energétique Exponentielle", Actes de CIFA 2000, 4-7 juillet 2000, Lille, France.

Hihi, H and A. Rahmani. 2007a. Structural properties of switching bond graph. *Eurosim 07, Ljubljana, Slovenia September 9-13, 2007*.

Hihi, H and A. Rahmani. 2007b. A sufficient and necessary conditions for the controllability of switching linear systems. *SMC07, Canada, October 2007*.

Junco, S. 1993. "Stability Analysis and Stabilizing Control Synthesis via Lyapunov's Second Method Directly on Bond-Graphs on Nonlinear Systems". Proceedings of IECON'93, pp. 2065-2069, Maui, HII, Nov. 17–20.

Junco, S. 1999. "Real- and Complex-Power Bond Graph Modeling of the Induction Motor", Proceedings ICBGM'99 /Simul. Series 31:1.

Junco, S. 2000. "Linéarisation Exacte Entrée-Sortie et Stabilité de la Dynamique des Zéros Directement sur Bond-Graphs de Systèmes Nonlinéaires", Actes de CIFA 2000, Lille, France.

Junco, S. 2001. "Lyapunov Second Method and Feedback Stabilization Directly on Bond Graphs". Proc. ICBGM'2001, SCS-Simulation Series, 33:1, 137-142.

Junco, S. 2004. Virtual Prototyping of Bond Graphs Models for Controller Synthesis through Energy and Power Shaping. Proc. IMAACA'2004, Genoa, Italy, Vol. 2, pp. 100-109.

Junco, S. and A. Donaire. 2005. BG-Supported Synthesis of Speed- and Position-Tracking Controllers for Brushless DC-Motor Drives. Proc. ICBGM'05, New Orleans (USA) pp. 245-251.

Junco, S.; G. Diéguez; and F. Ramírez. 2007. "On Commutation Modeling in Bond Graphs". In *Proc. ICBGM'07*, San Diego, CA, pp. 115-119.

Liberzon, D. 2003. *Switching in systems and control*. Boston: Birkhauser.

Margaliot, M. 2006. Stability analysis of switched systems using variational principles: An introduction. Automatica, Vol. 42, 2059-2077.

Morse, A.; C. Pantelides; S. Sastry; and J. Schumacher (Eds.) 1999. Special Issue on Hybrid Systems, Automatica, Vol. 35, No. 3.

Morvan, C. ; H. Cormerais ; J. Buisson ; and P.Y Richard. 2003. Deriving asymptotic stability of linear switching physical systems. Multiconference CESA'03, Lille, July 2003.

Mosterman, P. and G. Biswas. 1998. "A theory of discontinuities in physical systems models", *J. Franklin Institute*, 335B (3), 401 - 439.

Rahmani A. and G. Dauphin Tanguy. 2006. Structural analysis for switching devices modelled by bond graph. *J. of Mathematical and Computer Modelling of Dynamical Systems, volume 12, N°2-3, April-june 2006, Ed Taylor and Francis*

Reinschke, K. 1988. Multivariable control: A graph-theoretic approach. Lect. Notes in Cont and Inf. Sciences, n°108, Springer-Verlag.

Strömberg. Jan-Erik. 1994. "A mode switching modelling philosophy", Linköping Studies in Science and Technology. Dissertations, No. 353, Linköping.

Umarikar, A. and L. Umanand. 2005, "Modelling of switching systems in bond graphs using the concept of switched power junctions", *J. Franklin Institute*, 342, 131-147.

Wu, S.-T and K. Youcef-Toumi. 1995. "On Relative Degrees and Zero Dynamics From Physical System Modeling". Journal of Dynamic Systems, Measurement, and Control, Trans. ASME, Vol. 117, pp. 205-217.

AUTHOR BIOGRAPHIES

Sergio JUNCO is an Associate Professor at the Department of Electronics, Universidad Nacional de Rosario, Argentina. His research interests are in modeling and simulation of nonlinear physical systems with bond graphs, and discrete-event and hybrid systems with mixed techniques including the DEVS formalism. He is also interested in technical applications in mechatronics.

Ahmed RAHMANI received the PhD in Control Engineering from the Ecole Centrale Lille (France) in 1993. Currently, he is an associate Professor. His research interests include: hybrid systems and bond graphs with applications in robotics.

Hicham HIHI received the bachelor degree in Electrical Engineering from University of Artois (France) in 2001, the Master degree in Electronics, Electrical and Control Engineering from University of Sciences and Technologies of Lille (France) in 2002, and the Master degree in Control Engineering from Ecole Centrale Lille (France) in 2003. Currently, he is an PhD student in LAGIS (Laboratoire d'Automatique, de Génie Informatique et Signal), France.
His research interests include: analysis of switched and hybrid systems, bond graphs.

Alejandro DONAIRE received the degree in Electronic Engineering at the National University of Rosario (Argentina) in 2003, and is now finishing his PhD Thesis in Control Engineering at the same university. He is interested in modeling, simulation and control of nonlinear systems with energy-based techniques.

Modelling, Simulation and Control of Technological Processes

SIMULATION OF STEADY-STATE AND DYNAMIC BEHAVIOUR OF A TUBULAR CHEMICAL REACTOR

Petr Dostál, Vladimír Bobál, and Jiří Vojtěšek
Department of Process Control, Faculty of Applied Informatics
Tomas Bata University in Zlin
Nad Stráněmi 2911, Zlín 760 05, Czech republic
E-mail: dostalp@fai.utb.cz

KEYWORDS

Tubular chemical reactor, mathematical model, steady-state, dynamics.

ABSTRACT

The paper presents some results concerning analysis and simulation of steady-state and dynamic behaviour of a tubular chemical reactor. This analysis represents a necessary condition for the reactor control design purposes. The mathematical models used in simulations together with simulation results are contained.

INTRODUCTION

Tubular chemical reactor are units frequently used in chemical and biochemical industry. From the system theory point of view, tubular chemical reactors belong to a class of nonlinear distributed parameter systems. Their mathematical models are described by sets of nonlinear partial differential equations (PDR). The methods of modelling and simulation of such processes are described eg. in (Luyben 1989; Ingham et al. 1994; Severance 2001; Babu 2004). Relations between process behaviour and their control methods can be found in (Seborg et al. 1989; Ogunnaike and Ray 1994; Marlin 1995; Corriou 2004).

It is well known that the control of chemical reactors, and, tubular reactors especially, often represents very complex problem. The control problems are due to the process nonlinearity, its distributed nature, and high sensitivity of the state and output variables to input changes. In addition, the dynamic characteristics may exhibit a varying sign of the gain in various operating points, the time delay as well as non-minimum phase behaviour. Evidently, the process with such properties is hardly controllable by conventional control methods, and, its effective control requires application some of advanced methods (e. g. Adaptive Control, Predictive Control, Robust Control or any others). However, at all events, a previous analysis of the process behaviour is obligatory.

The paper presents all mathematical models used for simulations of both steady-state and dynamic charakteristics of the tubular chemical reactor together with results of some simulations. The combinations of observed variables are chosen in accordance with purposes of prospective control design.

MODEL OF THE PLANT

An ideal plug-flow tubular chemical reactor with a simple exothermic consecutive reaction $A \xrightarrow{k_1} B \xrightarrow{k_2} C$ in the liquid phase and with the countercurrent cooling is considered as shown in Fig. 1.
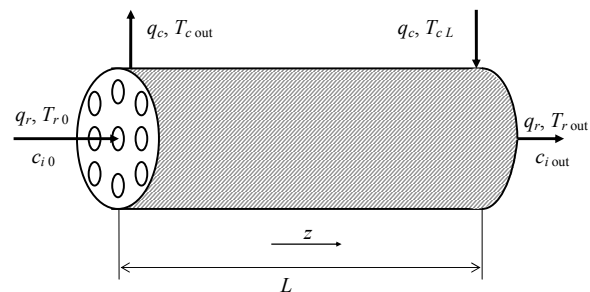
Figure 1: Tubular chemical reactor.

Heat losses and heat conduction along the metal walls of tubes are assumed to be negligible, but dynamics of the metal walls of tubes are significant. All densities, heat capacities, and heat transfer coefficients are assumed to be constant. Under above assumptions, the reactor model can be described by five PDRs in the form

$$\frac{\partial c_A}{\partial t} + v_r \frac{\partial c_A}{\partial z} = -k_1 c_A \tag{1}$$

$$\frac{\partial c_B}{\partial t} + v_r \frac{\partial c_B}{\partial z} = k_1 c_A - k_2 c_B \tag{2}$$

$$\frac{\partial T_r}{\partial t} + v_r \frac{\partial T_r}{\partial z} = \frac{Q_r}{(\rho c_p)_r} - \frac{4 U_1}{d_1 (\rho c_p)_r}(T_r - T_w) \tag{3}$$

$$\frac{\partial T_w}{\partial t} = \frac{4}{(d_2^2 - d_1^2)(\rho c_p)_w}\Big[d_1 U_1 (T_r - T_w) + \\ + d_2 U_2 (T_c - T_w) \Big] \tag{4}$$

$$\frac{\partial T_c}{\partial t} - v_c \frac{\partial T_c}{\partial z} = \frac{4 n_1 d_2 U_2}{(d_3^2 - n_1 d_2^2)(\rho c_p)_c}(T_w - T_c) \tag{5}$$

with initial conditions

$$c_A(z,0) = c_A^s(z), \quad c_B(z,0) = c_B^s(z), \quad T_r(z,0) = T_r^s(z),$$

$$T_w(z,0) = T_w^s(z) , \quad T_c(z,0) = T_c^s(z)$$

and boundary conditions

$$c_A(0,t) = c_{A0}(t) , \quad c_B(0,t) = c_{B0}(t) , \quad T_r(0,t) = T_{r0}(t) ,$$

$$T_c(L,t) = T_{cL}(t) .$$

Here, t is the time, z is the axial space variable, c are concentrations, T are temperatures, v are fluid velocities, d are diameters, ρ are densities, c_p are specific heat capacities, U are heat transfer coefficients, n_1 is the number of tubes and L is the length of tubes. The subscript $(\cdot)_r$ stands for the reactant mixture, $(\cdot)_w$ for the metal walls of tubes, $(\cdot)_c$ for the coolant, and the superscript $(\cdot)^s$ for for steady-state values.

The reaction rates and heat of reactions are nonlinear functions expressed as

$$k_j = k_{j0} \exp\left(\frac{-E_j}{RT_r}\right), \quad j = 1, 2 \qquad (6)$$

$$Q_r = (-\Delta H_{r1}) k_1 c_A + (-\Delta H_{r2}) k_2 c_B \qquad (7)$$

where k_0 are pre-exponential factors, E are activation energies, $(-\Delta H_r)$ are in the negative considered reaction entalpies, and R is the gas constant.

The fluid velocities are calculated via the reactant and coolant flow rates as

$$v_r = \frac{4 q_r}{\pi n_1 d_1^2} , \quad v_c = \frac{4 q_c}{\pi(d_3^2 - n_1 d_2^2)} \qquad (8)$$

The parameter values with correspondent units used for simulations are given in Table 1.

Table 1: Parameter values

$L = 8$ m	$n_1 = 1200$
$d_1 = 0.02$ m	$d_2 = 0.024$ m
$d_3 = 1$ m	
$\rho_r = 985$ kg/m^3	$c_{pr} = 4.05$ kJ/kg K
$\rho_w = 7800$ kg/m^3	$c_{pw} = 0.71$ kJ/kg K
$\rho_c = 998$ kg/m^3	$c_{pc} = 4.18$ kJ/kg K
$U_1 = 2.8$ kJ/m^{2}s K	$U_2 = 2.56$ kJ/m^{2}s K
$k_{10} = 5.61 \cdot 10^{16}$ 1/s	$k_{20} = 1.128 \cdot 10^{18}$ 1/s
$E_1/R = 13477$ K	$E_2/R = 15290$ K
$(-\Delta H_{r1}) = 5.8 \cdot 10^4$ kJ/kmol	$(-\Delta H_{r2}) = 1.8 \cdot 10^4$ kJ/kmol

From the system engineering point of view, $c_A(L,t) = c_{A\,\text{out}}$, $c_B(L,t) = c_{B\,\text{out}}$, $T_r(L,t) = T_{r\,\text{out}}$ and $T_c(0,t) = T_{c\,\text{out}}$ are the output variables, and, $q_r(t)$, $q_c(t)$, $c_{A0}(t)$, $T_{r0}(t)$ and $T_{cL}(t)$ are the input variables. Among them, for the control purposes, mostly $q_r(t)$ and $q_c(t)$ can be taken into account as the

control variables, whereas other inputs enter into the process as disturbances. As the controlled output, next to $c_{B\,\text{out}}$ also the mean reactant temperature given by

$$T_m(t) = \frac{1}{L} \int_0^L T_r(z,t)\, dz \qquad (9)$$

can be under some assumptions considered.

COMPUTATION MODELS

For computation of both steady-state and dynamic characteristics, the finite diferences method is employed. The procedure is based on substitution of the space interval $z \in\, <0, L>$ by a set of discrete node points $\{z_i\}$ for $i = 1, \dots , n$,and, subsequently, by approximation of derivatives with respect to the space variable in each node point by finite differences. Two types of finite differences are applied, either the backward finite difference

$$\left.\frac{\partial y(z,t)}{\partial z}\right|_{z=z_i} \approx \frac{y(z_i,t) - y(z_{i-1},t)}{h} = \frac{y(i,t) - y(i-1,t)}{h} \quad (10)$$

or the forward finite difference

$$\left.\frac{\partial y(z,t)}{\partial z}\right|_{z=z_i} \approx \frac{y(z_{i+1},t) - y(z_i,t)}{h} = \frac{y(i+1,t) - y(i,t)}{h} . \quad (11)$$

Here, a function $y(z,t)$ is continuously differentiable in $<0, L>$, and, $h = L/n$ is the diskretization step.

Dynamic Model

Applying the substitutions (10), (11) in (1) – (5) and, omitting the argument t in parenthesis, PDRs (1) – (5) are approximated by a set of ODRs in the form

$$\frac{dc_A(i)}{dt} = -\left[b_0 + k_1(i)\right] c_A(i) + b_0 c_A(i-1) \quad (12)$$

$$\frac{dc_B(i)}{dt} = k_1(i) c_A(i) - \left[b_0 + k_2(i)\right] c_B(i) + b_0 c_B(i-1) \qquad (13)$$

$$\frac{dT_r(i)}{dt} = b_1 Q_r(i) - (b_0 + b_2) T_r(i) + b_0 T_r(i-1) + b_2 T_w(i) \qquad (14)$$

$$\frac{dT_w(i)}{dt} = b_3\left[T_r(i) - T_w(i)\right] + b_4\left[T_c(i) - T_w(i)\right] \quad (15)$$

$$\frac{dT_c(m)}{dt} = -(b_5 + b_6) T_c(m) + b_5 T_c(m+1) + b_6 T_w(m) \qquad (16)$$

for $i = 1, \dots , n$ and $m = n - i + 1$, and, with initial conditions

$c_A(i,0) = c_A^s(i)$, $c_B(i,0) = c_B^s(i)$, $T_r(i,0) = T_r^s(i)$,

$T_w(i,0) = T_w^s(i)$ and $T_c(i,0) = T_c^s(i)$ for $i = 1,...,n$.

The boundary conditions enter into Eqs. (12) – (14) and (16) for $i = 1$.

Now, nonlinear functions in Eqs. (12) – (16) take the discrete form

$$k_j(i) = k_{j0} \exp\left(\frac{-E_j}{RT_r(i)}\right), \quad j = 1, 2 \qquad (17)$$

$$Q_r(i) = (-\Delta H_{r1}) k_1(i) c_A(i) + (-\Delta H_{r2}) k_2(i) c_B(i) \quad (18)$$

for $i = 1, ..., n$.

The parameters b in Eqs. (12) – (16) are calculated from formulas

$$b_0 = \frac{v_r}{h}, \quad b_1 = \frac{1}{(\rho c_p)_r}, \quad b_2 = \frac{4U_1}{d_1(\rho c_p)_r},$$

$$b_3 = \frac{4d_1 U_1}{(d_2^2 - d_1^2)(\rho c_p)_w}, \quad b_4 = \frac{4d_2 U_2}{(d_2^2 - d_1^2)(\rho c_p)_w} \qquad (19)$$

$$b_5 = \frac{v_c}{h}, \quad b_6 = \frac{4n_1 d_2 U_2}{(d_3^2 - n_1 d_2^2)(\rho c_p)_c}.$$

Here, the formula for computation of T_m (9) is rewriten to the discrete form

$$T_m(t) = \frac{1}{n} \sum_{i=1}^{n} T_r(z_i, t) \qquad (20)$$

Steady-State Model

Computation of the steady-state characteristics is necessary not only for a steady-state analysis but the steady state values $y^s(i)$ also constitute initial conditions in ODRs (12) – (16) (here, y presents some of the variable in the set (12) – (16)).

The steady-state model can simply be derived equating the time derivatives in (12) – (16) to zero. Then, after some algebraic manipulations, the steady-state model takes the form of difference equations

$$c_A^s(i) = \frac{b_0}{b_0 + k_1^s(i)} c_A^s(i-1) \qquad (21)$$

$$c_B^s(i) = \frac{1}{b_0 + k_2^s(i)} \left[k_1^s(i) c_A^s(i) + b_0 c_B^s(i-1) \right] \quad (22)$$

$$T_r^s(i) = \frac{1}{b_0 + b_2} \left[b_1 Q_r^s(i) + b_0 T_r^s(i-1) + b_2 T_w^s(i) \right] \quad (23)$$

$$T_w^s(i) = \frac{1}{b_3 + b_4} \left[b_3 T_r^s(i) + b_4 T_c^s(i) \right] \qquad (24)$$

$$T_c^s(m) = \frac{1}{b_5 + b_6} \left[b_5 T_c^s(m+1) + b_6 T_w^s(m) \right] \quad (25)$$

for $i = 1, ..., n$ and $m = n - i + 1$, and, nonlinear functions accordant with a steady-state are

$$k_j^s(i) = k_{j0} \exp\left(\frac{-E_j}{RT_r^s(i)}\right), \quad j = 1, 2 \qquad (26)$$

$$Q_r^s(i) = (-\Delta H_{r1}) k_1^s(i) c_A^s(i) + (-\Delta H_{r2}) k_2^s(i) c_B^s(i) \quad (27)$$

Here, the formula for computation T_m takes the form

$$T_m^s = \frac{1}{n} \sum_{i=1}^{n} T_r^s(z_i) \qquad (28)$$

SIMULATION RESULTS

The combinations of the inputs and outputs in all simulations of steady-state and dynamic characteristics were considered with respect to an importantance for a prospective control design.

Steady-state Characteristics

Steady-state characteristics were computed from Eqs. (21) – (27) using fixed point iterations algorithm and for $n = 200$. Typical concentration and temperature profiles along the reactor tubes computed for $c_{A0}^s = 2.85$, $c_{B0}^s = 0$, $T_{r0}^s = 323$, $T_{c0}^s = 299$, $q_c^s = 0.35$ and $q_r^s = 0.15$ are shown in Figs. 2, 3. The presence of a peak on the reactant B profile is given by considered reaction type. The existence of a maximum on the reactant temperature profile (so called *hot spot*) is a common property of many tubular reactors with exothermic reactions.

The concentration and reactant profiles for various values q_c^s are exhibited in Figs. 4 – 6. All simulation results document strong dependence of all profiles upon this input in the steady-state.

The dependences of some output variables upon q_c^s and for various q_r^s are shown in Figs. 7 – 9. The courses document strong sensitivity of outputs upon both flow rates which can be considered as control inputs. In this way obtained results are very important for the reactor control design.

The dependence of the component B concentration
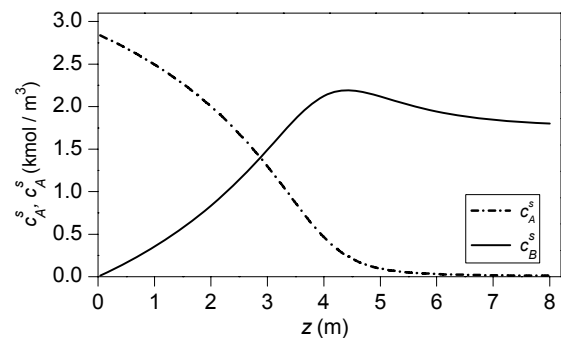
Figure 2: Concentration profiles along the reactor.

$c_{B\,\text{out}}^{s}$ on the reactant mean temperature is presented in Fig. 10. Also this result can be important when the output concentration c_B is non-measurable and, its desired (maximum) value could be achieved only by measured reactant temperatures along the reactor.
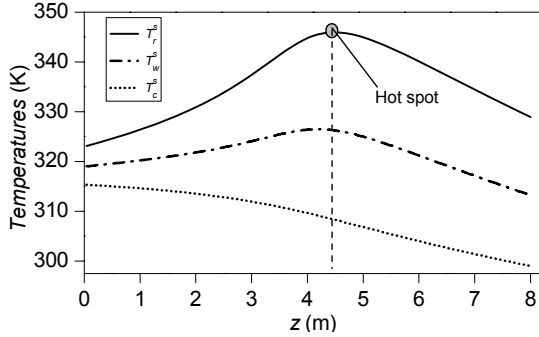
Figure 3: Temperature profiles along the reactor.
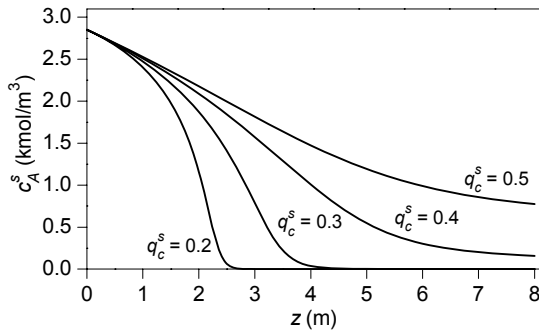
Figure 4: Concentration c_A^{s} profiles for various q_c^{s}.
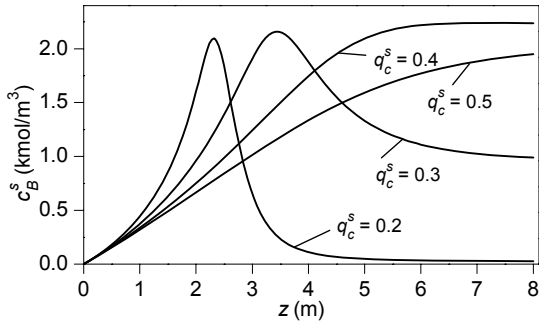
Figure 5: Concentration c_B^{s} profiles for various q_c^{s}.
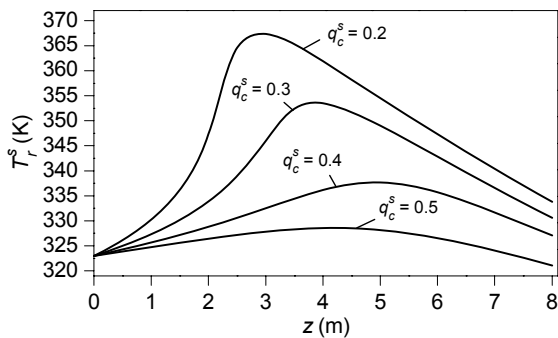
Figure 6: Reactant temperature profiles for various q_c^{s}.
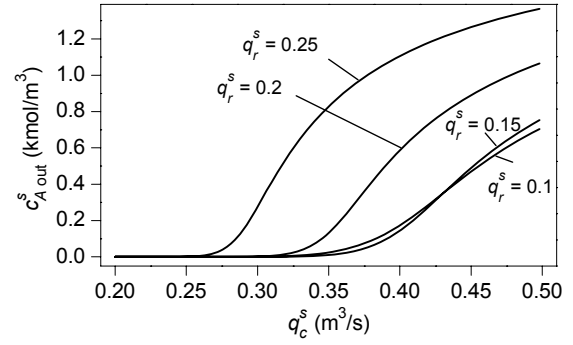
Figure 7: Dependence of $c_{A\,\text{out}}^{s}$ on q_c^{s} for various q_r^{s}.
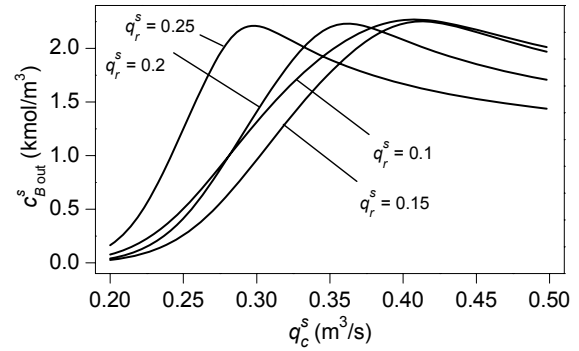
Figure 8: Dependence of $c_{B\,\text{out}}^{s}$ on q_c^{s} for various q_r^{s}.
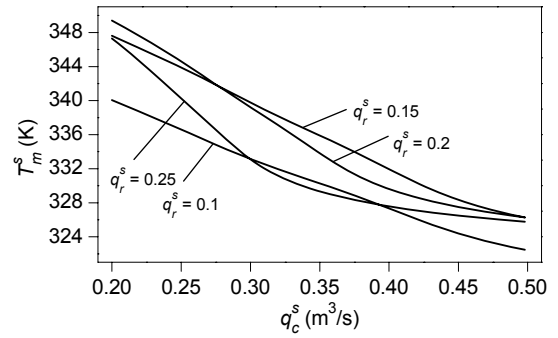
Figure 9: Dependence of T_m^{s} on q_c^{s} for various q_r^{s}.
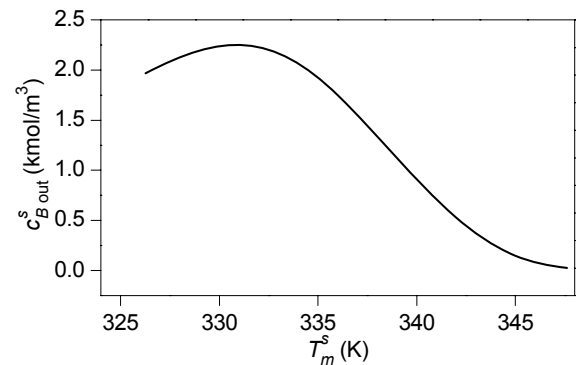
Figure 10: Dependence of c_B^{s} upon T_m^{s}.

Dynamic Characteristics

All dynamic charakteristics were computed using the Runge-Kutta method with a fixed step. All inputs and outputs were considered as deviations from their steady values. This form is frequently used in the control. The deviations are denoted as follows:

$$v_1(t) = q_c(t) - q_c^s$$

$$v_2(t) = c_{A0}(t) - c_{A0}^s$$

$$v_3(t) = T_{r0}(t) - T_{r0}^s$$

$$y_1(t) = c_{Bout}(t) - c_{Bout}^s$$

$$y_2(t) = T_m(t) - T_m^s$$

where $c_{Bout}^s = 1.345$ and $T_m^s = 337.77$ are the output steady-state values computed from the model (21) – (25) for the input steady-state values

$q_c^s = 0.25$, $q_r^s = 0.15$ $c_{A0}^s = 2.85$, $T_{r0}^s = 323$ and $T_{c0}^s = 299$.

The responses to q_c and c_{A0} step changes in Figs. 11-14 clearly document a strong nonlinearity of the process. Moreover, they show a better applicability of the reactant mean temperature as the controlled output in comparison with the output concentration c_B having in this regard very unfavourable properties.

As an influence illustration of the random disturbances on the process output, the responses to random signals c_{A0} and T_{r0} are shown in Figs. 15 – 18. These simulation results document high sensitivity of both outputs to random input signals.
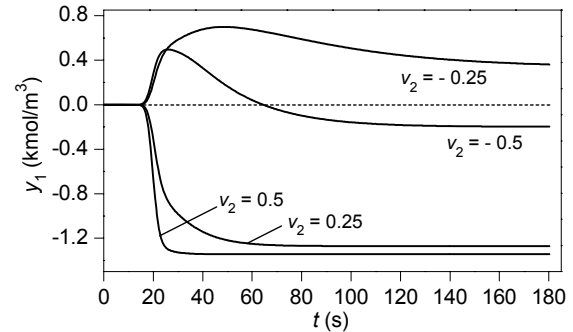
Figure 13: Output concentracion c_B step responses.
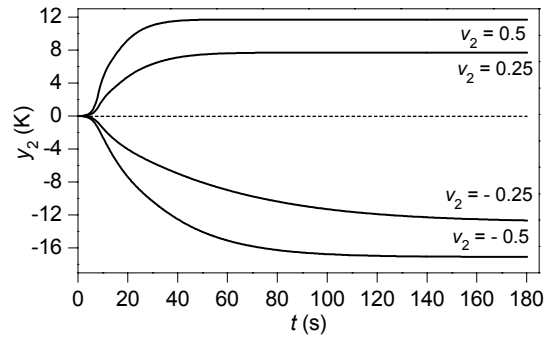
Figure 14: Mean temperature step responses.
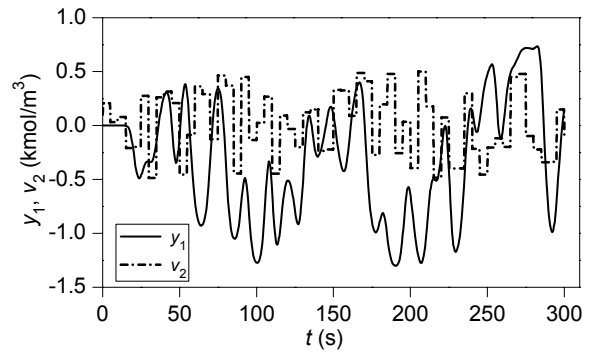
Figure 15: Output c_B response to random input v_2.

Figure 11: Output concentracion c_B step responses.

Figure 12: Mean temperature T_m step responses.
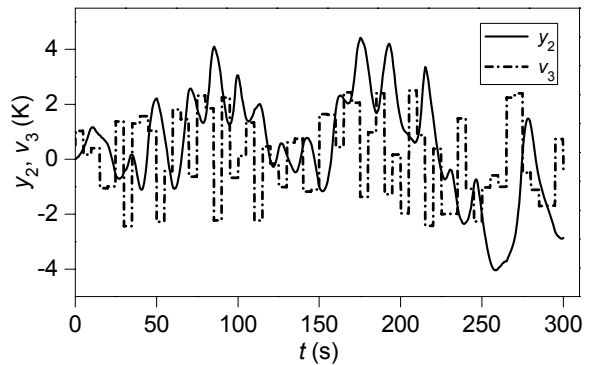
Figure 16: Output T_m responses to random input v_3.
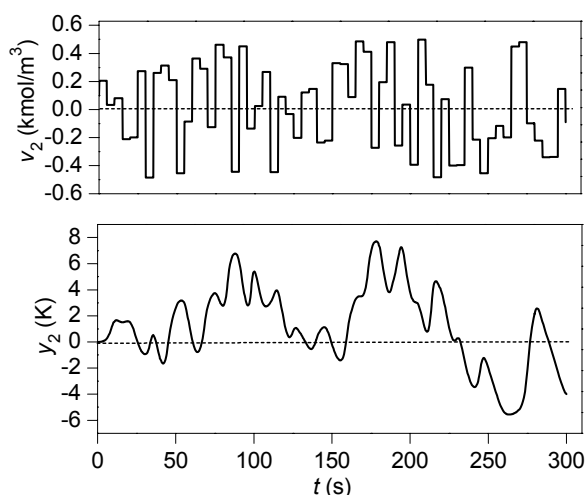
Figure 17: Output T_m responses to random input v_2.
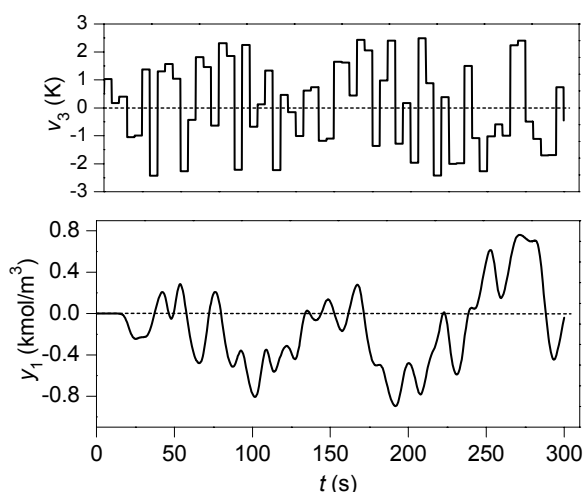
Figure 18: Output concentration c_B response to random input v_3.

CONCLUSIONS

In the paper, the mathematical model of a tubular chemical reactor with a consecutive exothermic reaction has been presented. The computer models for simulations of steady-state and dynamic characteristics in the form of sets of ordinary differential and difference equations were derived to an original model in the form of partial differential equations The simulation experiments were chosen with a view to a prospective control of the process.

ACKNOWLEDGMENTS

REFERENCES

Luyben, W.L. 1989. *Process modelling, simulation and control for chemical engineers*. McGraw-Hill, New York.

Ingham, J.; I. J. Dunn; E. Heinzle; and J. E. Přenosil. 1994. *Chemical Engineering Dynamics. Modelling with PC Simulation*. VCH Verlagsgesellschaft, Weinheim.

Severance, F.L. 2001. *System Modeling and Simulation*. Wiley, Chichester.

Babu, B.V. 2004. *Process Plant Simulation*. Oxford University Press, New Delhi.

Ogunnaike, B.A., Ray, W.H.: *Process dynamics, modeling, and control*. Oxford University Press, New York, 1994.

Seborg, D.E; T.F. Edgar; and D.A. Mellichamp. 1989. *Process Dynamics and Control*. Wiley, Chichester.

Marlin, T.E.: *Process control. Designing processes and control systems for dynamic performance*. McGraw-Hill, New York, 1995.

Corriou, J.-P. 2004. *Process Control. Theory and Applications*. Springer – Verlag, London.

AUTHOR BIOGRAPHIES

PETR DOSTÁL was born in Kněždub, Czech Republic in 1945. He studied at the Technical University of Pardubice, where he obtained his master degree in 1968 and PhD. degree in Technical Cybernetics in 1979. In the year 2000 he became professor in Process Control. He is now head of the Department of Process Control, Faculty of Applied Informatics of the Tomas Bata University in Zlín. His research interest are modeling and simulation of continuous-time chemical processes, polynomial methods, optimal and adaptive control. You can contact him on email address dostalp@fai.utb.cz

VLADIMÍR BOBÁL was born in Slavičín, Czech Republic in 1942. He graduated in 1966 from the Brno University of Technology. He received his Ph.D. degree in Technical Cybernetics at Institute of Technical Cybernetics, Slovak Academy of Sciences, Bratislava, Slovak Republic. He is now Professor in the Department of Process Control, Faculty of Applied Informatics of the Tomas Bata University in Zlín. His research interests are adaptive control systems, system identification and CAD for self-tuning controllers. You can contact him on email address bobal@fai.utb.cz

JIŘÍ VOJTĚŠEK was born in Zlin, Czech Republic and studied at the Tomas Bata University in Zlin, where he got his master degree in chemical and process engineering in 2002 and finished his Ph.D. in Technical Cybernetics in 2007. He works as a senior lecturer in the Department of Process Control, Faculty of Applied Informatics, Tomas Bata University in Zlin. You can contact him on email address vojtesek@fai.utb.cz.

LIBRARY FOR DESIGN AND SIMULATION VERIFICATION OF SELF-TUNING CONTROLLERS

Vladimír Bobál, Petr Chalupa and Petr Dostál
Department of Process Control, Faculty of Applied Informatics
Tomas Bata University in Zlín
Nad Stráněmi 4511, 760 05 Zlín, Czech Republic
E-mail:bobal@fai.utb.cz

KEYWORDS

Self-tuning control, ARX model, recursive least squares, PID control, polynomial approach, MATLAB–Simulink, simulation verification

ABSTRACT

This contribution present structure and usage of Self-tuning Controllers Simulink Library (STCSL). The STCSL was created for design, simulation verification and especially real-time implementation of single input - single output (SISO) digital self-tuning controllers. The proposed adaptive controllers which are included in library can be divided into three groups. The first group covers PID adaptive algorithms using traditional Ziegler-Nichols method for setting controller parameters, the second group of described controllers is based on the polynomial approach and the third group contains the controllers derived on the other approaches (minimum variance etc.). The controllers are implemented as an encapsulated Simulink blocks and thus allows users simple integration into existing Simulink schemas. The process of developing real-time applications using MATLAB Real-Time Workshop and several control courses is also presented. This Library is very successfully used in Adaptive Control Course in education practice for design and verification of self-tuning control systems in simulation and real-time conditions. It is suitable also for design and verification of the industrial digital controllers. The STCSL is available free of charge at Internet site - **http://www.utb.cz/stctool/**.

INTRODUCTION

Self-tuning controllers belong by their character to the class of adaptive control systems. The main aim of the adaptive control approach is to solve the control problem in cases where the characteristics of a controlled system are unknown or time variable. The basic principle of the adaptive control system is to change the controller characteristics based on the characteristics of control process. Self-tuning controllers use the combination of the recursive process identification on base of a selected model process and the controller synthesis based on knowledge of parameter estimates of controlled process.

The general task of optimal adaptive control with on-line process identification is very complicated and thus the method of **the forced separation of the identification and the control** is often used for the design of self-tuning controllers. The principle of this simplification is in the cyclic repeating of the following steps:

1. The process parameters are assumed to be known for the current control step and equal to their current estimations.
2. The control strategy for selected criterion of control quality is designed based on the previous assumption and controller output is calculated.
3. The next identification step is performed after the obtaining the new sample of the controlled variable (eventually the external measured disturbance) – the parameters of controlled process model are recomputed using a recursive identification algorithm.

The aim of this contribution is also to inform potential users about Self-tuning Controllers Simulink Library - STCSL (see Bobál and Chalupa, 2002) and to help for practical usage in the simulation and real time conditions. The individual self-tuning algorithms are introduced in the brief form in User's Guide that is attached into the STCSL. This library can be also the suitable bridge between theory and practice by presenting some digital controller algorithms in a form acceptable for industrial users. The theoretical background of self-tuning controllers which are contained in the STCSL is given in Bobál, *et al.* (1999, 2005).

All the explicit self-tuning controllers that are included into STCSL have been algorithmically modified in the form of mathematical relations or as flow diagrams so as to make them easy to program and apply. Some are original algorithms based on a modified Ziegler-Nichols criterion, others have been culled from publications and adapted to make them more accessible to the user.

RECURSIVE IDENTIFICATION

The regression (ARX) model of the following form

$$y(k) = \boldsymbol{\Theta}^T(k)\boldsymbol{\Phi}(k-1) + n(k) \qquad (1)$$

is used in the identification part of the designed controller algorithms, where

$$\boldsymbol{\Theta}^T(k) = \left[a_1, a_2, ..., a_{na}, b_1, b_{2,}, ..., b_{nb} \right] \quad (2)$$

is the vector of the parameters and

$$\boldsymbol{\Phi}^T(k-1) = \left[-y(k-1), -y(k-2), ..., -y(k-na), \right.$$
$$\left. u(k-1), u(k-2), ..., u(k-nb) \right]$$
$$(3)$$

is the regression vector ($y(k)$ is the process output variable, $u(k)$ is the controller output variable). The non-measurable random component $n(k)$ is assumed to have zero mean value $E[n(k)] = 0$ and constant covariance (dispersion) $R = E[n^2(k)]$.

The recursive least squares method for calculating of parameter estimates $\hat{\boldsymbol{\Theta}}(k)$ is utilized. Using the pure least squares method, the influence of all pairs of identified system inputs and outputs to the parameters estimates is the same. This property can be inconvenient for example when identifying the system with time-varying parameters. In this case, it is better to use least squares method with exponential forgetting where the influence of latter data to the calculation of the parameter estimates is greater then the influence of older data.

The exponential forgetting method can be further improved by adaptive directional forgetting (Kulhavý, 1987) which changes forgetting coefficient with respect to changes of input and output signal.

CONTROLLER ALGORITHMS

The proposed self-tuning controllers which are included in the library are divided into three groups:

- classical digital PI and PID controllers whose tuning is based on Ziegler-Nichols method, pole assignment method and its modifications,
- controllers based on polynomial approach (dead-beat strong and weak version, pole assignment, controllers based on minimization of quadratic criterion),
- controllers based on other approaches (minimum variance controllers, Dahlin's controller, Bányász-Keviczky's controller etc.).

Algorithms of digital PID Ziegler-Nichols controllers

The PID controllers are still widely used in industry. These types of controllers are more convenient for users due to their simple implementation, which is generally well known. Provided the controller parameters are well chosen they can control a considerable part of continuous technological processes. To get a digital version of the PID controller, it is necessary to discretize the integral and derivative component of the continuous-time controller. For discretizing the integral component we usually employ the forward rectangular method (FRM), backward rectangular method (BRM) or trapezoidal method (TRM). The derivative component is mostly replaced by the 1st order difference (two-point difference). The recurrent control algorithms which compute the actual value of the controller output $u(k)$ from the previous value $u(k\text{-}1)$ and from compensation increment seem to be suitable for practical use

$$u(k) = q_0 e(k) + q_1 e(k-1) + q_2 e(k-2) + u(k-1) \quad (4)$$

where q_0, q_1, q_2 are the controller parameters. As a matter of fact, the advantage of algorithm (4) is that it is not necessary to storage last input and output data in the computer storage.

It is subsequently possible to derive further variants of digital PID controllers. First group of proposed self-tuning PID controllers is based on the classical Ziegler and Nichols (1942) method. In this well-known approach the parameters of the controller are calculated from the ultimate (critical) gain K_{pu} and the ultimate period of oscillations T_u of the closed loop system. The analytical expressions for computing of these critical parameters are derived in Bobál, *et al.* (1999, 2005).

Algorithms of PID pole placement controllers

A controller based on the pole placement method in a closed feedback control loop is designed to stabilise the closed control loop whilst the characteristic polynomial should have a previously determined pole. It is possible to express the digital controllers in the form of a discrete transfer function

$$G_R(z) = \frac{U(z)}{E(z)} = \frac{Q(z^{-1})}{P(z^{-1})} \quad (5)$$

Let the controlled process be given by the transfer function

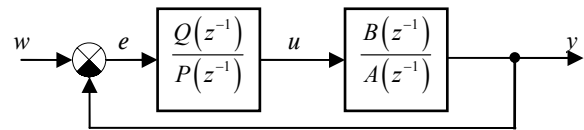$$G_P(z) = \frac{Y(z)}{U(z)} = \frac{B(z^{-1})}{A(z^{-1})} \quad (6)$$

Figure 1: Control loop with PID-A controller

PID - A controller structure. This controller structure is shown in Fig. 1, then the transfer function of closed loop circuit is

$$G_W(z) = \frac{Y(z^{-1})}{W(z^{-1})} = \frac{B(z^{-1})Q(z^{-1})}{A(z^{-1})P(z^{-1}) + B(z^{-1})Q(z^{-1})} \quad (7)$$

and the characteristic polynomial of the closed-loop system with a PID-A controller is in the form

$$A(z^{-1})P(z^{-1}) + B(z^{-1})Q(z^{-1}) = D(z^{-1}) \quad (8)$$

The pole placement of characteristic polynomial (8) determines the dynamic behaviour of the closed - loop system. The characteristic polynomial $D(z^{-1})$ can be specified by different methods.

PID - B controller structure. The structure of the control loop with controller PID-B is shown in Fig. 2). The characteristic polynomial has in this case form
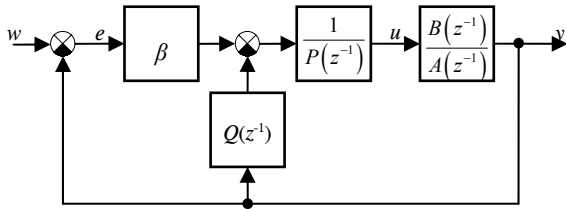
Figure 2: Control loop with PID-B controller

$$A(z^{-1})P(z^{-1}) + B(z^{-1})\left[Q(z^{-1}) + \beta\right] = D(z^{-1}) \quad (9)$$

where β is feed forward controller parameter. PID – B controller structure is in principle 2DOF (two-degree freedom) configuration.

Self-tuning controllers based on polynomial approach

Different forms of self-tuning controllers could be suggested according to the chosen type of a plant model (and, consequently according to the chosen identification method), according to a criterion of quality, according to a mathematical procedure during deriving of controller equations, etc. Next group of described controllers is based on the polynomial approach. These algorithms succeed various criterions for a control process course – dead-beat method, pole assignment method and LQ (linear-quadratic) control method. The design of these controllers results from the basic block diagram of 2DOF configuration (see Fig. 3). From Fig. 3 it is obvious that the controller equation has the form

$$P(z^{-1})K(z^{-1})u(k) = R(z^{-1})w(k) - Q(z^{-1})y(k) \quad (10)$$

where $K(z^{-1}) = 1 - z^{-1}$ is integrator.

Then the transfer function of closed loop 2DOF circuit is

$$G_W(z) = \frac{Y(z^{-1})}{W(z^{-1})} = \frac{R(z^{-1})B(z^{-1})}{A(z^{-1})K(z^{-1})P(z^{-1}) + B(z^{-1})Q(z^{-1})} \quad (11)$$
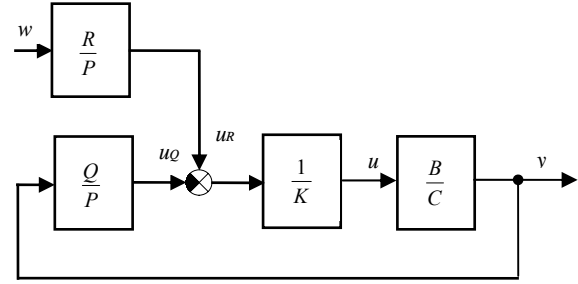
Figure 3: Closed loop 2DOF control system

The special case is a controller 1DOF (one degree of freedom - see Fig. 1)), which is valid for $R(z^{-1}) = Q(z^{-1})$ and which works with a tracking error $e(k) = w(k) - y(k)$. It could be proved that such a controller is only suboptimal for tasks of a reference value tracking.

The process of computing coefficients of polynomials $Q(z^{-1})$ and $P(z^{-1})$ is similar to 1DOF case. The presence of feedforward part brings the possibility of simpler processing of reference signal. If the 1DOF and 2DOF controllers have to fulfil the same requirements when controlling the same system, the degrees of polynomials in 1DOF control are usually higher then in 2DOF control.

SELF-TUNING CONTROLLERS SIMULINK LIBRARY

The Simulink is nowadays a word-wide standard in simulation, testing, and verification of behaviour of various dynamic systems. Simulink is a part of MATLAB system and supports linear or nonlinear systems modelled in continuous time, sampled time or a hybrid of the two. Systems can also be multirate, i.e. have different parts that are sampled or updated at different rates.

Based on monograph Bobál, et al. (2005) a library of self-tuning controllers in MATLAB/Simulink environment was created. The purpose was to create a framework suitable for creating and testing of self-tuning controllers. The library is available free of charge at internet site of Tomas Bata University in Zlín – www.utb.cz/stctool/ (see Bobál and Chalupa, 2002). The library was created using MATLAB version 6.5 (Release 13), but it can be ported with some changes to

lower MATLAB versions. Controllers are implemented in the library as standalone Simulink blocks, which allow an easy incorporation into existing simulation schemes and an easy creation of new simulation circuits. Only standard techniques of Simulink environment were used when creating the controller blocks and thus just basic knowledge of this environment is required for the start of work with the library. Controllers can be implemented to simulation schemes just by the copy or drag & drop operation and their parameters are set using dialog windows. Another advantage of used approach is a relatively easy implementation of user-defined controllers by modifying some suitable controller in the library.

Nowadays the library contains over 30 simple single input single output discrete self-tuning controllers, which use discrete ARX models of second and third order for the on-line system identification. All of these controllers use discrete control laws, where controller parameters are computed by various methods. Many methods calculate controller parameters on base of the ultimate parameters of controlled system (Ziegler-Nichols approach) or on base of pole placement approach. The library package contains not only the controllers but also reference manual with simple description of the algorithm and the internal structure of each controller.
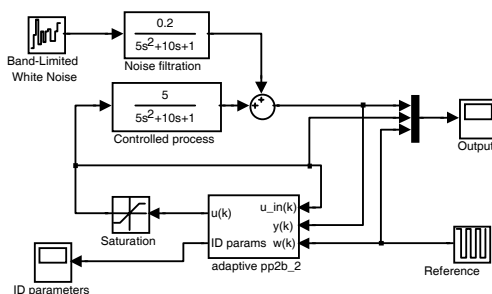
Figure 4: Control circuit in Simulink environment

The typical wiring of any library controller is shown in Fig. 4. Each self-tuning controller from the library uses 3 input signals and provides 2 outputs. The inputs are the reference signal (w) and the actual output of controlled process (y). The last controller input is the current input of controlled process – the control signal (u_in). The value of this signal does not have to be the same as controller output e.g. due to the saturation of controller output. The main controller output is, of course, control signal – the input signal of the controlled process. The second controller output consists of the current parameter estimates of the controlled process model. The number of parameters this output consists of depends on the model used by on-line identification.

A scheme analogous to the scheme in Fig. 4 can also be used to simulate both discrete and a continuous-time

controlled process with much more complicated structure. It is possible to implement processes with time-varying parameters, processes described by non-linear differential equations etc.

The dynamic behaviour of a controller can be influenced by changing the parameters available for a given controller. Controller parameters can be divided into two groups:

• parameters common to all controllers,
• controller specific parameters.

All controller parameters are set or changed by standard approaches of the Simulink environment, that is, by changing items in the dialogue window - invoked for example, by double-clicking with the mouse on the controller object. This dialogue window also contains a short description of the corresponding controller; and the "Help" button is used to invoke the corresponding part of the hypertext reference guide. The parameter setting dialog window of the *pp2a_1* controller is shown in Fig. 5.

Figure 5: Dialogue for *pp2a_1* Controller Setting Parameters

Each library controller is constructed as a mask of a subsystem, which consists of Simulink blocks and has inputs, outputs and parameters. Internal controller structure consists of Simulink blocks which provide, among others, the possibility of easy creation of a new

controller by a modification of some suitable library controller. The structure of controller *pp2a_1* is presented in Fig. 6 as an example. Each library controller consists of three basic parts:

- on-line identification block,
- block computing controller parameters,
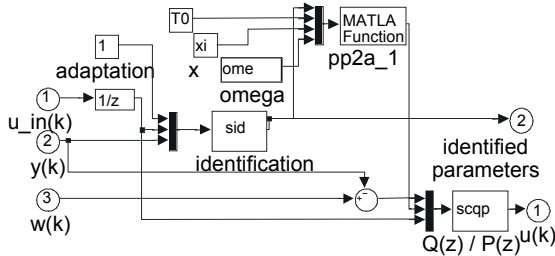- block computing controller output.

Figure 6: Block scheme of PP2A_1 controller

SIMULATION EXAMPLES IN THE SIMULINK ENVROMENT

Simulation is a useful tool for the synthesis of control systems, allowing one not only to create mathematical models of a process but also to design virtual controllers in a computer. The mathematical models provided are sufficiently close to a real object that simulation can be used to verify the dynamic characteristics of control loops when the structure or parameters of the controller change. The models of the processes may also be excited by various random noise generators which can simulate the stochastic characteristics of the processes noise signals with similar properties to disturbance signals measured in the machinery. Simulation results are valuable for implementation of a chosen controller (control algorithm) under laboratory and industrial conditions. It must be borne in mind, however, that the practical application of a controller verified by simulation cannot be taken as a routine event. Obviously simulation and laboratory conditions can be quite different from those in real plants, and therefore one must verify its practicability with regard to process dynamics and the required standard of control quality (for example maximum allowable overshoot, accuracy, settling time, etc.).

Simulation experiments of the control of some digital process models will be shown. Only higher order models which have been approximated by second- or third-order models in the identification procedure have been chosen for simulation verification. Simulation verification has been limited to digital PID controllers based on the Ziegler-Nichols and the pole assignment methods, which have proved to be the best in practical applications. A scheme of the control circuit used for simulation in MATLAB-Simulink environment is shown in Fig. 7.

As an example of verification by computer simulation a fourth-order system with the transfer function

$$G(s) = \frac{1}{(s+1)^4} \qquad (12)$$
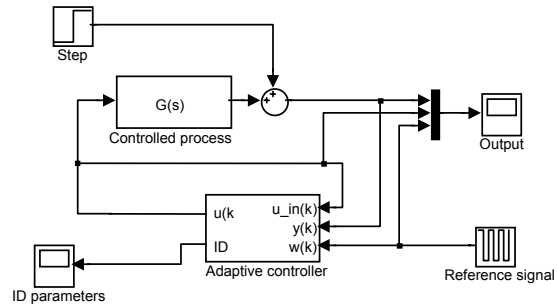
Figure 7: Control circuit used for simulation

was used. The simulation verification was realized using four types of the controllers included in the Self-tuning Controllers Simulink Library (Bobál and Chalupa 2002). In time interval $t = 150 - 200\,\text{s}$ a constant disturbance $v(k) = 1$ was injected on the system output. The controller output value $u(k)$ was limited within the range $\langle 0; 2 \rangle$. The initial value of the directional forgetting factor was chosen as $\varphi(0) = 1$. The initial values of the model parameter estimates are $\hat{\boldsymbol{\Theta}}^T(0) = [0.1, 0.2, 0.3, 0.4]$ for a second-order model, and $\hat{\boldsymbol{\Theta}}^T(0) = [0.1, 0.2, 0.3, 0.4, 0.5, 0.6]$ for a third-order model.

The control loop was simulated using a PID Takahashi controller (*zn2tak*) – see (Bobál and Chalupa, 2002, Bobál *et al*., 2005) with equation

$$u(k) = K_P \{ -y(k) + y(k-1) + \frac{T_0}{T_I}[w(k) - y(k)] + \\ + [2y(k-1) - y(k) - (k-2)] \} + u(k-1) \qquad (13)$$

where

$$K_P = 0.6 K_{Pu}\left(1 - \frac{T_0}{T_u}\right); \quad T_I = \frac{K_P T_u}{1.2 K_{Pu}}; \quad T_D = \frac{3 K_{Pu} T_u}{40 K_P} \qquad (14)$$

The sampling period was chosen $T_0 = 1$ s, critical parameters (proportional gain K_{Pu} and period of oscillations T_u) were computed for the identification of a second-order model. Fig. 8 illustrates the simulation control performance using controller (13), (14).

Fig. 9 illustrates the simulation control performance using controller (13), (14) (*zn3tak*) – see (Bobál and Chalupa, 2002, Bobál *et al*., 2005), the sampling period

was chosen $T_0 = 1.5$ s. Critical parameters were computed for the identification of a third-order model. Fig. 10 illustrates the simulation control performance using pole placement controller *pp2a_1* – see (Bobál and Chalupa, 2003, Bobál *et al*., 2005). The idea of this controller is to make the dynamic behaviour of the closed loop similar to that of the second order continuous system with characteristic polynomial (8) as stated by equation

$$D(s) = s^2 + 2\xi\omega_n s + \omega_n^2 \qquad (15)$$
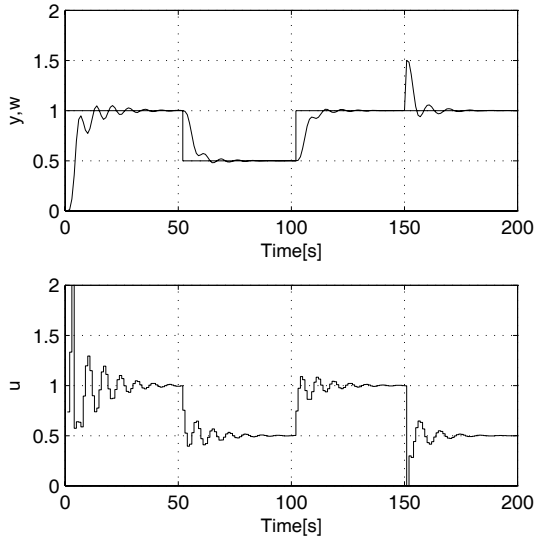
Figure 8: Simulation control performance using a PID Takahashi controller (*zn2tak*) for the identification of a second-order model
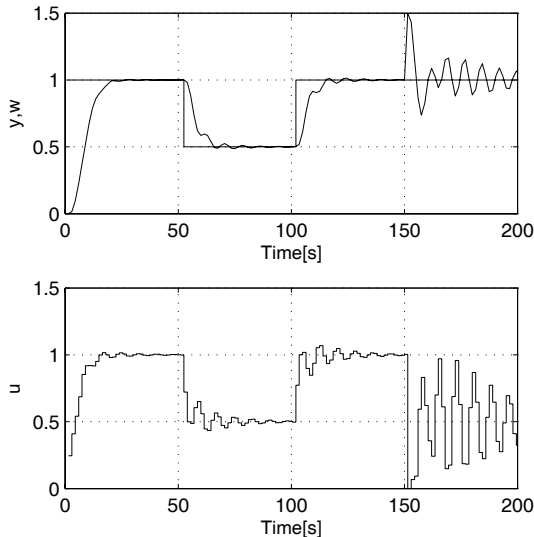
Figure 9: Simulation control performance using a PID Takahashi controller (*zn3tak*) for the identification of a third-order model

Structure of control circuit with this controller is shown in Fig 1and the control law is given by equation

$$u(k) = q_0 e(k) + q_1 e(k-1) + q_2 e(k-2) + (1-\gamma)u(k-1) + \gamma u(k-2) \qquad (16)$$

where q_0, q_1, q_2 and γ are controller parameters. Additional parameters were set to $T_0=2.5$ s, $\xi=1$ and $\omega_n=0.3$, which leads to asymptotic step responses.
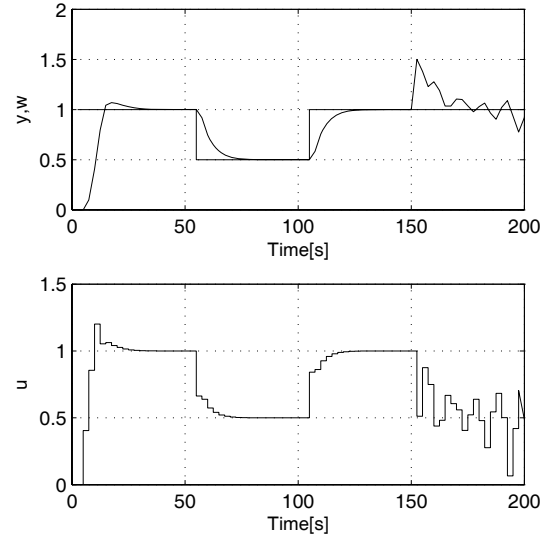
Figure: 10 Simulation control performance using a pole placement controller PID-1 (*pp2a_1*)

From Fig. 8 – 10 it is evident that the response of the controlled variable to step change of the reference signal in the control loop with the *pp2a_1* controller is aperiodical. In the control loop with the controller *zn3tak* occur negligible oscillations of particular variables. The controller *zn2tak* causes light stable oscillations in the closed loop. On the contrary, the best ability to eliminate step disturbances proved the controller *zn2tak*. In control loops with the controllers *zn3tak* and *pp2a_1* step disturbances cause oscillations of the variables.

CREATING APPLICATIONS WITH REAL - TIME WORKSHOP

The MATLAB/Simulink environment can also be used to generate code to be used in controllers in industrial practice. *Real-Time Workshop*, one of the toolboxes shipped with MATLAB, allows generating of source code and programs to be used outside the MATLAB environment. The process of generating the source code is controlled by special compiler files that are interpreted by *Target Language Compiler*. These files are identified by the *.tlc* (target language compiler) extension and describe how to convert Simulink schemes to target language. Thereby, source code is generated and after compiling and linking, the resulting application is created. Applications for various microprocessors and operating systems can be created by selecting corresponding *.tlc* files.

The *Target Language Compiler* can create applications to be used under the Windows environment, which

perform control algorithm and save the results in a binary file with a structure acceptable to the MATLAB. An analysis of the control process can then be performed using the advantages of MATLAB functions and commands. Selecting another *.tlc* file leads to creation of an MS-DOS application, or an application to be used on PC - based industrial computers without the requirement for an operating system.

Many manufactures of industrial computers and controllers have created their own target language compiler files used to create applications for the equipment they produce. *Real-Time Workshop* provides a relatively open environment for the conversion of block schemes to various platforms, where any users can create their own *.tlc* files for converting the block scheme to a source code and hence, achieve compatibility with any hardware.

Application creation only requires selection of the appropriate *.tlc* file, or eventually, setting the compiler parameters and then initiation of the compilation process.

CONCLUSIONS

The Self-tuning Controllers Simulink Library is used in university course of Adaptive Control Systems. Its architecture enables an easy user orientation in Simulink block schemes and generates source code for controllers' functions. The controllers provided are suitable for modification and thereby implementation of user-defined controllers. The compatibility with *Real-Time Workshop* ensures not only the possibility of laboratory testing using real time models but also the possibility of creating applications for industrial controllers.

ACKNOWLEDGEMENTS

REFERENCES

Bobál, V., J. Böhm, J. Fessl and J. Macháček. 2005. *Practical Aspects of Self-tuning Controllers: Algorithms,Implementation and Applications*. Springer-Verlag, London.

Bobál, V., J. Böhm, and R. Prokop. 1999. "Practical Aspects of Self-tuning Controllers". *International Journal of Adaptive Control and Signal Processing*, 13, 671-690.

Bobál, V. and P. Chalupa. 2002. "Self-tuning Controllers Simulink Library". http: //www.utb.cz/stctool/.

Kučera, V. 1979. *Discrete Linear Control: The Polynomial Equation Approach*. John Wiley, Chichester.

Kučera, V. 1991. *Analysis and Design of Discrete Linear Control Systems*. Prentice Hall, London.

Kulhavý, R. 1987. "Restricted Exponential Forgetting in Real Time Identification". *Automatica*, 23, 586-600.

Ziegler, J. G. and N. B. Nichols. 1942. "Optimum Settings for Automatic Controllers". *Trans. ASME*, 64, 759 – 768.

AUTHOR BIOGRAPHIES

VLADIMÍR BOBÁL was born in Slavičín, Czech Republic in 1942. He graduated in 1966 from the Brno University of Technology. He received his Ph.D. degree in Technical Cybernetics at Institute of Technical Cybernetics, Slovak Academy of Sciences, Bratislava, Slovak Republic. He is now Professor in the Department of Process Control, Faculty of Applied Informatics of the Tomas Bata University in Zlín. His research interests are adaptive control systems, system identification and CAD for self-tuning controllers. You can contact him on email address `bobal@fai.utb.cz`

PETR CHALUPA was born in Zlin, Czech Republic in 1976. He graduated from Brno University of Technology in 1999. He obtained his Ph.D. in Technical Cybernetics at Tomas Bata University in Zlin in 2003. Nowadays he works as a researcher at Centre of Applied Cybernetics at Tomas Bata University in Zlin. His research interests are adaptive and predictive control of real-time systems. You can contact him on email address `chalupa@fai.utb.cz`

PETR DOSTÁL was born in Kněždub, Czech Republic in 1945. He studied at the Technical University of Pardubice, where he obtained his master degree in 1968 and PhD. degree in Technical Cybernetics in 1979. In the year 2000 he became professor in Process Control. He is now head of the Department of Process Control, Faculty of Applied Informatics of the Tomas Bata University in Zlín. His research interests are modelling and simulation of continuous-time chemical processes, polynomial methods, optimal and adaptive control. You can contact him on email address `dostalp@fai.utb.cz`

MINIMUM ENERGY CONTROL PROBLEM OF POSITIVE FRACTIONAL DISCRETE-TIME SYSTEMS

Tadeusz Kaczorek
Faculty of Electrical Engineering
Białystok Technical University
Wiejska 45D, 15-351 Białystok, Poland
E-mail: kaczorek@isep.pw.edu.pl

KEYWORDS

Fractional, positive discrete-time, linear system, minimum energy control, reachability.

ABSTRACT

The minimum energy control problem of positive fractional-discrete time linear systems is addressed. Necessary and sufficient conditions for the reachability of the system are established. Sufficient conditions for the solvability of the minimum energy control of the positive fractional discrete-time systems are given. A procedure for computation of the optimal sequence of inputs minimizing the quadratic performance index is proposed.

INTRODUCTION

In positive systems inputs, state variables and outputs take only non-negative values. Examples of positive systems are industrial processes involving chemical reactors, heat exchangers and distillation columns, storage systems, compartmental systems, water and atmospheric pollution models. A variety of models having positive linear systems behaviour can be found in engineering, management science, economics, social sciences, biology and medicine, etc.

Positive linear systems are defined on cones and not on linear spaces. Therefore, the theory of positive systems is more complicated and less advanced. An overview of state of the art in standard positive systems is given in the monographs (Farina and Rinaldi 2000; Kaczorek 2002). The realization problem for positive standard and singular continuous-time systems with delays was formulated and solved in (Kaczorek 2007c, 2007d). The reachability, controllability and minimum energy control of positive linear discrete-time systems with time-delays have been considered in (Busłowicz and Kaczorek 2004; Xie and Wang 2003). The realization problem for cone systems has been addressed in (Kaczorek 2006). The reachability and controllability to zero of positive fractional linear systems have been investigated in (Kaczorek 2008a, 2007a, 2007b, 2008b; Klamka 2002; Klamka 2005). Mathematical fundamentals of fractional calculus are given in the monographs (Miller and Ross 1993; Nishimoto 1984; Oldham and Spanier 1974; Oustalup 1993; Podlubny 1999). The fractional order controllers have been developed in (Oustalup 1993). A generalization of the Kalman filter for fractional order systems has been proposed in (Sierociuk and Dzieliński 2006). Some other applications of fractional order systems can be found in (Ferreira and Machado 2003; Moshrefi-Torbati and Hammond 1998; Ortigueira 1997; Ostalczyk 2000, 2004a, 2004b; Podlubny 2002; Samko et al. 1993; Vinagre et al. 2002; Vinagre and Feliu 2002; Gałkowski and Kummert 2005).

The minimum energy control problem has been solved for different classes of linear systems in (Klamka 1991, 1976, 1983; Kaczorek and Klamka 1986).

In this paper the minimum energy control problem will be addressed for positive fractional discrete-time linear systems.

The paper is organized as follows. In section 2 the solution of the state equation and the necessary and sufficient conditions for the positivity of the fractional systems are recalled. Necessary and sufficient conditions for the reachability of the positive fractional systems are established in section 3. The main result of the paper is presented in section 4 in which the minimum energy control problem is formulated and solved. Concluding remarks are given in section 5.

To the best knowledge of the author the minimum energy control problem for the positive fractional discrete-time linear systems have not been considered yet.

POSITIVE FRACTIONAL SYSTEMS

Let $\Re^{n \times m}$ be the set of $n \times m$ real matrices and $\Re^n := \Re^{n \times 1}$. The set of $m \times n$ matrices with nonnegative entries will be denoted by $\Re_+^{m \times n}$ and $\Re_+^n := \Re_+^{n \times 1}$. The set of nonnegative integers will be denoted by Z_+ and the $n \times n$ identity matrix by I_n.

In this paper definition of the fractional difference of the form (Kaczorek 2007a)

$$\Delta^\alpha x_k = \sum_{j=0}^{k} (-1)^j \binom{\alpha}{j} x_{k-j}, \qquad (1)$$

$$n-1 < \alpha < n \in N = \{1, 2, ...\}, k \in Z_+$$

will be used, where $\alpha \in R$ is the order of the fractional difference and

$$\begin{pmatrix} \alpha \\ j \end{pmatrix} = \begin{cases} 1 & \text{for } j = 0 \\ \dfrac{\alpha(\alpha-1)\cdots(\alpha-j+1)}{j!} & \text{for } j = 1,2,\ldots \end{cases} \quad (2)$$

Consider the fractional discrete linear system, described by the state-space equations

$$\Delta^{\alpha} x_{k+1} = A x_k + B u_k, \quad k \in Z_+ \quad (3a)$$

$$y_k = C x_k + D u_k \quad (3b)$$

where $x_k \in \Re^n$, $u_k \in \Re^m$, $y_k \in \Re^p$ are the state, input and output vectors and $A \in \Re^{n \times n}$, $B \in \Re^{n \times m}$, $C \in \Re^{p \times n}$, $D \in \Re^{p \times m}$.

Using definition (1) we may write the equations (3) in the form

$$x_{k+1} + \sum_{j=1}^{k+1} (-1)^j \begin{pmatrix} \alpha \\ j \end{pmatrix} x_{k-j+1} = A x_k + B u_k, \quad k \in Z_+ \quad (4a)$$

$$y_k = C x_k + D u_k \quad (4b)$$

Theorem 1. (Kaczorek 2007a) The solution of equation (4a) with initial condition x_0 is given by

$$x_k = \Phi_k x_0 + \sum_{i=0}^{k-1} \Phi_{k-i-1} B u_i \quad (5)$$

where Φ_k is determined by the equation

$$\Phi_{k+1} = (A + I_n \alpha) \Phi_k + \sum_{i=2}^{k+1} (-1)^{i+1} \begin{pmatrix} \alpha \\ i \end{pmatrix} \Phi_{k-i+1} \quad (6)$$

with $\Phi_0 = I_n$.

Definition 1. The system (4) is called the (internally) positive fractional system if and only if $x_k \in \Re_+^N$ and $y_k \in \Re_+^p$, $k \in Z_+$ for any initial conditions $x_0 \in \Re_+^n$ and all input sequences $u_k \in \Re_+^m$, $k \in Z_+$.

It is easy to show (Kaczorek 2007a) that for $0 < \alpha < 1$

$$(-1)^{i+1} \begin{pmatrix} \alpha \\ i \end{pmatrix} \Phi_{k-i+1} > 0, \ i = 1,2,\ldots \quad (7)$$

Theorem 2. (Kaczorek 2007a) The fractional system (4) for $0 < \alpha < 1$ is positive if and only if

$$(A + I_n \alpha) \in \Re_+^{n \times n}, \quad B \in \Re_+^{n \times m}, \quad C \in \Re_+^{p \times n}, \quad D \in \Re_+^{p \times m}. \quad (8)$$

From (7), (8) and (5) we have $\Phi_k \in \Re_+^{n \times n}$ for $k = 1, 2, \ldots$ and the impulse response matrix

$$g_i = \begin{cases} D & \text{for } i = 0 \\ C \Phi_{i-1} B + D & \text{for } i = 1, 2 \end{cases} \quad (9)$$

of the positive fractional system (4) is nonnegative, $g_i \in \Re_+^{p \times m}$ for $i \in Z_+$.

REACHABILITY

Definition 2. A state $x_f \in \Re_+^n$ of the fractional system (4) is called reachable in q steps if there exists a input sequence $u_k \in \Re_+^m$, $k \in Z_+$ which steers the state from zero initial state $(x_0 = 0)$ to the final state x_f. If every state $x_f \in \Re_+^n$ is reachable in q steps then the system is called reachable in q steps. If for every state $x_f \in \Re_+^n$ there exists a natural number q, such that the state is reachable in k steps then the system is called reachable.

Let e_i, $i = 1, \ldots, n$ be the ith column of the identity matrix I_n. A column $a e_i$ for $a > 0$ is called the monomial column.

Theorem 3. The fractional system (4) is reachable in q steps if and only if the reachability matrix

$$R_q := [B, \Phi_1 B, \ldots, \Phi_{q-1} B] \quad (10)$$

contains n linearly independent monomial columns.

Proof. Using (5) for $k = q$ and $x_0 = 0$ we obtain

$$x_f = x_q = \sum_{i=0}^{q-1} \Phi_{q-i-1} B u_i = R_q \begin{bmatrix} u_{q-1} \\ u_{q-2} \\ \vdots \\ u_0 \end{bmatrix} \quad (11)$$

From Definition 2 and (11) it follows that for every $x_f \in \Re_+^n$ there exists a input sequence $u_i \in \Re_+^m$, $i = 0, 1, \ldots, q-1$ if and only if the matrix (21) contains n linearly independent monomial columns. ∎

Example 1. Consider the fractional system (4) for $0 < \alpha < 1$ with

$$A = \begin{bmatrix} 1 & 0 \\ 0 & -\alpha \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad (n = 2) \qquad (12)$$

The fractional system is positive since

$$A + I_n \alpha = \begin{bmatrix} 1+\alpha & 0 \\ 0 & 0 \end{bmatrix} \in \Re_+^{2\times 2}$$

Using (6) for $k = 0,1,...$ we obtain diagonal matrices of the forms

$$\Phi_1 = (A + I_n\alpha)\Phi_0 = \begin{bmatrix} 1+\alpha & 0 \\ 0 & 0 \end{bmatrix},$$

$$\Phi_2 = (A + I_n\alpha)\Phi_1 - \binom{\alpha}{2}\Phi_0 =$$

$$= \begin{bmatrix} \dfrac{\alpha^2 + 5\alpha + 2}{2} & 0 \\[2mm] 0 & \dfrac{(1-\alpha)\alpha}{2} \end{bmatrix},$$

$$\Phi_3 = (A + I_n\alpha)\Phi_2 - \binom{\alpha}{2}\Phi_1 + \binom{\alpha}{3}\Phi_0 =$$

$$= \begin{bmatrix} \dfrac{(\alpha^2 + 5\alpha + 2)(\alpha+1) - \alpha(\alpha-1)(\alpha+5)}{6} & 0 \\[2mm] 0 & \dfrac{\alpha(1-\alpha)(\alpha-2)}{2} \end{bmatrix},...$$

Note that the reachability matrices (10) for $q = 2,3,...$ have the forms

$$R_2 = [B, \Phi_1 B] = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix},$$

$$R_3 = [B, \Phi_1 B, \Phi_2 B] = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & \dfrac{(1-\alpha)\alpha}{2} \end{bmatrix},$$

$$R_4 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 2 & \dfrac{(1-\alpha)\alpha}{2} & \dfrac{\alpha(1-\alpha)(\alpha-2)}{6} \end{bmatrix},...$$

and they contain only one linearly independent monomial column. Therefore, by Theorem 3 the system with (12) is unreachable.

Example 2. Consider the fractional system (4) for $0 \le \alpha \le 1$ with

$$A = \begin{bmatrix} -\alpha & 0 \\ 1 & 2 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad (n = 2) \qquad (13)$$

The system is positive since

$$A + I_n \alpha = \begin{bmatrix} 0 & 0 \\ 1 & 2+\alpha \end{bmatrix} \in \Re_+^{2\times 2}$$

Using (6) for $k = 0$ we obtain

$$\Phi_1 = (A + I_n\alpha)\Phi_0 = \begin{bmatrix} 0 & 0 \\ 1 & 2+\alpha \end{bmatrix}$$

The reachability matrix (10) for $q = 2$ has the form

$$R_q = [B, \Phi_1 B] = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

It contains two linearly independent monomial columns. Therefore, the fractional system with (13) is reachable in two steps.

Remark 1. From (6) and (10) it follows that the positive fractional system (4) is reachable only if the matrix

$$[B, A + I_n\alpha] \qquad (14)$$

contains n linearly independent columns.

MINIMUM ENERGY CONTROL

Problem formulation

Consider the positive fractional system (4). If the system is reachable in q steps then there exist many input sequences that steer the state of the system from $x_0 = 0$ to the final state $x_f \in \Re_+^n$. Among these input sequences we are looking for the sequence $u_i \in \Re_+^m$, $i = 0,1,...,q-1, \ i \in Z_+$ that minimizes the performance index

$$I(u) = \sum_{j=0}^{q-1} u_j^T Q u_j \qquad (15)$$

where $Q \in \Re^{m\times m}$ is a symmetric positive define matrix and q is a given number of steps in which the state of the system is transferred from $x_0 = 0$ to $x_f \in \Re_+^n$.

The minimum energy control problem for the positive fractional system (4) can be stated as follows. Given matrices A, B and the order α of the system (4), the number of steps q, $x_f \in \Re_+^n$ and the matrix Q of the performance index (15), find a sequence of inputs

$u_i \in \mathfrak{R}^m_+$, $i = 0,1,...,q-1$, that steers the state of the system from $x_0 = 0$ to $x_f \in \mathfrak{R}^n_+$ and minimizes the performance index (15).

Problem solution

To solve the problem we define the matrix

$$W = W(q,Q) = R_q \overline{Q} R_q^T \in \mathfrak{R}^{n \times n} \qquad (16)$$

where R_q is defined by (10) and

$$\overline{Q} = \text{block diag}\left[Q^{-1},...,Q^{-1}\right] \in \mathfrak{R}^{qm \times qm} \qquad (17)$$

From (16) it follows that the matrix W is nonsingular if and only if rank $R_q = n$. If the condition is met then the system is reachable in q steps (as standard but not as a positive systems). In this case we may define for a given $x_f \in \mathfrak{R}^n_+$ the following sequence of inputs

$$\hat{u}_{0q} = \begin{bmatrix} \hat{u}_{q-1} \\ \hat{u}_{q-2} \\ \vdots \\ \hat{u}_0 \end{bmatrix} := \overline{Q} R_q^T W^{-1} x_f \qquad (18)$$

From (18) it follows that $\hat{u}_i \in \mathfrak{R}^m_+$, $i = 0,1,...,q-1$ if

$$\overline{Q} R_q^T W^{-1} \in \mathfrak{R}^{m \times n}_+ \qquad (19)$$

and this holds if

$$Q^{-1} \in \mathfrak{R}^{m \times m}_+ \qquad (20a)$$

and

$$W^{-1} \in \mathfrak{R}^{n \times n}_+ \qquad (20b)$$

Theorem 4. Let the positive fractional system (4) be reachable in q steps and the conditions (20) are satisfied. Moreover let $\overline{u}_i \in \mathfrak{R}^m_+$, $i = 0,1,...,q-1$ be a sequence of inputs that steers the state of the system from $x_0 = 0$ to $x_f \in \mathfrak{R}^n_+$. Then the sequence of inputs $\hat{u}_i \in \mathfrak{R}^m_+$, $i = 0,1,...,q-1$ defined by (18) also steers the state of the system from $x_0 = 0$ to $x_f \in \mathfrak{R}^n_+$ and minimizes the performance index (15), i.e.

$$I(\hat{u}) \leq I(\overline{u}) \qquad (21)$$

The minimal value of (15) for (18) is given by

$$I(\hat{u}) = x_f^T W^{-1} x_f \qquad (22)$$

Proof. If the fractional system (4) is positive and reachable in q steps and the assumptions (20) are satisfied then for $x_f \in \mathfrak{R}^n_+$ we have $\hat{u}_i \in \mathfrak{R}^m_+$ for $i = 0,1,...,q-1$. We shall show that the sequence (18) steers the state of the system from $x_0 = 0$ to $x_f \in \mathfrak{R}^n_+$. Using (5) for $k = q$, $x_0 = 0$ and (18), (16) we obtain

$$x_q = R_q \hat{u}_{0q} = R_q \overline{Q} R^T W^{-1} x_f = x_f \qquad (23)$$

since $R_q \overline{Q} R^T W^{-1} = I_n$.

The both sequences of inputs $\overline{u}_{0q}$ and $\hat{u}_{0q}$ steer the state of the system from $x_0 = 0$ to x_f. Hence $x_f = R_q \hat{u}_{0q} = R_q \overline{u}_{0q}$ and

$$R_q [\hat{u}_{0q} - \overline{u}_{0q}] = 0 \qquad (24)$$

Using (24) and (18) we shall show that

$$[\hat{u}_{0q} - \overline{u}_{0q}]^T \hat{Q} \hat{u}_{0q} = 0 \qquad (25)$$

where $\hat{Q} = \text{block diag}[Q,...,Q]$.

Transposition of (24) yields $[\hat{u}_{0q} - \overline{u}_{0q}]^T R_q^T = 0$.

Postmultiplying the equality by $W^{-1} x_f$ we obtain

$$[\hat{u}_{0q} - \overline{u}_{0q}]^T R_q^T W^{-1} x_f = 0. \qquad (26)$$

Using (18) and (26) we obtain (25) since

$$[\hat{u}_{0q} - \overline{u}_{0q}]^T \hat{Q} \hat{u}_{0q} = [\hat{u}_{0q} - \overline{u}_{0q}]^T \hat{Q} \overline{Q} R^T W^{-1} x_f =$$
$$= [\hat{u}_{0q} - \overline{u}_{0q}]^T R^T W^{-1} x_f = 0$$

and $\hat{Q}\overline{Q} = I_{qm}$.

Using (25) it is easy to verify that

$$\overline{u}_{0q}^T \overline{Q} \hat{u}_{0q} = \hat{u}_{0q}^T \hat{Q} \hat{u}_{0q} + [\overline{u}_{0q} - \hat{u}_{0q}]^T \hat{Q}[\overline{u}_{0q} - \hat{u}_{0q}] \qquad (27)$$

From (27) it follows that the inequality (21) holds, since

$$[\overline{u}_{0q} - \hat{u}_{0q}]^T \hat{Q}[\overline{u}_{0q} - \hat{u}_{0q}] \geq 0$$

To find the minimal value of the performance index we substitute (18) into (15) and we use (16). Then we obtain

$$I(\hat{u}) = \hat{u}_{0q}^T \hat{Q} \hat{u}_{0q} = (\bar{Q} R_q^T W^{-1} x_f)^T \hat{Q} (\bar{Q} R_q^T W^{-1} x_f) =$$
$$= x_f^T W^{-1} R_q \bar{Q} R_q^T W^{-1} x_f = x_f^T W^{-1} x_f$$

since $\hat{Q}\bar{Q} = I_{qm}$ and $W^{-1} R_q \bar{Q} R_q^T = I_n$. ∎

If the conditions of Theorem 4 are met then the minimal energy control problem can be solved by the use of the following procedure.

Procedure

Step 1. Knowing A, B, α, q, Q and using (10) and (17)
find R_q and $\bar{Q}$

Step 2. Knowing R_q, $\bar{Q}$ and using (16) find the
matrix W

Step 3. Using (18) compute the sequence of inputs
$\hat{u}_0, \hat{u}_1, ..., \hat{u}_{q-1}$

Step 4. Using (22) compute $I(\hat{u})$

Example 3. Given the positive fractional system (4) for $0 < \alpha < 1$ with (13). Find an optimal sequence of inputs that steers the state of the system from $x_0 = 0$ to

$$x_f = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$ in two steps $(q = 2)$ and minimizes the

performance index (15) for $Q = [2]$.

In Example 2 it was shown that the system is reachable in two steps. It is easy to see that the conditions of Theorem 4 are met. Using Procedure we obtain the following.

Step 1. In this case

$$R_2 = [B, \Phi_1 B] = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

and

$$Q = \mathrm{diag}[Q^{-1} \quad Q^{-1}] = \frac{1}{2} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Step 2. Using (16) we obtain

$$W = R_q \bar{Q} R_q^T = \bar{Q} = \frac{1}{2} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Step 3. Using (18) we obtain

$$\hat{u}_{02} = \begin{bmatrix} \hat{u}_1 \\ 0 \end{bmatrix} = \bar{Q} R_2^T W^{-1} x_f = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \qquad (28)$$

It is easy to verify that the sequence (28) steers the state of the system in two steps from $x_0 = 0$ to $x_f = [1 \quad 1]^T$.

Step 4. The minimal value of the performance index in this case is equal to

$$I(\hat{u}) = x_f^T W^{-1} x_f = \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = 4$$

CONCLUDING REMARKS

The minimum energy control problem of positive fractional discrete linear systems has been addressed. Necessary and sufficient conditions for the reachability of the systems have been established. Sufficient conditions for the solvability of the minimum energy control of the positive fractional discrete-time systems have been given and a procedure for computation of the optimal sequence of inputs minimizing the performance index (15) has been proposed. The considerations can be extended for positive fractional discrete-time systems with delays and continuous-time linear systems.
An extension of these considerations for positive 2D linear systems is an open problem.

REFERENCES

Busłowicz M., Kaczorek T. 2004. "Reachability and minimum energy control of positive linear discrete-time systems with one delay", In 12th *Mediterranean Conference on Control and Automation*, Kusadasi, Izmir, Turkey, CD-ROM.

Farina L., Rinaldi S. 2000. *Positive Linear Systems; Theory and Applications*, J. Wiley, New York.

Ferreira N.M.F., Machado J.A.T. 2003 "Fractional-order hybrid control of robotic manipulators", In *Proc. 11th Int. Conf. Advanced Robotics, ICAR'2003*, Coimbra, Portugal, pp. 393-398.

Gałkowski K. Kummert A. 2005. "Fractional polynomials and nD systems". In *Proc IEEE Int. Symp. Circuits and Systems*, ISCAS'2005, Kobe, Japan, CD-ROM.

Kaczorek T. 2002. *Positive 1D and 2D Systems*, Springer-Verlag, London.

Kaczorek T., 2006. "Computation of realizations of discrete-time cone systems". *Bull. Pol. Acad. Sci. Techn*. Vol. 54, No. 3, pp.347-350.

Kaczorek T. 2007. "Reachability and controllability to zero of positive fractional discrete-time systems", *Machine Intelligence and Robotic Control*, vol. 6, No. 4.

Kaczorek T. 2007. "Rechability and controllability to zero of cone fractional linear systems". *Archives of Control Sciences*. Vol. 17, No. 3, pp. 357-367.

Kaczorek T. 2007. "Realization problem for positive continuous-time systems with delays". *Intern. J. Comput. Intellig. and Appl.*, vol. 6, No. 2, pp. 289-298.

Kaczorek T. 2007. "Realization problem for singular positive continuous-time systems with delays". *Control and Cybernetics*. Vol. 36, No. 1, pp. 2-11.

Kaczorek T., 2008. "Reachability and controllability to zero tests for standard and positive fractional discrete-time systems". *Journal of Automation and Systems Engineering* (in Press).

Kaczorek T. 2008. "Fractional positive continuous-time linear systems and their reachability"*, Int. J. Appl. Math. Comp. Sci.*, vol. 18, No. 2. (in Press).

Kaczorek T. and Klamka J. 1986. "Minimum energy control of 2D linear systems with variable coefficients". Intern. Journal of Control, vol. 44, No. 3, pp 654-650.

Klamka J. 1991. Controllability of Dynamical Systems, Kluwer Academic Publ., Dordrecht.

Klamka J. 1976. "Relative controllability and minimum energy control of linear systems with distributed delays in control" *IEEE Trans. Autom. Contr.*, vol. AC-21, No. 4, pp.594-595.

Klamka J. 1983. "Minimum energy control of 2D systems in Hilbert spaces" *Systems Science*, vol. 9, No. 1-2, pp. 33-42.

Klamka J. 2002. "Positive controllability of positive systems" In *Proc. of American Control Conference*, ACC-2002, Anchorage, CD-ROM.

Klamka J. 2005. "Approximate constrained controllability of mechanical systems" *Journal of Theoretical and Applied Mechanics*, vol. 43, No. 3, pp. 539-554.

Miller K.S. and Ross B. 1983. *An Introduction to the Fractional Calculus and Fractional Differential Equations*. Willey, New York.

Moshrefi-Torbati M. and Hammond K. 1998. "Physical and geometrical interpretation of fractional operators". *J. Franklin Inst.,* Vol. 335B, no. 6, pp. 1077-1086.

Nishimoto K. 1984. *Fractional Calculus*. Koriyama: Decartes Press.

Oldham K. B., Spanier J. 1974. *The Fractional Calculus*. New York: Academmic Press.

Ortigueira M.D. 1997. "Fractional discrete-time linear systems", In *Proc. of the IEE-ICASSP 97*, Munich, Germany, IEEE, New York, vol. 3, 2241-2244

Ostalczyk P. 2000. "The non-integer difference of the discrete-time function and its application to the control system synthesis*". Int. J. Syst, Sci.* vol. 31, no. 12, pp. 1551-1561.

Ostalczyk P. 2004. "Fractional-Order Backward Difference Equivalent Forms Part I – Horner's Form". In *Proc. 1-st IFAC Workshop Fractional Differentation and its Applications, FDA'04*, Enseirb, Bordeaux, France, pp. 342-347.

Ostalczyk P. 2004. "Fractional-Order Backward Difference Equivalent Forms Part II – Polynomial Form". In *Proc. 1st IFAC Workshop Fractional Differentation and its Applications, FDA'04,* Enseirb, Bordeaux, France, pp. 348-353.

Oustaloup A. 1993. *Commande CRONE*. Paris, Hermés.

Oustaloup A. 1995. *La dérivation non entiére*. Paris: Hermés.

Podlubny I. 1999. *Fractional Differential Equations*. San Diego: Academic Press.

Podlubny I. 2002. "Geometric and physical interpretation of fractional integration and fractional differentation". *Fract. Calc. Appl. Anal*. Vol. 5, No. 4, pp. 367-386.

Samko S. G., Kilbas A.A., Maritchev O.I. 1993. *Fractional Integrals and Derivative. Theory and Applications*. London: Gordon&Breach.

Sierociuk D. and Dzieliński D. 2006. "Fractional Kalman filter algorithm for the states, parameters and order of fractional system estimation", *Int. J. Appl. Math. Comp. Sci.*, Vol. 16, No. 1, pp. 129-140.

Vinagre B. M., Monje C. A., Calderon A.J. 2002. "Fractional order systems and fractional order control actions". *Lecture 3 IEEE CDC'02 TW#2: Fractiional calculus Applications in Autiomatic Control and Robotics*.

Vinagre B. M., Feliu V. 2002. "Modeling and control of dynamic system using fractional calculus: Application to electrochemical processes and flexible structures". In *Proc. 41st IEEE Conf. Decision and Control*, Las Vegas, NV, pp. 214-239.

Xie G., Wang L. 2003. "Reachability and controllability of positive linear discrete-time systems with time-delays", In *L. Benvenuti, A. De Santis and L. Farina (eds): Positive Systems, LNCIS 294*, Springer-Verlag, Berlin, pp. 377-384.

AUTHOR BIOGRAPHY

TADEUSZ KACZOREK, born on 27 April 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 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 20 books (six in English) and almost 700 scientific papers. He supervised 63 PhD theses. He is Editor-in-Chief of Bulletin of Polish Academy of Scences, Technology Sciences and Editorial Member of about ten international journals. Tadeusz Kaczorek can be contacted at: `kaczorek@isep.pw.edu.pl`

SIMULATION ANALYSES OF CONTINUOUS STIRRED TANK REACTOR

Jiri Vojtesek and Petr Dostal
Faculty of Applied Informatics
Tomas Bata University in Zlin
Nad Stranemi 4511, 760 05 Zlin, Czech Republic
E-mail: {vojtesek,dostalp}@fai.utb.cz

KEYWORDS

CSTR, Adaptive control, Polynomial approach, LQ, Recursive identification

ABSTRACT

This paper presents simulation experiments on the continuous stirred tank reactor (CSTR) which is widely used equipment mainly in the chemical industry. The behaviour of these types of systems is usually nonlinear with other negative properties such as a time-delay or a non-minimum phase behaviour and simulation could help us with the understanding of them without making real experiments which could be dangerous, time or cost demanding. The simple iteration method and the Runge-Kutta's method were used for solving of a steady-state and dynamics of the system. Used adaptive control is based on the recursive identification of an External Linear Model (ELM) as a representation of the originally nonlinear system. The polynomial approach together with the LQ approach gives sufficient control results although the system has negative control properties.

INTRODUCTION

The simulation of the system on a computer enroll big boom nowadays when speed and availability of the computer technologies grows rapidly. On the contrary, the purchasing prise and running costs are relatively low.

The computer simulation is usually connected with the mathematical model as a result of modeling procedure (Ingham et al. 2000). Material and heat balances are one way how to describe the system and relations between unknown quantities in the mathematical way. These balances are then represented by ordinary or partial differential equations depending on the type of systems.

The Continuous Stirred Tank Reactor (CSTR) is typical equipment used in the industry for its good properties from control point of view. The CSTR belongs to the class of lumped parameters systems, a mathematical model of which is described by the set of ordinary differential equations (ODE).

The simulation analysis of the system usually consists of steady-state and dynamic analyses (Ingham et al. 2000, Luyben 1989). The simple iteration method and Runge-Kutta's method (Lyuben 1989) were used in the work for numerical solving of the steady-state and

dynamic analyses. These methods are well known, simple and Runge-Kutta's method is fully implemented in the used mathematical software Matlab.

Results from simulation experiments are then used for choosing of the control strategy and designing of the controller. The nonlinearity and negative control properties of the system should be overcome with the use of Adaptive control (Åström 1989).

Adaptive approach used in this work is based the choice of an External Linear Model (ELM) parameters of which are recomputed recursively during the control (Bobal et al. 2005). The external delta models (Middleton and Goodwin 2004) were used for parameter estimation. Although delta models belong to the range of discrete models, parameters of these models approaches to their continuous-time counterparts up to some assumptions (Stericker and Sinha 1993). Ordinary recursive least squares method (Fikar and Mikles 1999) was used for parameter estimation during the control. A polynomial approach with one degree-of-freedom (1DOF) configuration used for the controller synthesis has satisfied basic control requirements and connected with the LQ control technique, it fulfills the requirements of stability, asymptotic tracking of the reference signal and compensation of disturbances (Kucera 1993).

MODEL OF THE PLANT

As it is written above, the chemical process under consideration is the Continuous Stirred Tank Reactor (CSTR). The schematical representation of the CSTR is in Figure 1.

We supposed that reactant is perfectly mixed and react to the final product with the concentration $c_A(t)$. The heat produced by the reaction is represented by the temperature of the reactant $T(t)$. Furthermore we expect that volume, heat capacities and densities are constant during the control due to simplification.

A mathematical model of this system is derived from the material and heat balances of the reactant and cooling. The resulted model is then set of two Ordinary Differential Equations (ODEs) (Gao et al. 2002):

$$\frac{dT}{dt} = a_1 \cdot (T_0 - T) + a_2 \cdot k_1 \cdot c_A + a_3 \cdot q_c \cdot \left(1 - e^{\frac{a_4}{q_c}}\right) \cdot (T_0 - T) \quad (1)$$

$$\frac{dc_A}{dt} = a_1 \cdot (c_{A0} - c_A) - k_1 \cdot c_A$$

Where a_{1-4} are constants computed as

$$a_1 = \frac{q}{V}; \; a_2 = \frac{-\Delta H}{\rho \cdot c_p}; \; a_3 = \frac{\rho_c \cdot c_{pc}}{\rho \cdot c_p \cdot V}; \; a_4 = \frac{-h_a}{\rho_c \cdot c_{pc}} \quad (2)$$

variable t in previous equations denotes time, T is used for temperature of the reactant, V is volume of the reactor, c_A represents concentration of the product, q and q_c are volumetric flow rates of the reactant and cooling respectively. Indexes $(\cdot)_0$ denotes inlet values of the variables and $(\cdot)_c$ is used for variables related to the cooling. The fixed values of the system are shown in Table 1 (Gao et al. 2002).

The reaction rate, k_1, is computed from Arrhenius law:

$$k_1 = k_0 \cdot e^{\frac{-E}{R \cdot T}} \quad (3)$$

where k_0 is reaction rate constant, E denotes an activation energy and R is a gas constant. As you can see, this reaction rate is nonlinear function of the temperature T and we can say, that this system is **a nonlinear system with lumped parameters**.
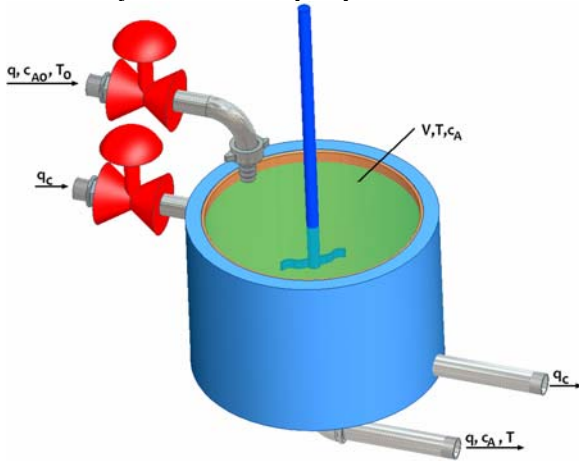
Figure 1: Continuous Stirred Tank Reactor

Table 1: Fixed parameters of the reactor

Reactant's flow rate	$q = 100 \; l.min^{-1}$
Reactor's volume	$V = 100 \; l$
Reaction rate constant	$k_0 = 7.2 \cdot 10^{10} \; min^{-1}$
Activation energy to R	$E/R = 1 \cdot 10^4 \; K$
Reactant's feed temperature	$T_0 = 350 \; K$
Reaction heat	$\Delta H = -2 \cdot 10^5 \; cal.mol^{-1}$
Specific heat of the reactant	$c_p = 1 \; cal.g^{-1}.K^{-1}$
Specific heat of the cooling	$c_{pc} = 1 \; cal.g^{-1}.K^{-1}$
Density of the reactant	$\rho = 1 \cdot 10^3 \; g.l^{-1}$
Density of the cooling	$\rho_c = 1 \cdot 10^3 \; g.l^{-1}$
Feed concentration	$c_{A0} = 1 \; mol.l^{-1}$
Heat transfer coefficient	$h_a = 7 \cdot 10^5 \; cal.min^{-1}.K^{-1}$

STEADY-STATE AND DYNAMIC ANALYSES

The system is submitted to the steady-state and dynamic analyses to obtain information about the behaviour of the system.

Steady-state Analysis

The steady-state analysis shows behaviour of the system in the steady-state, i.e. in $t \rightarrow \infty$ and results in optimal working point in the sense of maximal effectiveness and concentration yield. Mathematical meaning of the steady-state is that derivatives with respect to time variable are equal to zero, $d(\cdot)/dt = 0$. The mathematical model (1) is then transferred to the set of two nonlinear equations:

$$T^s = \frac{a_1 \cdot T_0 + a_2 \cdot k_1 \cdot c_A^s + a_3 \cdot q_c \cdot T_0 \cdot \left(1 - e^{\frac{a_4}{q_c}}\right)}{a_1 + a_3 \cdot q_c \cdot \left(1 - e^{\frac{a_4}{q_c}}\right)} \quad (4)$$

$$c_A^s = \frac{a_1 \cdot c_{A0}}{a_1 + k_1}$$

The simple iteration method was used for solving of this set of equation and the results are shown in figures.

The steady-state analysis was done for different volumetric flow rate of the reactant $q = <100; 200>$ in $l.min^{-1}$ and different volumetric flow rate of the cooling $q_c = <20; 100> \; l.min^{-1}$.
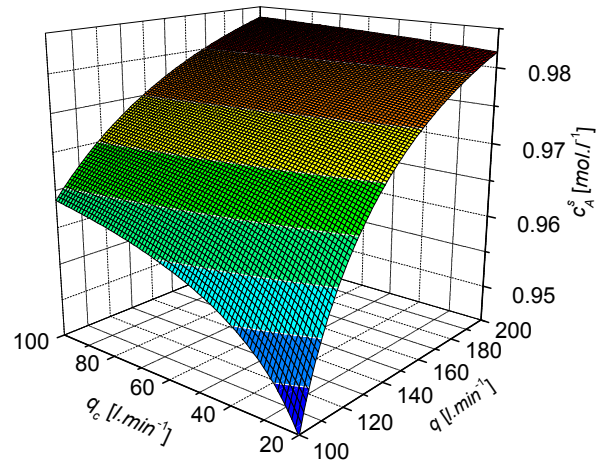
Figure 2: Steady-state values of concentration c_A for different volumetric flow rates q and q_c
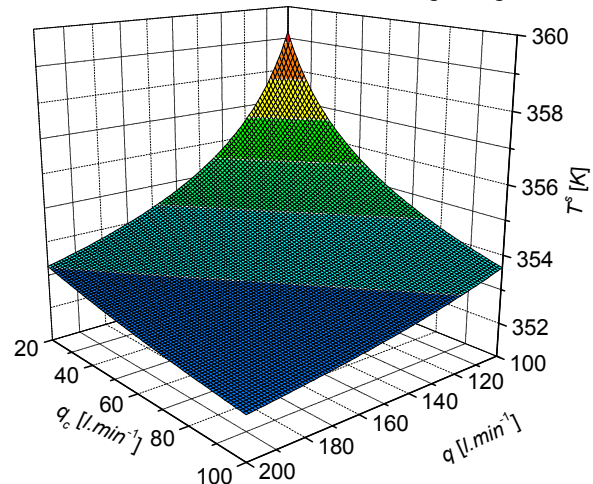

Figure 3: Steady-state values of temperature T for different volumetric flow rates q and q_c

As it could be seen in previous figures, system has nonlinear behaviour as we expected from the mathematical model. We cannot choose the exact optimal working point from figures but from the practical and mainly cost point of view is eligible to choose volumetric flow rates as low as possible. The working point is then characterized by the pair of volumetric flow rates:

$$q_c = 80 \ l.min^{-1} \quad q = 100 \ l.min^{-1} \quad (5)$$

The steady-state values of state variables T and are c_A for this working point

$$T^s = 354.26 \ K \quad c_A^s = 0.9620 \ mol.l^{-1} \quad (6)$$

Dynamic Analysis
This analysis means that we observe course of the state variables in time after the step change of some input variable. The step changes of volumetric flow rates q and q_c are input variables in our case and the steady-statel values in Equation (6) are initial conditions for the set of ODE (1). The Runge-Kutta's fourth order method was used for numerical solving of the set of ODE.

Six step changes of each input variable ($\pm80\%$, $\pm40\%$, $\pm20\%$ of its value in working point (5)) were done and the results are shown in Figure 4 – Figure 7 . The output variables y_1 and y_2 represents difference between the actual value and the steady-state value of the variable:

$$y_1(t) = T(t) - T^s; \quad y_2(t) = c_A(t) - c_A^s \quad (7)$$
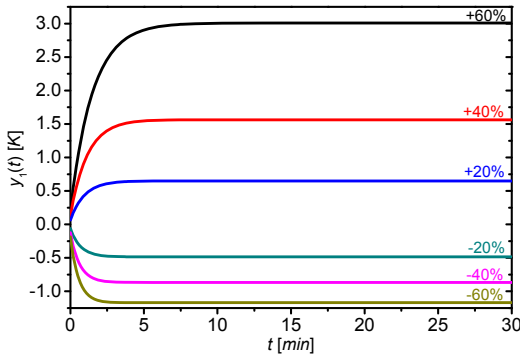
Figure 4: Time response of the output y_1 for various step changes of the input volumetric flow rate of cooling Δq_c
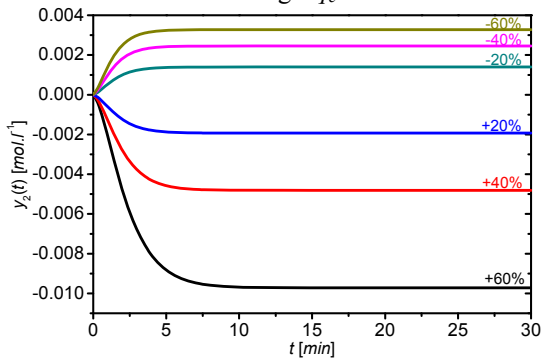
Figure 5: Time response of the output y_2 for various step changes of the input volumetric flow rate of cooling Δq_c

Step responses in Figure 4 and Figure 5 show that the output temperature, y_1, could be described by the first or the second order transfer function and the second order transfer function could be used as a description of the output – concentration c_A represented by y_2.
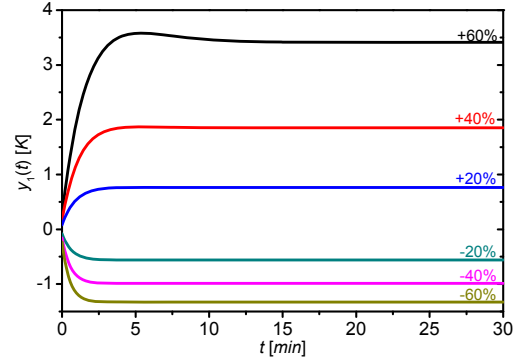
Figure 6: Time response of the output y_1 for various step changes of the input volumetric flow rate of reactant Δq
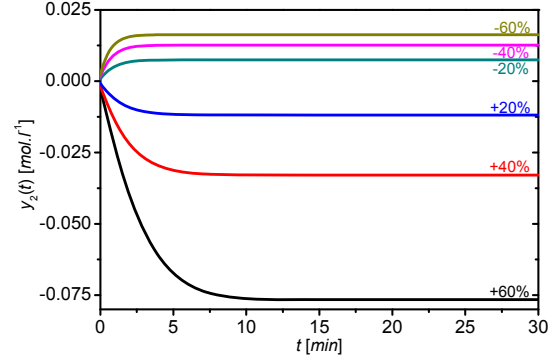
Figure 7: Time response of the output y_2 for various step changes of the input volumetric flow rate of reactant Δq

The second dynamic analysis for different step changes of the reactant's flow rate results in similar responses as in the previous case – see Figure 6 and Figure 7.

ADAPTIVE CONTROL

The input (control) variable is the change of the volumetric flow rate of the coolant and the output (controlled) variable is temperature of the reactant, i.e.

$$u(t) = \frac{q_c(t) - q_c^s}{q_c^s} \cdot 100 \ [\%] \quad y(t) = T(t) - T^s \ [K] \quad (8)$$

External Linear Model(ELM)
Although the original system has nonlinear behaviour, the External Linear Model (ELM) is used as a reprentation of the controlled system.
The controlled output is shown in Figure 4 which means that the transfer function could be the second order transfer function with relative order one:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{b(s)}{a(s)} = \frac{b_1 s + b_0}{s^2 + a_1 s + a_0} \quad (9)$$

This transfer function fulfils the condition of properness $\deg b \le \deg a$.

The ELM can be in continuous time or discrete time form. In this work, δ–model was used as an ELM. This model belongs to the class of discrete models but its properties are different from the classical discrete model in the Z-plain. If we want to convert Z-model to δ–model, we must introduce a new complex variable γ computed as (Mukhopadhyay et al. 1992)

$$\gamma = \frac{z-1}{\alpha \cdot T_v \cdot z + (1-\alpha) \cdot T_v} \qquad (10)$$

We can obtain infinitely many models for optional parameter α from the interval $0 \le \alpha \le 1$ and a sampling period T_v, however a *forward δ-model* was used in this work which has γ operator computed via

$$\alpha = 0 \Rightarrow \gamma = \frac{z-1}{T_v} \qquad (11)$$

The differential equation for ELM in the form of (9) is

$$\begin{aligned} y_\delta(k) = &-a_1 y_\delta(k-1) - a_0 y_\delta(k-2) + \\ &+ b_1 u_\delta(k-1) + b_0 u_\delta(k-2) \end{aligned} \qquad (12)$$

where y_δ is the recomputed output to the δ-model:

$$y_\delta(k) = \frac{y(k) - 2y(k-1) + y(k-2)}{T_v^2}$$

$$y_\delta(k-1) = \frac{y(k-1) - y(k-2)}{T_v}$$

$$y_\delta(k-2) = y(k-2) \qquad (13)$$

$$u_\delta(k-1) = \frac{u(k-1) - u(k-2)}{T_v}$$

$$u_\delta(k-2) = u(k-2)$$

and T_v is a sampling period, the data vector is then

$$\begin{aligned} \boldsymbol{\phi}_\delta^T(k-1) = &\left[-y_\delta(k-1), -y_\delta(k-2), \ldots \right. \\ &\left. \ldots, u_\delta(k-1), u_\delta(k-2) \right] \end{aligned} \qquad (14)$$

The vector of estimated parameters

$$\hat{\boldsymbol{\theta}}_\delta^T(k) = \left[a'_1, a'_0, b'_1, b'_0 \right] \qquad (15)$$

can be computed from the ARX (Auto-Regressive eXtrogenous) model

$$y_\delta(k) = \boldsymbol{\theta}_\delta^T(k) \cdot \boldsymbol{\varphi}_\delta(k-1) \qquad (16)$$

by some of the recursive least squares methods.

The parameters a'_1, a'_0, b'_1 and b'_0 are parameters of the delta model which are not identical to the parameters of the continuous-time model in Equation (9) but it was proofed for example in (Stericker and Sinha 1993) that parameters of polynomials $a'(\delta)$ and $b'(\delta)$ approach the parameters of the continuous-time model with decreasing value of the sampling period T_v.

The Recursive Least-Squares (RLS) method is well-known and widely used for the parameter estimation (Fikar and Mikles, 1999). It could be modified with some kind of forgetting, exponential or directional (Kulhavy and Karny, 1984), because parameters of the identified system can vary during the control which is typical for nonlinear systems and the use of some forgetting factor could result in better output response.

The RLS method with exponential forgetting is described by the set of equations:

$$\begin{aligned} \varepsilon(k) &= y(k) - \boldsymbol{\varphi}^T(k) \cdot \hat{\boldsymbol{\theta}}(k-1) \\ \gamma(k) &= \left[1 + \boldsymbol{\varphi}^T(k) \cdot \mathbf{P}(k-1) \cdot \boldsymbol{\varphi}(k) \right]^{-1} \\ \mathbf{L}(k) &= \gamma(k) \cdot \mathbf{P}(k-1) \cdot \boldsymbol{\varphi}(k) \\ \mathbf{P}(k) &= \frac{1}{\lambda_1(k-1)} \left[\mathbf{P}(k-1) - \frac{\mathbf{P}(k-1) \cdot \boldsymbol{\varphi}(k) \cdot \boldsymbol{\varphi}^T(k) \cdot \mathbf{P}(k-1)}{\lambda_1(k-1) + \boldsymbol{\varphi}^T(k) \cdot \mathbf{P}(k-1) \cdot \boldsymbol{\varphi}(k)} \right] \\ \hat{\boldsymbol{\theta}}(k) &= \hat{\boldsymbol{\theta}}(k-1) + \mathbf{L}(k) \varepsilon(k) \end{aligned} \qquad (17)$$

Several types of exponential forgetting can be used, e.g. RLS with constant exponential forgetting, RLS with increasing exp. forgetting etc. RLS with the changing exp. forgetting is used for parameter estimation here, where the changing forgetting factor λ_1 is computed

$$\lambda_1(k) = 1 - K \cdot \gamma(k) \cdot \varepsilon^2(k) \qquad (18)$$

K is Equation (18) small number, in our case $K = 0.001$.

Configuration of the Controller

The configuration with one degree-of-freedom (1DOF) was used for the control system set-up. This form has a controller in the feedback part (see Figure 8).
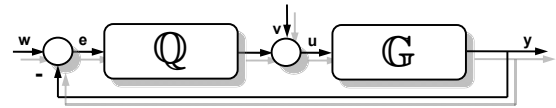

Figure 8: 1DOF control configuration

The block G in the Figure 8 represents the transfer function of the plant (9), w is the wanted value (reference signal), e stands for the control error ($e = w - y$), v is a disturbance, u is used for the control variable and y denotes the controlled output. Block Q is a transfer function of the controller which ensures stability, asymptotic tracking of the reference signal and load disturbance attenuation and it can be described by the polynomials in s-plain as

$$Q(s) = \frac{q(s)}{s \cdot \tilde{p}(s)} \qquad (19)$$

where degrees of the polynomials are computed from

$$\deg q(s) = \deg a(s), \ \deg \tilde{p}(s) \ge \deg a(s) - 1 \quad (20)$$

and parameters of the polynomials $\tilde{p}(s)$ and $q(s)$ are computed from a Diophantine equation (Kucera 1993):

$$a(s) \cdot s \cdot \tilde{p}(s) + b(s) \cdot q(s) = d(s) \qquad (21)$$

Polynomials $a(s)$ and $b(s)$ are known from the recursive identification and the polynomial $d(s)$ on the right side of (21) is an optional stable polynomial. Roots of this polynomial are called poles of the closed-loop and their position affects quality of the control.

This polynomial could be designed for example with the use of Pole-placement method (Vojtesek *et al.*, 2004).

The method presented here uses Linear Quadratic (LQ) approach which is based on the minimization of the cost function

$$J_{LQ} = \int_0^\infty \left\{ \mu_{LQ} \cdot e^2(t) + \varphi_{LQ} \cdot \dot{u}^2(t) \right\} dt \qquad (22)$$

where $\varphi_{LQ} > 0$ and $\mu_{LQ} \geq 0$ are weighting coefficients, $e(t)$ is control error and $\dot{u}(t)$ denotes difference of the input variable. Polynomial $d(s)$ in this case is

$$d(s) = g(s) \cdot n(s) \qquad (23)$$

and polynomials $n(s)$ and $g(s)$ are computed from the spectral factorization

$$(a \cdot f)^* \cdot \varphi_{LQ} \cdot a \cdot f + b^* \cdot \mu_{LQ} \cdot b = g^* \cdot g$$
$$n^* \cdot n = a^* \cdot a \qquad (24)$$

for control variable $u(t)$ and disturbance $v(t)$ from the ring of step functions $f(s) = s$. The resulted controller is strictly proper and the degree of $d(s)$ is computed via

$$\deg d = \deg(g \cdot n) = 2 \deg a + 1 \qquad (25)$$

The transfer function of the controller in (19) is for this case

$$\tilde{Q}(s) = \frac{q_2 s^2 + q_1 s + q_0}{s \cdot (s^2 + p_1 s + p_0)} \qquad (26)$$

and the polynomial $d(s)$ is from (25) of the fifth degree. The parameters of $n(s)$ and $g(s)$ are computed from Equation (24) as

$$g_0 = \sqrt{\mu_{LQ} b_0^2}, \; g_1 = \sqrt{2g_0 g_2 + \varphi_{LQ} a_0^2 + \mu b_1^2},$$
$$g_2 = \sqrt{2g_1 g_3 + \varphi_{LQ}(a_1^2 - 2a_0)}, g_3 = \sqrt{\varphi_{LQ}}, \qquad (27)$$
$$n_0 = \sqrt{a_0^2}, n_1 = \sqrt{2n_0 + a_1^2 - 2a_0}$$

The resultant controller works in continuous-time and in our case its structure corresponds to the structure of the real PID controller but its parameters vary according to the actual working point.

Control simulation results
Simulation experiments were done in the mathematical software Matlab, version 6.5.1. The sampling period was $T_v = 0.3$ *min*, the simulation time 1000 *min* and 5 different step changes were done during this time.
The input variable $u(t)$ was limited due to the physical realization to the bounds $u(t) = <-80\%; +80\%>$. The initial vector of parameters used for identification was $\hat{\theta}^T = [0.1, 0.1, 0.1, 0.1]$ and the initial covariance matrix was $P_{ii} = 1 \cdot 10^7$ for $i = 1,..,4$.
The tuning parameters of the controller are weighting factors φ_{LQ} and μ_{LQ}, where $\mu_{LQ} = 1$ and three simulation studies for different $\varphi_{LQ} = 0.25$, 1 and 2 were done. The results are shown in following figures.
The course of the output variable in Figure 9 shows that the proposed controller with 1DOF has problem only in

the very beginning of the control mainly because of recursive identification which has no adequate information about the system at the start.
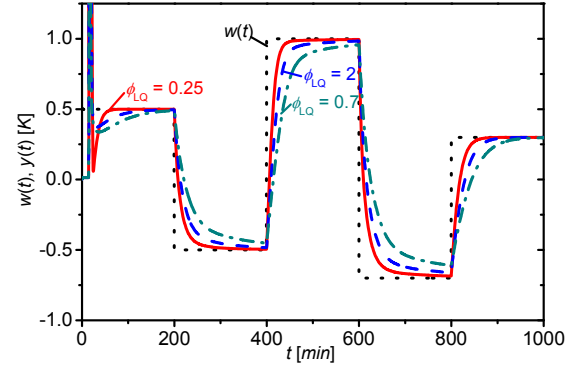
Figure 9: Output variable $y(t)$ for various values of weighting factor φ_{LQ}

However, the controller has a smooth course after next step changes. The increasing value of the weighting factor φ_{LQ} results in smoother but slower course of the output variable. As you can see, time 200 *min* which was used for each step is not enough for the controller with $\varphi_{LQ} = 2$ to reach the wanted value $w(t)$.
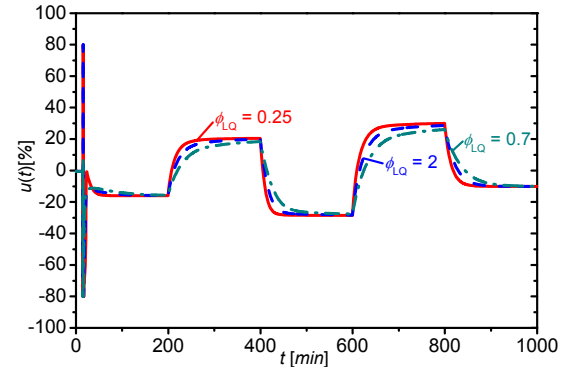

Figure 10: Input variable $u(t)$ for various values of weighting factor φ_{LQ}
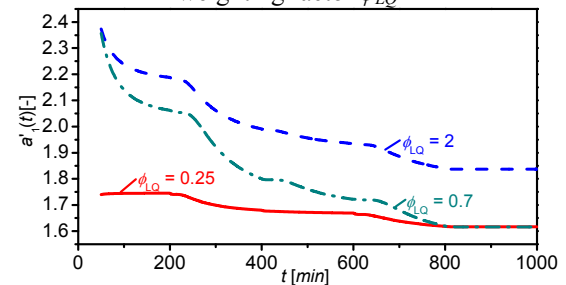

Figure 11: The course of the identified parameter a'_1 during the control

The effect of to the φ_{LQ} to the input variable is not very obvious, the courses in Figure 10 are nearly similar.
Used recursive identification with exponential forgetting has no problem with the recursive identification (see Figures 11 – 14) except the very beginning of the control as it was mentioned before.
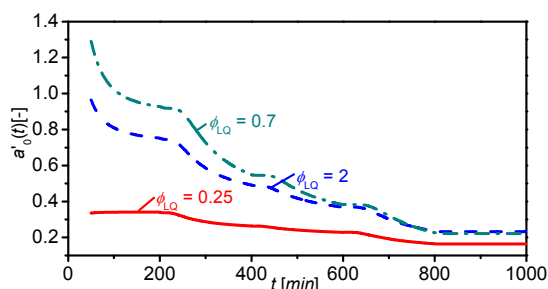
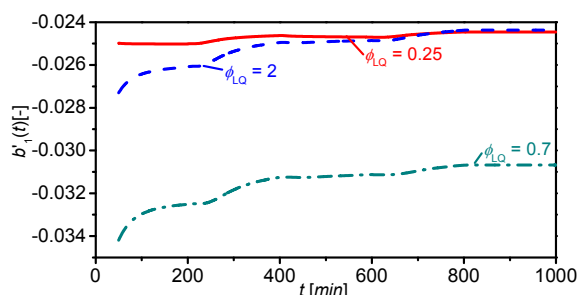Figure 12: The course of the identified parameter a'_0 during the control

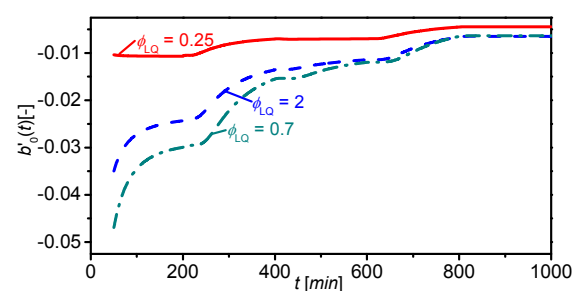Figure 13: The course of the identified parameter b'_1 during the control

Figure 14: The course of the identified parameter b'_0 during the control

CONCLUSION

Paper shows simulation analysis of a continuous stirred tank reactor with ordinary exothermic reaction inside and cooling in the jacket. The mathematical model is constructed by two ordinary differential equations. The simple iteration method and Runge-Kutta's method were used for numerical solving of steady-state and dynamic analyses. These analyses show mainly nonlinearity of the system and results in the choice of the second order transfer function with relative order one as ELM. Used adaptive control has good control results except for the very beginning of the control because of problems with recursive identification. The output responses after next step changes have smooth course without overshoots. The controller should be tuned via weighting factor φ_{LQ} where increasing value of this factor results in smoother but slower response. The used recursive least-squares method with exponential forgetting used for parameter estimation has no problem with the identification after initial "tuning" time. The next step which should follow after this simulation analyses is verification on the real system.

REFERENCES

Åström, K.J. a B. Wittenmark 1989. *Adaptive Control*. Addison Wesley, Reading, MA.

Bobal, V., Böhm, J., Fessl, J. Machacek, J (2005). *Digital Self-tuning Controllers: Algorithms, Implementation and Applications*. Advanced Textbooks in Control and Signal Processing. Springer-Verlag London Limited

Fikar, M., Mikles J. 1999. *System Identification*. STU Bratislava

Gao, R., O'dywer, A., Coyle, E. 2002. A Non-linear PID Controller for CSTR Using Local Model Networks. *Proc. of 4th World Congress on Intelligent Control and Automation*, Shanghai, P. R. China, 3278-3282

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 Verlagsgesellshaft, Weinheim.

Kucera, 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.

Middleton, R.H., Goodwin, G.C. 2004 *Digital Control and Estimation - A Unified Approach*. Prentice Hall, Englewood Cliffs.

Mukhopadhyay, S., Patra, A.G., Rao, G.P. (1992). New class of discrete-time models for continuos-time systeme. *International Journal of Control*, vol.**55**, 1161-1187

Stericker, D.L., Sinha, N.K. 1993. Identification of continuous-time systems from samples of input-output data using the δ-operator. *Control-Theory and Advanced Technology*, vol. 9, 113-125

Vojtěšek, J., Dostál, P., Haber, R. 2004. Simulation and Control of a Continuous Stirred Tank Reactor. In: *Proc. of Sixth Portuguese Conference on Automatic Control CONTROLO 2004*. Faro. Portugal, p. 315-320.

AUTHOR BIOGRAPHIES

JIRI VOJTESEK was born in Zlin, Czech Republic and studied at the Tomas Bata University in Zlin, where he got his master degree in chemical and process engineering in 2002. He has finished his Ph.D. focused on Modern control methods for chemical reactors in 2007. His contact is vojtesek@fai.utb.cz.

PETR DOSTAL studied at the Technical University of Pardubice. He obtained his PhD. degree in Technical Cybernetics in 1979 and he became professor in Process Control in 2000. 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@fai.utb.cz.

ANALYSIS OF A SEMI-BATCH REACTOR FOR CONTROL PURPOSES

František Gazdoš and Lubomír Macků
Faculty of Applied Informatics
Tomas Bata University in Zlín
Nad stráněmi 4511, 760 05, Zlín, Czech Republic
E-mail: gazdos@fai.utb.cz

KEYWORDS

Exothermic Semi-batch Reactor, Chromium Waste, Modelling, Analysis, Simulation.

ABSTRACT

The paper presents a control system design oriented analysis of a semi-batch reactor used for tanning waste recovery. The analysis is performed with the help of modelling and simulation means and provides useful information for optimal control design of the process.

INTRODUCTION

The tannery is an essential industry process today. Its product is a natural hide. Some of the leather properties (such as softness, plasticity, stability, absorption) cannot be replaced by any artificial material.

There are a lot of technological operations during the leather-to-hide conversion, starting with washing, continuing by tanning etc. These are usually connected with huge water, energy and chemicals consumption resulting in negative impact on the environment. Recently, one of the very fundamental operations is chromium salt tanning. It is possible to obtain only 200 [kg] of hide from 1000 [kg] of leather, but over 600 [kg] of solid waste originates from this process; the rest is drained away in the form of liquid waste during the chromium salt tanning process. The USA produces almost 60 000 tons of this solid waste and the worldwide production is approximately 10 times bigger (Cabeza et al. 1998). Currently, majority of this solid waste is land filled. It can result in leakage of Cr3+ into groundwater. After oxidation, for example in sewage treatment plants during purification from unhealthy bacteria, the Cr3+ to Cr6+ conversion can occur. The Cr6+ compounds belong to cancerous substances so they are dangerous to health. As a consequence, at present, alternative methods of dealing with the chromium waste are sought and investigated (Aloy and Vulliermet 1998; Tiravanti et al. 1996, 1997). The enzymatic hydrolysis is one of the considered alternatives (Kolomazník et al. 1996). This technique separates the chrome from protein in the form of the chromium filter cake. All products of this process are usable – it is a waste-free technology.

This paper deals with an analysis of a chemical reactor for chromium sludge (chromium filter cake) recovery. The reactor is used for the enzymatic hydrolysis and the analysis is performed from the control theory point of

view by simulation means mainly to obtain useful information for subsequent optimal control design.

The contribution is structured as follows: after the introduction, a detailed description of the reactor follows. In the next sections, a mathematical model of the system is derived and all variables are defined together with their physical values and limits. Further, steady-state and dynamical behaviour is studied by simulation means. A detailed analysis of the reactor from the systems theory point of view follows in next sections, giving useful information for optimal control design. Possible control strategies are discussed at the end of the contribution together with areas for possible future research.

A SEMI-BATCH REACTOR

The chromium sludge is processed in a chemical reactor sketched in Fig.1. by an exothermic chemical reaction with chrome sulphate acid. During this reaction a considerable quantity of heat is developing so that control of the reaction is necessary.
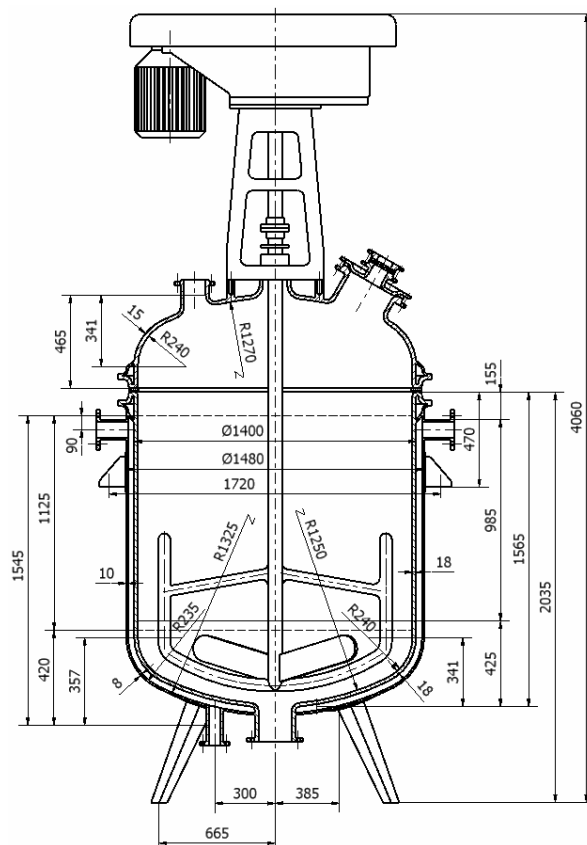

Figure 1: Chemical Reactor

System description

In order to investigate main properties of the real process, a mathematical model of the chemical reactor was derived based on Fig.2.
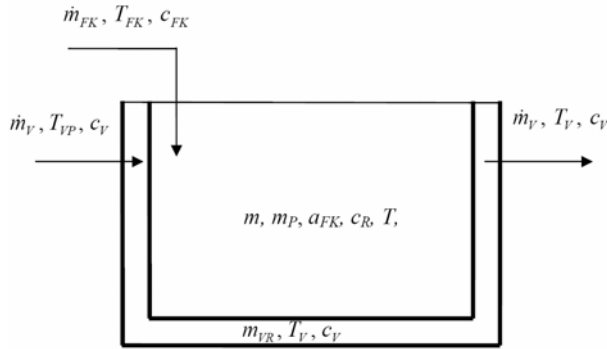

$\dot{m}_{FK}, T_{FK}, c_{FK}$

$\dot{m}_V, T_{VP}, c_V$ $\dot{m}_V, T_V, c_V$

$m, m_P, a_{FK}, c_R, T,$

m_{VR}, T_V, c_V

Figure 2: Chemical Reactor Scheme

The scheme above shows a chemical semi-batch reactor with initial filling m_P[kg] given by the solution of chemicals without the chromium sludge (filter cake). This is fed into the reactor by $\dot{m}_{FK}$ [kg/s] to control the developing heat since the temperature has to stay under a certain critical level ($T(t) \prec 100^o\text{C}$), otherwise the reactor could be destroyed. On the other hand it is desirable to utilise the maximum capacity of the reactor to process the maximum amount of waste in the shortest possible time (higher temperature is desirable). Therefore an optimal control strategy has to find a trade-off between these opposite requirements.

Mathematical model

Under usual simplifications, based on the mass and heat balance, the following 4 nonlinear ordinary differential equations can be derived (Macků 2005, 2004):

$$\dot{m}_{FK} = \frac{d}{dt} m(t) \qquad (1)$$

$$\dot{m}_{FK} = k\, m(t)\, a_{FK}(t) + \frac{d}{dt}\big[m(t)\, a_{FK}(t)\big]$$

$$\dot{m}_{FK} c_{FK} T_{FK} + \Delta H_r k\, m(t)\, a_{FK}(t) =$$

$$= K S\big[T(t) - T_v(t)\big] + \frac{d}{dt}\big[m(t)\, c_R\, T(t)\big]$$

$$\dot{m}_v c_v T_{vp} + K S\big[T(t) - T_v(t)\big] = \dot{m}_v c_v T_v(t) + m_{vR} c_v \frac{d}{dt} T_v(t)$$

The first equation expresses the total mass balance of the chemical solution in the reactor. The symbol $\dot{m}_{FK}$ [kg.s^{-1}] expresses the mass flow of the entering chromium sludge and $m'(t)$ [kg.s^{-1}] describes the accumulation of the in-reactor content.

The second equation represents the chromium sludge mass balance. The input is $\dot{m}_{FK}$ [kg.s^{-1}] again, the accumulation is given by the last term $\big[m(t)\, a_{FK}(t)\big]'$ [kg.s^{-1}], where $a_{FK}(t)[-]$ denotes the mass concentration of the chromium sludge in the reactor and $m(t)[kg]$ describes weight of the reaction components in the system. The expression $k\, m(t)\, a_{FK}(t)$ [kg.s^{-1}] defines the chromium sludge extinction by the chemical reaction. Here, k[s^{-1}] is the reaction rate constant expressed by the Arrhenius equation (2) where A [s^{-1}], E [J.mol^{-1}] and R [J.mol^{-1}.K^{-1}] are pre-exponential factor, activation energy and gas constant respectively.

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

The third equation describes the enthalpy balance. The input heat entering the reactor in the form of the chromium sludge is expressed by the term $\dot{m}_{FK} c_{FK} T_{FK}$, the heat arising from the chemical reaction is given by the expression $\Delta H_r k\, m(t)\, a_{FK}(t)$ and the heat transmission through the reactor wall is expressed by the formula $K S\big[T(t) - T_v(t)\big]$. The individual symbols used above mean: c_{FK} [J.kg^{-1}.K^{-1}] – chromium sludge specific heat capacity, c_R [J.kg^{-1}.K^{-1}] – specific heat capacity of the reactor content, T_{FK} [K] – chromium sludge temperature, ΔH_r [J.kg^{-1}] – reaction heat, K [J.m^{-2}. K^{-1}.s^{-1}] – conduction coefficient, S [m^2] – heat transfer surface, $T(t)$ [K] - temperature of reaction components in the reactor, $T_v(t)$ [K] – temperature of a coolant in the reactor double wall.

The last equation describes coolant heat balance. The input heat is given by $\dot{m}_v c_v T_{vp}$, the heat entering the coolant by the reactor wall is expressed by $K S\big[T(t) - T_v(t)\big]$, the heat going out with the coolant is described as $\dot{m}_v c_v T_v(t)$ and the heat accumulated in the double wall describes the last term $m_{vR} c_v T_V'(t)$. The symbols mean: $\dot{m}_v$ [kg.s^{-1}] – coolant mass flow, c_v [J.kg^{-1}.K^{-1}] – coolant specific heat capacity, T_{vp} [K] – input coolant temperature, m_{vR} [kg] – coolant mass weight in the reactor double wall.

Variables $\dot{m}_{FK}, \dot{m}_v, T_{FK}, T_{vp}$ are manipulated signals, however, from practical point of view, only $\dot{m}_{FK}$ and $\dot{m}_v$ are usable. The temperature change of T_{FK} or T_{vp} is inconvenient due to the economic reasons (great energy demands).

For simulation purposes, the following values of the constants used in the mathematical model (1) were adopted (obtained analytically, experimentally, estimated or taken from the literature, for details see Macků 2004): A = 219.6 s^{-1}, E = 29968 J.mol^{-1}, R = 8.314 J.mol^{-1}.K^{-1}, c_{FK} = 4400 J.kg^{-1}.K^{-1}, c_v = 4118

J.kg^{-1}.K^{-1}, c_R = 4500 J.kg^{-1}.K^{-1}, m_{vR} = 220 kg, ΔH_r = 1392350 J.kg^{-1}, K = 200 J.m^{-2}.K^{-1}.s^{-1}, S = 7.36 m^2, $\dot{m}_v$ = 1 kg.s^{-1}, T_{FK} = 293.15K = 20°C, T_{vp} = 288.15K = 15°C. The $\dot{m}_{FK}$ feeding varied from 0 to 3kg.s^{-1}. Initial conditions were defined as: $m(0) = m_p = 1810\,\text{kg}$ (initial reactor filling), $a_{FK}(0) = 0$ (initial mass concentration of the chromium sludge in the reactor), $T(0) = 323.15\,\text{K} =$ 50°C (initial temperature of the reactor filling) and initial coolant temperature $T_v(0) = 293.15\,\text{K} = 20°C$.

Technological limits and variables saturation

Maximum filling of the reactor is limited by its volume to the value of $m < 2450\,\text{kg}$ approximately. Then the process of feeding by the chromium sludge $\dot{m}_{FK}$ has to be stopped. The feeding can be practically realized in the range $\dot{m}_{FK} \in \langle 0; 3 \rangle$ kg.s^{-1}. As stated in the system description, the temperature cannot exceed the limit $T(t) \prec 100^o C$; this holds also for the coolant (water) but it is not so critical in this case as shown by the further experiments.

SIMULATION ANALYSIS

In order to prepare a convenient control strategy for the process described above, first a steady-state and dynamic analyses were performed with the help of simulation means.

Steady-state analysis

The reactor described above, i.e. a semi-batch reactor, has only one steady state given by:

$$\dot{m}_{FK} = 0; \quad a_{FK} = 0; \quad T = T_v = T_{vp} \qquad (3)$$

It means that the steady state can occur only at the beginning of the process (before feeding the reactor) or at the end of the reaction running inside.

Dynamics analysis

Dynamical behaviour analysis of a system is also an important part in the control design process. It gives insight into stability and other important properties of the system, such as time-constants, gains and non-minimum-phase behaviour. Using simulation means (MATLAB/Simulink), the following graphs were recorded as a step-responses of the input variable $\dot{m}_{FK}$ (other variables were kept at their operating values defined previously). The input variable was chosen in the range: $\dot{m}_{FK} = \begin{bmatrix} 0.05 & 0.1 & 0.5 & 1 & 3 \end{bmatrix}\ \begin{bmatrix} kg \cdot s^{-1} \end{bmatrix}$.
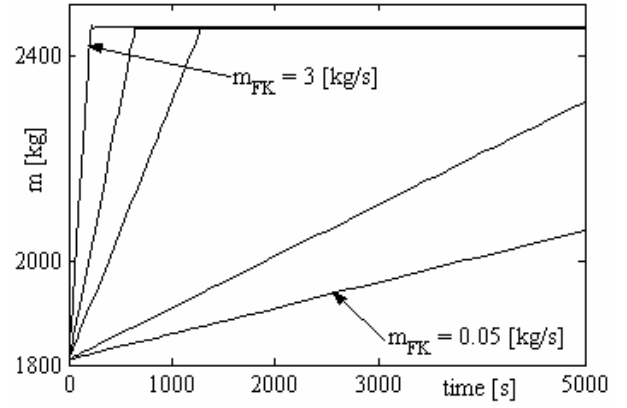

Figure 3: Total Mass Response

The figure above shows increase of the total mass in the reactor for various input flow rates of the chromium sludge. The simulation reveals integrating, astatic behaviour and it already respects the limitations introduced previously (max. mass in the reactor $m_{max} = 2450 [kg]$).
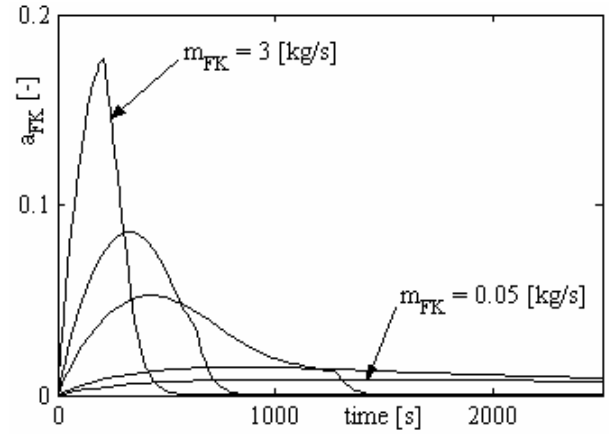

Figure 4: Chromium Sludge Mass Concentration Response

The response presented in Fig. 4 reveals derivative behaviour of the variable $a_{FK}(t)$ for various values of $\dot{m}_{FK}$.
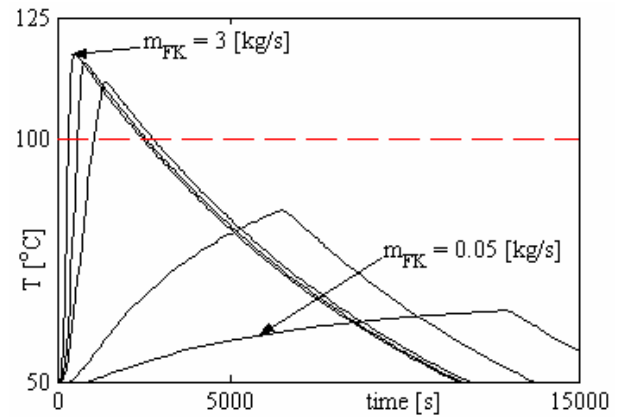

Figure 5: Temperature-in-the-Reactor Response

The response of Fig. 5 shows temperature increase inside the reactor – it can be seen how the temperature rises as a result of the chemical reaction. The faster input flow rate of the chromium sludge, the faster reaction and temperature increase. Then, the next increase is limited by the restriction on the maximum possible mass in the reactor followed by gradual temperature fall. From the graph it is also clear that for the simulated range of $\dot{m}_{FK}$ the temperature goes beyond the allowed limit $T(t) \prec 100$ [°C] for higher $\dot{m}_{FK}$-values, therefore the process needs to be controlled properly.
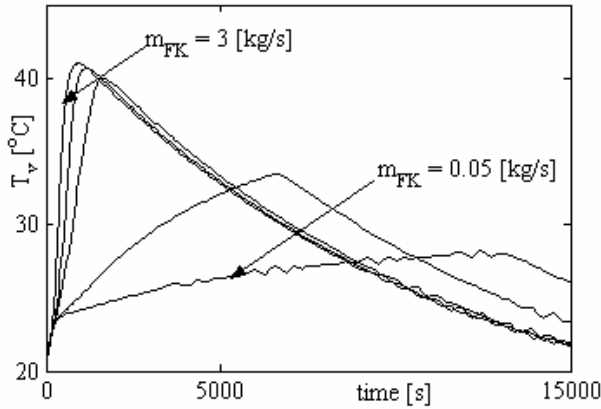
Figure 6: Coolant Temperature Response

A record of the coolant temperature is presented in Fig. 6. As can be seen from the graph, the temperature of the media for the whole range of $\dot{m}_{FK}$ is not critical since water is used for the cooling (provided the defined flow rate $\dot{m}_v = 1$ [kg/s] is ensured).

CONTROL THEORY POINT OF VIEW

From the systems theory point of view the reactor has four input signals $\dot{m}_{FK}(t)$, $\dot{m}_v(t)$, $T_{FK}(t)$ and $T_{vp}(t)$, four state variables $m(t)$, $a_{FK}(t)$, $T(t)$, $T_v(t)$ and one output signal to be controlled given by the temperature inside the reactor $T(t)$. Hence, it can be generally seen as a Multi Input – Multi Output (MIMO) system of 4th order. In addition it possesses strongly nonlinear behaviour. Practically, the only manipulated variables are input flow rates of the chromium sludge $\dot{m}_{FK}(t)$ and of the coolant $\dot{m}_v(t)$. Therefore, input temperatures of the filter cake $T_{FK}(t)$ and of the coolant $T_{vp}(t)$ can be alternatively seen as disturbances. For further analysis, the reactor model described by the system of differential equations (1) is transformed into a linear time-variant (LTV) system.

Linear model

Having generally a nonlinear model defined by a system of formulas

$$x'(t) = f\left[t, x(t), u(t)\right], \qquad (4)$$

where $x(t)$ defines a vector of state-variables $\left[x_1(t) \quad x_2(t) \quad \cdots \quad x_n(t)\right]$, $u(t)$ vector of input variables $\left[u_1(t) \quad u_2(t) \quad \cdots \quad u_m(t)\right]$ and f is a nonlinear vector function $\left[f_1 \quad f_2 \quad \cdots \quad f_n\right]$, then the linear model in a given operating (steady-state) point $\left(u^s, y^s\right)$ can be generally obtained using formulae:

$$x'(t) = Ax(t) + Bu(t) \qquad (5)$$

with constant matrices A, B calculated as:

$$A = \begin{bmatrix} \dfrac{\partial f_1}{\partial x_1} & \dfrac{\partial f_1}{\partial x_2} & \cdots & \dfrac{\partial f_1}{\partial x_n} \\ \dfrac{\partial f_2}{\partial x_1} & \dfrac{\partial f_2}{\partial x_2} & \cdots & \dfrac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \vdots & \vdots \\ \dfrac{\partial f_n}{\partial x_1} & \dfrac{\partial f_n}{\partial x_2} & \cdots & \dfrac{\partial f_n}{\partial x_n} \end{bmatrix}_{x = x^s} \qquad (6)$$

$$B = \begin{bmatrix} \dfrac{\partial f_1}{\partial u_1} & \dfrac{\partial f_1}{\partial u_2} & \cdots & \dfrac{\partial f_1}{\partial u_m} \\ \dfrac{\partial f_2}{\partial u_1} & \dfrac{\partial f_2}{\partial u_2} & \cdots & \dfrac{\partial f_2}{\partial u_m} \\ \vdots & \vdots & \vdots & \vdots \\ \dfrac{\partial f_n}{\partial u_1} & \dfrac{\partial f_n}{\partial u_2} & \cdots & \dfrac{\partial f_n}{\partial u_m} \end{bmatrix}_{u = u^s}$$

As the reactor embodies astatic behaviour, it is not possible to compute the matrices in a chosen (steady-state) operating point. However, the linearization can be performed generally, resulting in a time-variant system:

$$x'(t) = A(t)x(t) + B(t)u(t) \qquad (7)$$

where the matrices $A(t)$, $B(t)$ are no longer constant but time-dependent. Using the formulas above, the originally nonlinear model of the reactor has been transformed into a linear time-variant model with the matrices $A(t)$, $B(t)$ of the form (8),

$$A(t) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ a_{21}(t) & a_{22}(t) & a_{23}(t) & 0 \\ a_{31}(t) & a_{32}(t) & a_{33}(t) & a_{34}(t) \\ 0 & 0 & a_{43} & a_{44}(t) \end{bmatrix} \quad (8)$$

$$B(t) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ b_{21}(t) & 0 & 0 & 0 \\ b_{31}(t) & 0 & b_{33}(t) & 0 \\ 0 & b_{42}(t) & 0 & b_{44}(t) \end{bmatrix}$$

where the particular ralations describing the coefficients (except for a_{43} all coefficients are time-dependent) are presented in the appendix. Generally, output from a linear system is defined as:

$$y(t) = Cx(t) + Du(t) \quad (9)$$

with matrices C, D defined in our case as:

$$C = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix}, D = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix} \quad (10)$$

(output for control purposes is only the temperature inside the reactor and the system is strictly proper).

Transfer function

If we have the matrices A, B, C, D of a linear (or linearized) model then it is possible to compute a transfer function (matrix in a MIMO case) of a system, which is a useful relation for control system analysis and design. It gives insight into input-output behaviour of the system variables and provides useful information, such as stability, non-minimum-phase behaviour, time-constants, gain etc. For constant matrices mentioned above, it can be computed simply as:

$$G(s) = C(sI_n - A)^{-1} B, \quad (11)$$

where s is the complex Laplace variable and I_n is the n-by-n identity matrix. As the reactor analyzed in this contribution is astatic, the linearized model is time-dependent and the "proper" transfer function introduced above cannot be computed. However, in a given time-moment t, the time-dependent matrices $A(t)$, $B(t)$ can be considered constant, i.e. it is possible to compute a "pseudo" transfer function (t.f.) describing instantaneous relation between the output and input signals. Then this "pseudo" transfer function is also time-dependent, but for a given time-moment it yields instantaneous transfer function of the system:

$$\tilde{G}(s,t) = C(sI_n - A(t))^{-1} B(t). \quad (12)$$

Using this idea and the relations (8)-(10) it is possible to compute a "pseudo" transfer function of the reactor.

Since the system generally has one output $T(t)$ and 4 inputs $\dot{m}_{FK}(t)$, $\dot{m}_v(t)$, $T_{FK}(t)$, $T_{vp}(t)$, the resultant t.f. is a vector of the 1-by-4 size:

$$\tilde{G} = \begin{bmatrix} G_{T/\dot{m}_{FK}} & G_{T/\dot{m}_v} & G_{T/T_{FK}} & G_{T/T_{vp}} \end{bmatrix}. \quad (13)$$

The first term $G_{T/\dot{m}_{FK}}$ describes the relation between the temperature inside the reactor $T(t)$ and the input flow rate of the chromium sludge $\dot{m}_{FK}(t)$. The other terms describes the relations between the temperature and the variables $\dot{m}_v(t)$, $T_{FK}(t)$, $T_{vp}(t)$ respectively. As stated earlier, the only practically manipulated variables are $\dot{m}_{FK}(t)$ and $\dot{m}_v(t)$. Therefore, next analysis is focused on the relation between these variables and the inner reactor temperature. The "pseudo" transfer function for these two input variables have these general forms (using (8)-(10) and (12)):

$$G_{T/\dot{m}_{FK}}(s,t) = \frac{B_3(t)s^3 + B_2(t)s^2 + B_1(t)s + B_0(t)}{s\left(s^3 + A_2(t)s^2 + A_1(t)s + A_0(t)\right)}, \quad (14)$$

$$G_{T/\dot{m}_v}(s,t) = \frac{B_1'(t)s + B_0'(t)}{s^3 + A_2(t)s^2 + A_1(t)s + A_0(t)}, \quad (15)$$

where relations for the time-dependent coefficients $A_i(t)$, $B_i(t)$ are given in the appendix (due to the limited space, for information only formulas concerning the most important t.f. $G_{T/\dot{m}_{FK}}$ are presented).

From the equations presented above, it can be seen that the relation between $T(t)$ and $\dot{m}_{FK}(t)$ is generally integrative whereas between $T(t)$ and $\dot{m}_v(t)$ only proportional. At present, the only practically manipulated variable is $\dot{m}_{FK}(t)$, therefore, the further investigation is focused on the analysis of the transfer function $G_{T/\dot{m}_{FK}}$. Although it would be also useful to investigate also $G_{T/\dot{m}_v}$, due to the limited space of this paper, this analysis is not provided.

Transfer function coefficients range

In order to determine the range of $G_{T/\dot{m}_{FK}}$ coefficients, a series of simulation experiments were performed in the MATLAB/Simulink environment. The variable $\dot{m}_{FK}(t)$ was changing in the admissible range from 0 to 3 [kg/s] with the step 0.1 and both maximum and minimum values of the $G_{T/\dot{m}_{FK}}$ coefficients were recorded. Summary of the achieved results is presented in Table 1.

Table 1: Range of $G_{T/\dot{m}_{FK}}$ Coefficients

Coefficient	Minimum value	Maximum value
B_3	-0.0426	-8.632×10^{-4}
B_2	9.944×10^{-5}	0.0016
B_1	5.221×10^{-7}	1.022×10^{-5}
B_0	-1.220×10^{-8}	5.810×10^{-11}
A_2	0.0051	0.0301
A_1	-8.777×10^{-6}	1.770×10^{-4}
A_0	-1.336×10^{-8}	1.787×10^{-7}

The table shows that some of the coefficients are very small and consequently they could be possibly neglected for the control system design.

Poles and zeros

Poles p_i of a system are defined as roots of the transfer function denominator whereas zeros z_i are given by the numerator roots. Given the range of coefficients from the Table 1, it is possible to compute also the range of poles and zeros of $G_{T/\dot{m}_{FK}}$. Results are summarized in Table 2 where the roots are divided into their real and imaginary parts for the case they are complex.

Table 2: Range of $G_{T/\dot{m}_{FK}}$ Poles p_i and Zeros z_i

z_i / p_i	Real min.	Real max.	Imag. min.	Imag. max.
z_1	0.0185	0.1601	0	0
z_2	-0.0062	-0.0062	0	0
z_3	-1.973×10^{-5}	0.0016	0	0
p_1	0	0	0	0
p_2	-0.0226	-0.0059	0	5.590×10^{-4}
p_3	-0.0067	0.0021	-5.590×10^{-4}	9.108×10^{-4}
p_4	-0.0018	8.464×10^{-4}	-9.108×10^{-4}	0

Zeros at (or very close to) the origin indicate derivative behaviour whereas poles at the same position signalize integrative properties. The table shows that one pole (p_1) is directly at the origin resulting in integrative behaviour of the temperature $T(t)$ with respect to $\dot{m}_{FK}(t)$. If the poles are located in the left part of the complex plane (their real parts are negative), the system is stable. From this point of view the table shows that generally the system embodies also instability. In addition, when the poles are complex (they also have imaginary parts), it indicates oscillatory behaviour. As revealed by the table, in some conditions the system may embody oscillatory behaviour, however absolute values of complex parts of the poles are relatively small which shows that this effect is not so significant. From the results, it can be also deduced that the system possesses non-minimum phase (NMP) behaviour – some of the zeros may become positive (unstable). Generally, NMP-systems are more difficult to control. Having the range of poles, it would be also possible to compute the range of time-constants and of the gain for the system. Although this information would be also interesting from the controller design point of view, due to the limited space it is not provided.

Discussion

The computed range of $G_{T/\dot{m}_{FK}}$ coefficients from Table 1 shows that for the control design, some of the coefficients could be neglected (e.g. B_0, possibly also B_1, A_0). Consequently, the linear time-variant model will be of a simpler form, which is useful.

Suggested control strategies

Havig the approximate uncertainty intervals of the $G_{T/\dot{m}_{FK}}$ coefficients, it suggests using the robust control approach (e.g. Morari and Zafirou 1989). As parameters of the linearized model change, an alternative idea could also be the usage of adaptive control strategies (e.g. Åström and Wittenmark 1989). Another possible approach which proved to be successful is the predictive control (Srinivasan and Bonvin 2007, Sámek and Macků 2007). At present, the system is controlled only as a Single Input – Single Output (the temperature inside the reactor is controlled only by means of the chromium sludge input flow rate). It would be useful to include also the coolant flow rate $\dot{m}_v(t)$ as a manipulated variable and use the MIMO control design strategies (e.g. Skogestad and Postlethwaite 1996).

CONCLUSION

This contribution was focused on the analysis of a reactor used for the tanning waste recovery. The analysis was made by both theoretical and simulation means in order to provide useful information for optimal controller design. The nonlinear model of the system was derived and transformed into a linear time-variant system which was used for further investigation of the process properties. The range of transfer function coefficients was determined and possible control strategies were outlined. In this work, the main focus was on the relationship between the controlled variable – temperature inside the reactor and the manipulated variable given by the chromium sludge input flow rate. From the controller design point of view, it would be useful to investigate also the relationship between the second practically possible manipulated variable – the coolant mass flow rate and the temperature.

ACKNOWLEDGEMENTS

APENDIX

Formulas for coefficients of the linear time-variant model

$$a_{21}(t) = -\frac{\dot{m}_{FK}(t)}{m^2(t)}\left(1 - a_{FK}(t)\right), \; a_{22}(t) = -\frac{\dot{m}_{FK}(t)}{m(t)} - Ae^{-\frac{E}{RT(t)}}$$

$$a_{23}(t) = -Ae^{-\frac{E}{RT(t)}}\left(\frac{E}{RT^2(t)}\right)a_{FK}(t)$$

$$a_{31}(t) = -\frac{1}{m^2(t)c_R}\left[\dot{m}_{FK}(t)\left(c_{FK}T_{FK}(t) - c_R T(t)\right) + KS\left(T_v(t) - T(t)\right)\right]$$

$$a_{32}(t) = \frac{Ae^{-\frac{E}{RT(t)}}\Delta H_r}{c_R} \; , \; a_{34}(t) = \frac{KS}{m(t)c_R}$$

$$a_{33}(t) = \frac{A\Delta H_r E a_{FK}(t)}{RT^2(t)c_R}e^{-\frac{E}{RT(t)}} - \frac{KS}{m(t)c_R} - \frac{\dot{m}_{FK}(t)}{m(t)}$$

$$a_{43} = \frac{KS}{m_{vR}c_v} \; , \; a_{44}(t) = -\frac{1}{m_{vR}}\left(\dot{m}_v(t) + \frac{KS}{c_v}\right)$$

$$b_{21}(t) = \frac{1 - a_{FK}(t)}{m(t)} \; , \; b_{31}(t) = \frac{1}{m(t)}\left(\frac{c_{FK}T_{FK}(t)}{c_R} - T(t)\right)$$

$$b_{33}(t) = \frac{\dot{m}_{FK}(t)c_{FK}}{m(t)c_R} \; , \; b_{42}(t) = \frac{1}{m_{vR}}\left(T_{vp}(t) - T_v(t)\right)$$

$$b_{44}(t) = \frac{\dot{m}_v(t)}{m_{vR}}$$

Formulas for transfer function coefficients

$$B_3(t) = b_{31}(t)$$

$$B_2(t) = a_{32}(t)b_{21}(t) - b_{31}(t)a_{22}(t) + a_{31}(t) - a_{44}(t)b_{31}(t)$$

$$B_1(t) = a_{21}(t)a_{32}(t) - a_{31}(t)a_{22}(t) - \cdots$$

$$\cdots - a_{44}(t)\left[a_{32}(t)b_{21}(t) - b_{31}(t)a_{22}(t) + a_{31}(t)\right]$$

$$B_0(t) = -a_{44}(t)\left[a_{21}(t)a_{32}(t) - a_{31}(t)a_{22}(t)\right]$$

$$A_2(t) = -a_{44}(t) - a_{22}(t) - a_{33}(t)$$

$$A_1(t) = a_{22}(t)a_{44}(t) - a_{34}(t)a_{43} - a_{32}(t)a_{23}(t) + \cdots$$

$$\cdots + a_{33}(t)a_{44}(t) + a_{22}(t)a_{33}(t)$$

$$A_0 = -a_{22}(t)a_{33}(t)a_{44}(t) + a_{22}(t)a_{34}(t)a_{43} + \cdots$$

$$\cdots + a_{32}(t)a_{23}(t)a_{44}(t)$$

REFERENCES

Aloy, M. and B. Vulliermet. 1998. "Membrane technologies for the treatment of tannery residual floats." *Journal of the Society of Leather Technologies & Chemists* 82, No.4 (Jul-Aug), 140-142.

Åström, K.J. and B. Wittenmark. 1989. Adaptive Control. Addison-Wesley, Reading, MA.

Cabeza, L.F., Taylor, M.M., Dimaio, G.L., Brown, E.M., Mermer, W.N., Carrio, R., Celma, P.J. and J. Cot. 1998. "Processing of leather waste: pilot scale studies on chrome shavings. Isolation of potentially valuable protein products and chromium." *Waste Management* 18, No.3, 211-218.

Kolomazník, K., Mládek, M., Langmaier, F., Taylor, M., Diefendorf, E.J., Marmer, W.N., Brown, E. M. and L. Tribula. 1996. CR Patent 280 655, Czech Republic (Feb).

Macků, L. 2004. "Control design for the preparation of regenerate for tanning". Ph.D. thesis. Tomas Bata University in Zlín, Czech Republic (in Czech).

Macků, L. 2005. "Modeling of tanning salts regeneration process." In *Proceedings of the 15th Int. Conf. Process Control 2005* (High Tatras, Slovakia, Jun.7-10). Bratislava: Slovak University of Technology, 127/1-127/4.

Morari, M. and E. Zafirou. 1989. Robust Process Control. Prentice Hall, Englewood Cliffs, New Jersey.

Sámek, D. and L. Macků. 2007. "Simulation of model predictive control of semi-batch reactor." In *Proc. Int. Symp. on Systems Theory SINTES 13* (Craiova, Romania, Oct.18-20). Craiova: University of Craiova, 180-185.

Skogestad, S. and I. Postlethwaite. 1996. Multivariable Feedback Control: Analysis and Design. John Wiley & Sons, Chichester, England.

Srinivasan, B. and D. Bonvin. 2007. "Controllability and stability of repetitive batch processes." *Journal of Process Control* 17, 285-295.

Tiravanti, G., D. Petruzzelli and R. Passino. 1996. "Low and non waste technologies for metals recovery by reactive polymers." *Waste Management* 16, No.7, 597-605.

Tiravanti, G., D. Petruzzelli and R. Passino. 1997. "Pretreatment of tannery wastewaters by an ion exchange process for Cr(III) removal and recovery." *Water Science and Technology* 36, No.2-3, 197-207.

AUTHOR BIOGRAPHIES

FRANTIŠEK GAZDOŠ was born in Zlín, Czech Republic, and graduated from the Technical University of Brno in 1999 with MSc. degree in Automation. He then followed studies of Technical cybernetics at Tomas Bata University in Zlín, obtaining Ph.D. degree in 2004. Now he works as a senior lecturer in the Department of Process Control, Faculty of Applied Informatics of the same University. E-mail: gazdos@fai.utb.cz.

LUBOMÍR MACKŮ was born in Uherské Hradiště, Czech Republic and graduated from the Technical University of Brno in 1996 with MSc. degree in Process Control Engineering. He followed studies of Technical cybernetics at Tomas Bata University in Zlín, obtaining Ph.D. degree in 2004. Now he woks as a senior lecturer in the Department of Electrotechnics and Measurements, Faculty of Applied Informatics of the same University. E-mail:macku@fai.utb.cz.

SIMULATION AND ROBUST CONTROL OF CONTINUOUS TIME CIRCULATING REACTOR

Zdenka Prokopová and Zuzana Oplatková
Department of Applied Informatics
Tomas Bata University in Zlín,
Nad Stráněmi 4511, 760 05 Zlín, Czech Republic
E-mail: prokopova@fai.utb.cz

KEYWORDS

Modelling, Simulation, Circulating reactor, Algebraic approach, Robust control, Uncertainty.

ABSTRACT

The aim of presented paper is analysis, mathematical modelling, simulation and control of hydrogenation process. Cyclohexane, as an important base material for polyamide manufacturing, makes by catalytic hydrogenation of benzene. A large number of reaction heat transfers during hydrogenation process. Therefore, benzene and cyclohexane mixture hydrogenation proceed under hydrogen excess. We suppose the hydrogenation mechanism of benzene as a first-order reaction. Matlab-Simulink toolbox was used for robust control simulation with application to the circulating reactor. Robust controllers are obtained via solutions of Diophantine equations in the ring of stable and proper rational functions. Uncertainty and robustness are studied through the infinity norm H_∞. For controller design was proposed a scalar parameter m>0 as a tuning knob.

INTRODUCTION

The behavior of real chemical process is obviously too complex for complete mathematical analysis. We are obliged to use an approximate description which is able to deliver sufficiently accurate results. Cardinal simplifications of the mathematical description can be obtained by the following supposals:

- the process does not change its properties during the observation period,
- the relation between input and output variables of the process is linear,
- all process variables can be measured continuously and have a continuous-time trend.

A useful and interesting application of reactor design for chemical engineers can be seen in (Froment and Bischotf 1990), (King and Winterbottom 1998). The treatment of biochemical and polymer reaction engineering is described very extensively in (Nauman 2002). There are emphases on numerical solutions which are needed for most practical problems in chemical reactor design. Sophisticated numerical techniques are

rarely necessary. The aim is to make the techniques understandable and easily accessible and allow continued focus on the chemistry and physics of the process.

Robust controllers and plant uncertainty became a useful and popular discipline in control theory during the last decade. The necessity of robust control was naturally developed by the situation when the nominal plant (used in control design) differs from the real (perturbed) one. A suitable tool for parameter uncertainty is the infinity norm H_∞. Hence, a polynomial description of transfer functions had to be replaced by another one. A convenient description adopted from (Vidyasagar 1995), (Kučera 1993), (Doyle et al 1992) is a factorization approach where transfer functions are expressed as a ratio of two Hurwitz stable and proper rational functions (R_s). Then conditions of robust stability can be easily formulated in algebraic parlance and all controllers are obtained via linear Diophantine equations in an appropriate ring.

MODEL DESCRIPTION

Benzene and cyclohexane mixture hydrogenation proceed in continuous time circulating reactor under hydrogen excess. The presented mathematical model should respect the following conditions:

- Reaction mixture is perfectly mixed by circulation.
- The volume of reaction mixture is constant.
- All technological parameters in reactor are constant.
- Heat transfer on the both sides of wall is ideal.

The hydrogenation mechanism of benzene is supposed to be a first-order reaction:

$$r_B = -k \cdot c_B \tag{1}$$
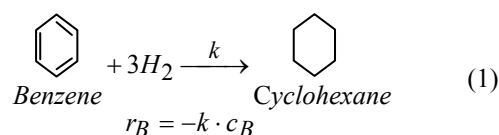
where k is a constant of proportionality known as the *rate constant* and r_B is *the rate of the reaction*. There is not the rate at which a particular component reacts. Components benzene and hydrogen are consumed by the reaction and thus are "formed" at a negative rate. The material balance for the component benzene satisfies the

following differential equation (for constant volume V of reactor):

$$q_v.c_{Bv} = q.c_B + V.k.c_B + V\frac{dc_B}{dt} \qquad (2)$$

where q_v and q are input and output flow and c_{Bv} and c_B are input and output concentration of component benzene.

The temperature balance in the reactor satisfies the differential equation:

$$q_v.\rho_v.c_{pv}.T_v + (-\Delta H_R)V.k.c_B = \\ q.\rho.c_p.T + F.\alpha(T - T_x) + V.\rho.c_p\frac{dT}{dt} \qquad (3)$$

where the relation $(-\Delta H_R)V.k.c_B$ is the heat generated by the reaction and $F.\alpha(T - T_x)$ represents the heat transfer into surroundings.

The rate constant k for elementary reactions is expressed by Arrhenius equation:

$$k = k_0 T^n \exp\left(\frac{-E}{RT}\right) \qquad (4)$$

where parameter $n=0$ (or $n=0.5$, 1 for the special cases of reactions) depends on used theoretical model. E is *activation energy* and the fraction E/R is called an *activation temperature.*

After some mathematical manipulations the equation (2) of the benzene concentration variation takes the form:

$$\frac{dc_B}{dt} = \frac{q_v}{V} \cdot c_{Bv} - \frac{q}{V} \cdot c_B - k_0 e^{-\frac{E}{RT}} \cdot c_B \qquad (5)$$

Similarly, variation of the reaction mixture temperature in the reactor is:

$$\frac{dT}{dt} = \frac{q_v}{V}T_v + (-\Delta H_R).\frac{k.c_B}{\rho.c_p} - \frac{q}{V}T \\ - \frac{F.\alpha}{V.\rho.c_p}(T - T_x) \qquad (6)$$

The time response of the reaction mixture concentration depends on the input flow of the benzene, the concentration of input stream c_{Bv}, the volume of reactor filling, the specific heat capacity of reaction mixture c_p and the rate constant k.

The solution of ordinary differential equation (5) requires an initial condition: $c_{Bv} = c_{B0}$ at $t=0$.

From all mentioned parameters the input flow stream into the reactor q_v and its initial concentration c_{Bv} can be changed.

CONTOL DESIGN

The plant uncertainty can be expressed conveniently in terms of the fractional description. It means that linear transfer functions are no more represented as a ratio of two polynomial but two elements of an another ring. For the purposes of robust control in the sense of this contribution any transfer function $H(s)$ of a linear system can be expressed as a ratio of two elements:

$$H(s) = \frac{b(s)}{a(s)} = \frac{B(s)}{A(s)} \\ A(s) = \frac{a(s)}{m(s)}; B(s) = \frac{b(s)}{m(s)} \qquad (1) \\ m(s) = (s + m)^n; n = \max\{\deg a; \deg b\}$$

where a, b are polynomials. Elements $A, B,...$ constitute a sub ring $R_m(s)$ of the ring of all Hurwitz stable and proper rational functions $R_s(s)$. A class of PID like controllers is generated by first and second order systems and all controlled plants will be approximated by such systems. Systems of the second order will be approximated by the following fractional representation in $R_m(s)$:

$$H(s) = \frac{b_0}{a_2 s^2 + a_1 s + a_0} = \\ = \frac{\dfrac{b_0}{(s+m)^2}}{\dfrac{a_2 s^2 + a_1 s + a_0}{(s+m)^2}} = \frac{B(s)}{A(s)} \qquad (2)$$

For $m = 0$ subring $R_m(s)$ expands to the traditional ring $R_s(s)$ (see Vidyasagar, 1985). An infinity norm (convenient for uncertainty) in both rings is defined by

$$\|H\| = \sup_{\text{Re}\,s \geq 0} |H(s)| = \sup_{\omega \in E}|H(j\omega)| \\ \|H_1 H_2\| = \left\|\begin{matrix} H_1 \\ H_2 \end{matrix}\right\| \sup_{\text{Re}\,s \geq 0}\left\{H_1(s)^2 + |H_2(s)|^2\right\}^{\frac{1}{2}} \qquad (3)$$

This norm is the radius of the smallest circle containing the Nyquist plot of the transfer function. Almost all mathematical models differ from physical systems. Let $H(s) = \frac{B(s)}{A(s)}$ be a nominal plant and consider a family of perturbed systems $H'(s) = \frac{B'(s)}{A'(s)}$ where

$$\|A - A'\| \leq \varepsilon_1 \quad \|B - B'\| \leq \varepsilon_2 \\ \text{or} \quad \|A - A' \quad B - B'\| \leq \varepsilon \qquad (4)$$

Control systems depicted in Figure 1. and Figure 2. are considered for design of a robust FB and FBFW controllers.
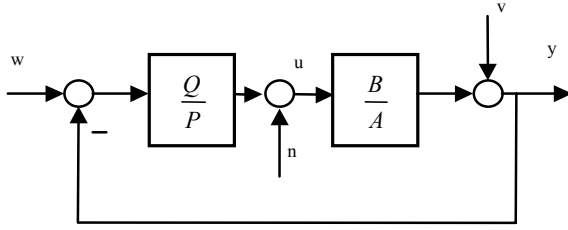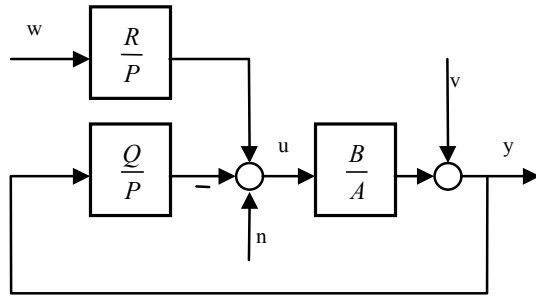
Figure 1. Feedback control system

Figure 2. Feedback-feedforward control system.

Transfer functions $w = \dfrac{G_w}{F_w}$ and $v = \dfrac{G_v}{F_v}$ represent the reference and disturbance signal, respectively. The control design in $R_m(s)$ is proposed in (Prokop and Corriou 1997), (Prokop and Dostál 1995). For the given nominal plant $H(s)$ in the form of (7) it consists of the following steps:

Step 1: All stabilizing (feedback) controllers are given by all solutions of the equation:

$$AP + BQ = 1 \qquad (5)$$

in the parameter form expressed by the fraction:

$$\frac{Q}{P} = \frac{Q_0 - AT}{P_0 + BT} \qquad (6)$$

where P_0, $Q_0 \in R_m(s)$ are particular solutions and T is arbitrary in $R_m(s)$.

Step 2: Feedback controllers (if necessary) are given in a similar way by all solutions of

$$F_w S + BR = 1 \qquad (7)$$

Step 3: Compute $P \in R_m(s)$ such that

F_w divides P for asymptotic tracking problems
F_v divides P for disturbance rejection problems

$$(8)$$

Step 4: For perturbed plants (10) choose such P, Q in (11), (12) which fulfils the conditions

$$\varepsilon_1 \|P_0 + BT\| + \varepsilon_2 \|Q_0 - AT\| \le 1$$

$$\text{or} \quad \varepsilon \left\| \begin{array}{c} P_0 + BT \\ Q_0 - AT \end{array} \right\| \le 1 \qquad (9)$$

Steps 1 - 3 represent the first and simplest case of robust design. All solutions of Diophantine equations in $R_m(s)$ are expressed as functions of the parameter $m > 0$. The value of this parameter strongly influences dynamics as well as robustness of the proposed control system. For a deeper insight into robustness the notions of the sensitivity function:

$$\in = \frac{y}{v} = \frac{1}{1 + HC_b} = A(P_0 + BT) \qquad (10)$$

can be utilized as the second possibility. Similarly, for SISO systems $\in$ is also a function of a scalar parameter $m > 0$ and it can be minimized by a simple scalar optimization method. In this way the "most robust" controller for a given nominal plant is obtained. The third case covers situations where perturbed plants are known and the norms in (10) can be evaluated. Then inequalities (15) yield generally nonlinear inequalities for the scalar parameter $m > 0$ which can be solved numerically.

SIMULATION EXPERIMENTS

The continuous-flow circulation reactor for catalytic hydrogenation of benzene with $V = 0.5$ m^3 volume was used for the simulation of the concentration and temperature characteristics. Initial input flow into the reactor was $q_v = 0.0067$ m^{3}s^{-1} with the initial concentration of benzene $c_{Bv} = 1.9$ kg/m^3. Initial rate constant was considered $k = 1.616$.e14 s^{-1}, density of mass $\rho = 985$ kg m^{-3}, specific heat capacity $c_p = 4.050$ J/kg.K, transfer heat coefficient $E = 435.00$ W/m^2.K, surface $F = 5.5$ m^2, activation energy $E = 4.8$e4 J mol^{-1}, initial mass temperature $T_v = 320$ K and temperature of surrounding $T_x = 290$ K.

Steady-State Analysis Results
The reactor steady-state characteristics were obtained by solution of equations (4)-(6). The initial conditions were computed by a standard optimization method - iterative procedure.

Dynamic Analysis Results
For dynamic analysis purposes one input and one output choices were considered. Then were defined input u and output y as deviations from their steady-state values – $u(t) = q(t) - q^s(t)$ and $y(t) = c_B(t) - c_B^s(t)$.
Graphical interpretation of simulation experiments – the output y time responses to input u step changes are depicted on Figure 3.
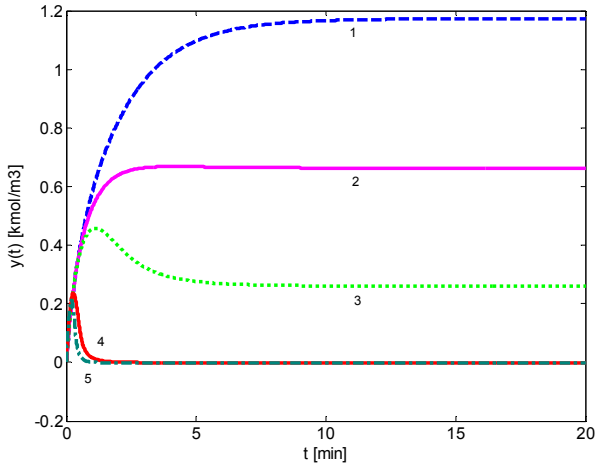
Figure 3: The output y time responses to input u step changes, u=-0.08(*1*), -0.07(*2*), -0.05(*3*), 0.05(*4*), 0.08(*5*)

Model approximation

For control requirement, described continuous time circulating reactor was approximated by the second order system in the form:

$$H_1(s)=\frac{1.9}{71s^2+10s+0.5}\approx\frac{B(s)}{A(s)} \qquad (11)$$

The approximation was obtained by a four-parameter method adopted from (Astrom and Hagglund 1995) applied to step responses.

For the additional testing of designed robust controllers there was used another nominal system as second order system with unit parameters:

$$H_2(s)=\frac{1}{1s^2+1s+1}\approx\frac{B(s)}{A(s)} \qquad (12)$$

Control results

One of the main problems of the catalytic hydrogenation of benzene in continuous-flow circulation reactor is progressive degeneration of reaction surfactant. This is cause that the reaction is slowing down, the rate constant k is decreasing and the concentration of benzene in output mixture is increasing. The aim is to keep the concentration of benzene under the critical value. One of the possibilities how to solve the problem is reduction of input flow q of the benzene and cyclohexane mixture into the reactor.

In spite of the situation, it is desirable to find a robust controller which is "sufficiently acceptable". More precisely, the control goal is to design (in a robust way) a FB and FBFW controllers according to above mentioned methodology for nominal plants and choose such values of $m>0$ which give suitable control responses.

Both FB and FBFW control schemes gave the same optimal value of user defined "tuning knob" m. Derivation of controller parameters does not bring any problems.

Simulation 1: Nominal plant: H_1, m=2, set point w=0.002, FB controller: $\frac{Q(s)}{P(s)}=\frac{2.87s^2+2.09s+0.53}{0.01s^2+0.05s}$.
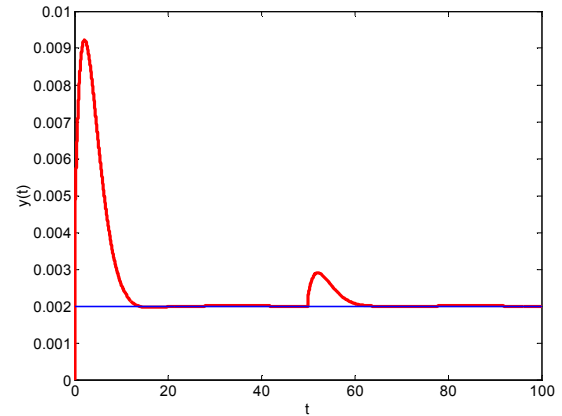
Figure 4. Simulation 1 – output y and reference w.

Simulation 2: Nominal plant: H_1, m=2, set point w=0.002, FBFW controller:

$$\frac{Q(s)}{P(s)}=\frac{2.87s^2+2.09s+0.53}{0.01s^2+0.05s}, \frac{R(s)}{P(s)}=\frac{0.53s^2+1.05s+0.53}{0.01s^2+0.05s}.$$

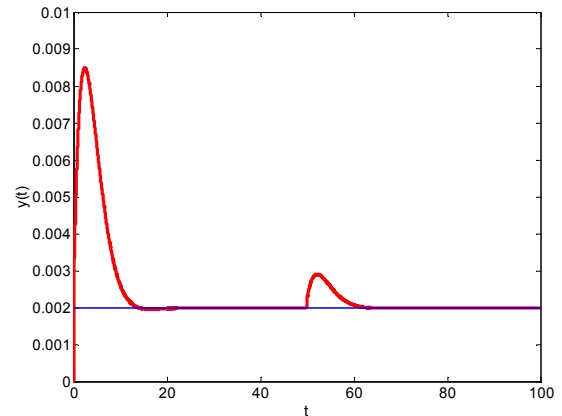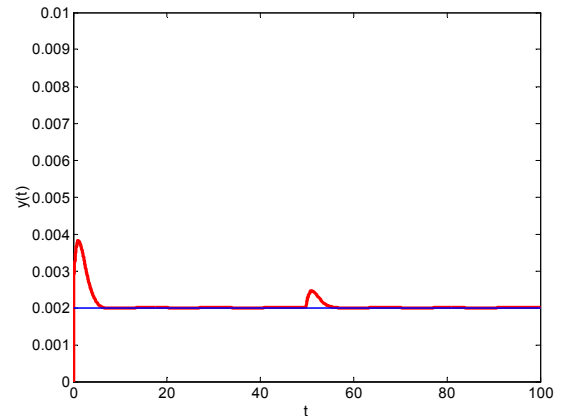Figure 5. Simulation 2 – output y and reference w.

Simulation 3: Nominal plant: H_1, m=2, set point w=0.002, FB controller: $\frac{Q(s)}{P(s)}=\frac{12.05s^2+16.81s+8.42}{0.01s^2+0.11s}$.

Figure 6. Simulation 3 – output y and reference w.

Simulation 4: Nominal plant: H_1, m=2, set point w=0.002, FBFW controller: $\dfrac{R(s)}{P(s)} = \dfrac{2.51s^2 + 8.42s + 8.42}{0.01s^2 + 0.05s}$,

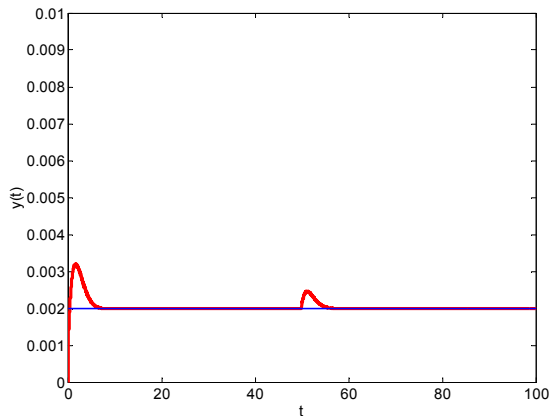$\dfrac{Q(s)}{P(s)} = \dfrac{12.05s^2 + 16.81s + 8.42}{0.01s^2 + 0.11s}$.

Figure 7. Simulation 4 – output _y_ and reference _w_.

Simulation 5: Nominal plant: H_2, m=1, set point w=0.002, FB controller: $\dfrac{Q(s)}{P(s)} = \dfrac{2s^2 + 1s + 1}{1s^2 + 3s}$.
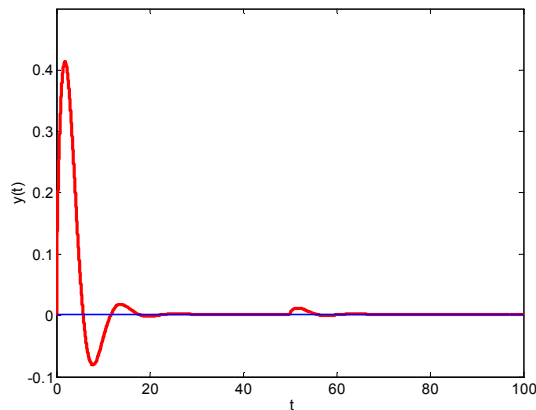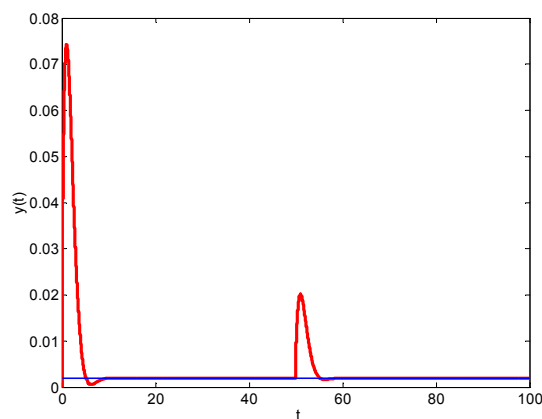

Figure 8. Simulation 5 – output y and reference w.

Simulation 6: Nominal plant: H_2, m=2, set point w=0.002, FB controller: $\dfrac{Q(s)}{P(s)} = \dfrac{16s^2 + 25s + 16}{1s^2 + 7s}$.

Figure 9. Simulation 6 – output _y_ and reference _w_.

Figure 4 to Figure 9 show the results of _Simulation 1_ to _Simulation_ 6 of proposed controllers for various nominal plants (11) and (12) and feedback and feedback-feedforward scheme, respectively. Simulation results document that there exist possibility to use robust approach for control of systems with unknown parameters. It is clear that behavior of controlled output value is better when the parameters of nominal plant are closer to real process.

CONCLUSION

The proposed methodology enables to tune and influence the robustness and control behavior by a single scalar positive parameter. Control laws for first order systems yield a class of PID - like controllers. As we can see on the figures the control results for only feedback and feedback feedforward controllers are very similar in this case. Only the overshooting is slightly smaller in the feedback case. A design method based on the fractional representation was developed for SISO systems generally. All computations and simulation were performed in the MATLAB+SIMULINK environment.

REFERENCES

Astrom, K.J., Hagg1und, T. 1995. _PID Controllers: Theory, Design and Tuning,_ 343 p. Instmment Society of America.

Dean, A., J. 1992. _Lange's Handbook of Chemistry_. 14th edition. McGraw-Hill, INC, New York.

Doyle J.C., Francis B.A., Tannenbaum A.R. 1992. _Feedback Control Theory._ Maxwell McMillan.

Eaton, A.D., L.S. Clesceri and A.E. Greenberg. 1995. "Standard methods of the examination of water and wastewater". In: _19th Edn. APHA_, Washington D.C.

Felder, R. M. and Rousseau, R. W. 2000. _Elementary Principles of Chemical Processes,_ 3rd ed., Wiley, New York.

Froment, F. and K. B. Bischotf. 1990. _Chemical Reactor Analysis and Design,_ 2nd ed., Wiley, New York..

Janáčová, D.; Kolomazník, K.; Vašek, V. et al. 1997. "Washing Processes Optimization". In _Proceedings of International Union of Leather Technologists and Chemists Societies_, pp. 750, Centenary Congress. British Leather Centre. London.

King, M. B. and M. B. Winterbottom. 1998. _Reactor Design For Chemical Engineers,_ Chapman & Hall, London..

Kolomazník, K.; Janáčová, D. & Langmaier, F., et. al. 1998. "Enzymatic Digestion of ChromeTanned Solid Waste". In _Proceedings of the 1st International Fur Congress_. p. 134-146. Kastoria, Macedonia Research Center, Greece

Kučera, V. 1993. "Diophantine Equations in Control – A Survey", _Automatica,_ 29, No.6, pp. 1361-75.

Nauman, E. B. 2002. _Chemical reactor design, optimization, and scaleup._ McGRAW-HILL, New York. ISBN: 0-07-137753-0.

Prokop, R., Corriou, J.P. 1997. "Design and Analysis of Simple Robust Controllers". *Int. JournalofControl,* Vol. 66, No. 6, pp. 905-921.

Prokop, R., Dostál, P. 1995. "Design of Simple Robust Controllers". In *10th International Conference on PC'95*. Tatranské Matliare, pp. 103-108.

Prokopová, Z. 2005. "Analysis and mathematical models of rotary reactors". In *15th International Conference on PC'05*. Bratislava : STU, pp. 501-504. ISBN 80-227-2235-9.

Rangel-Serrano, A., M.V. Maldonado, and K Kösters. 2003. "Characterization of Waste Materials in Tanneries for Better Ecological Uses". *The Journal of the American Leather Chemists Association*, Vol. 98, No. 2., pp. 43-48.

Vidyasagar, M. 1985. *Control System Synthesis: A Factorization Approach.* MlT Press, Cambridge.

AUTHOR BIOGRAPHIES
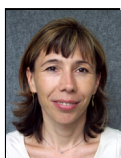
ZDENKA PROKOPOVÁ was born in Rimavská Sobota, Slovak Republic. She graduated from Slovak Technical University in 1988, with a master's degree in automatic control theory. Doctor's degree she has received in technical cybernetics in 1993 from the same university. Since 1995 she is working in Tomas Bata University in Zlin, Faculty of Applied Informatics. She is now working there as an associating professor. Research activities: mathematical modeling, simulation, control of technological systems, programming and application of database systems. Her e-mail address is : prokopova@fai.utb.cz.

ZUZANA OPLATKOVÁ was born in Zlín, Czech Republic. She graduated from Tomas Bata University in Zlín in 2003, with a master's degree in automatic control theory. Since 2003 she is working in Tomas Bata University in Zlín, Faculty of Applied Informatics as assistant. Research activities: evolution algorithms, symbolic regression, analytic programming, genetic programming. Her e-mail address is : oplatkova@fai.utb.cz.

ARTIFICIAL NEURAL NETWORKS IN PREDICTION AND PREDICTIVE CONTROL

David Samek
Dept. of Production Engineering
Tomas Bata University in Zlin
Mostni 5139, 76001 Zlin
E-mail: samek@ft.utb.cz

Petr Dostal
Department of Process Control
Tomas Bata University in Zlin
Nad Stranemi 4511, 76005 Zlin
E-mail: dostalp@fai.utb.cz

KEYWORDS

Artificial neural networks, prediction, model predictive control.

ABSTRACT

In this contribution the three various artificial neural networks are tested on CATS prediction benchmark. The results are compared and evaluated. Furthermore, these artificial neural networks are tested in model predictive control on the t-variant system. The aim of this paper is to present and compare artificial neural networks as interesting way how to model and predict nonlinear systems even with t-variant parameters. The key features of this paper are emphasis of the computational costs of the selected predictors and usage of adaptive linear network which offers short learning times and remarkable prediction error.

INTRODUCTION

The increasing demand on the quality, reliability, and economical profits leads to using of new modeling and control methods in the process industry. In past few decades the predictive control techniques have become very popular. One of the most used approaches is the Model Predictive Control (MPC) method (Camacho and Bordons 1995).

The appropriate predictive model is a key question in nonlinear model predictive control. The predictive models can be divided into two main groups (Verdunen and Jong 2003): white box models and black box models. The white box modeling is established on a prior knowledge of mathematic description of basic physical rules of controlled process. White box models are excellent for process modeling and product development. The model constants have a physical meaning and are not dependent on process design. The main disadvantage of white box models is the time of development and higher complexity. Conversely, black box models such as artificial neural network (ANN) and fuzzy logic models are data-driven. They provide general method for process dynamics description from input-output data. First and foremost, the learning ability makes artificial neural networks versatile, user friendly and powerful tool for many practical applications (Hussain 1999).

Many predictive control techniques based on MPC, which use artificial neural network as a predictor, are established on multilayer feed-forward neural networks (Hagan et al. 2002; Kanjilal 1995). In spite of the fact that the multilayer feed-forward neural networks (MFFNNs) have many advantages, such as simple design and scalability, they have also many drawbacks, such as long training times and choice of an appropriate learning stop time (the over-learning versus the early stopping). Nevertheless, there are quite a number of ANN types suitable for the modeling and prediction (Liu 2001; Meszaros et al. 1999; Chu et al. 2003). Moreover, features of these ANNs exceed abilities of the MFFNN in many cases. One of these ANNs is ADALINE (ADAptive LInear NEuron). What is more, ADALINE has one special feature – adaptivity. Owing to its simple structure it offers interesting way how to design adaptive neural predictor with reasonable computational demands. This paper is organized as follows: In the beginning multilayer feed-forward neural networks and adaptive linear networks are briefly introduced. Then the methodology of the simulations is explained, after that the results are presented and the paper is concluded by final remarks.

MULTILAYER FEED-FORWARD NEURAL NETWORKS

Multilayer feed-forward neural networks were derived by generalization from Rosenblatt's perceptron, thus they are often called multilayer perceptrons (MLP). This type of artificial neural networks uses supervised training. One of the most known methods of supervised training is backpropagation algorithm; hence these ANNs are sometimes also called backpropagation networks.

In the MFFNN the signals flow between the neurons only in the forward direction i.e. towards the output. Neurons in MFFNN are organized in layers and neurons of the certain layer can have inputs from any neurons of the earlier layer. The ability to predict of ANN is determined by capability of modeling of certain process. By applying the Kolmogorov theorem it was proved that for general function approximation is sufficient two-layer MFFNN (one hidden layer) if non-polynomial transfer functions are used and the hidden layer has enough neurons (Leshno et al. 1993).
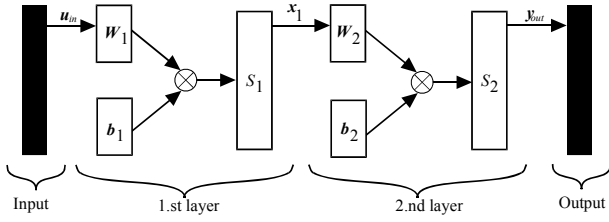
Figure 1: Simplified Scheme of Two-layer MFFNN

The two-layer MFFNN, which contains one output layer and one hidden layer, is depicted in the figure 1 (this structure is implemented in this paper). This MFFNN can be described by two equations:

$$x_1 = S_1(b_1 + W_1 \cdot u_{in}) \qquad (1)$$

$$y_{out} = S_2(b_2 + W_2 \cdot x_1) \qquad (2)$$

Where y_{out} is the network output vector, S_i is transfer function of i-th layer, b_i is bias vector of i-th layer, W_i is weighting matrix of i-th layer, x_1 is output vector of the first layer and u_{in} is the network input vector.

ADAPTIVE LINEAR NETWORKS

ADALINE was described by B. Widrow and M.E. Hoff as an adaptive threshold logic element in 1960 (Widrow and Hoff 1960). Though, the original version of ADALINE had only simple two-state threshold transfer function with the range of function {-1;+1}, nowadays ADALINE is also used with linear transfer function (Arbib 2002; Demuth and Beale 2002). Although this structure has limited skills, it is possible to connect more of ADALINEs together to obtain a MADALINE (Multiple ADALINE).

Despite the fact that ADALINEs are able to solve only linearly separable problems, it has been shown in practice that they can approximate nonlinear functions with sufficient accuracy while using enough number of neurons (Freeman and Skapura 1991).

Because of their main advantage, that is very fast learning, they have many practical applications, e.g. noise reduction, signal processing and signal prediction in control and communication systems. Simplified scheme of ADALINE is shown in figure 2.
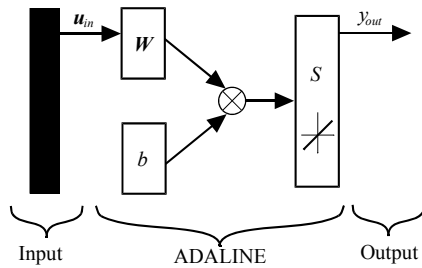

Figure 2: Simplified Scheme of ADALINE with Linear Transfer Function

The learning procedure is based on an iterative search process, where performance feedback is used to guide the search process. In other words, a designer "trains" the system by "showing" it examples of inputs and the respective outputs. In this way, system competence is directly and quantitatively related to the amount of experience the system was given. The most popular learning method is simple LMS (Least Mean Square) algorithm (Widrow and Hoff 1960), often called the Widrow Hoff Delta Rule (Rumelhart et al. 1986), which is adopted in this paper. This method is based on the minimization of Mean Square Error (MSE).

METHODOLOGY

The three different artificial neural networks were tested in prediction problem. There were chosen two different variants of two-layer feed-forward neural network. The first structure followed from Kolmogorov theorem (Leshno et al. 1993) and had hyperbolic tangent transfer function in both layers (this structure will be in the following text denoted as *mffnntt*), while the second structure uses very common layout with hyperbolic tangent transfer function in hidden layer and linear transfer function in output layer (this structure will be in the following text denoted as *mffnntp*). The third predictor was based on ADALINE (this structure will be in the following text denoted as *adaline*).

The artificial neural networks were tested in two tasks. The first task was the prediction of time series defined by CATS benchmark (Vandewalle et al. 2007; Lendasse et al. 2004). Then the artificial neural networks were used as the predictors of Model Predictive Controller.

CATS Benchmark

The CATS (Competition on Artificial Time Series) benchmark was designed in order to compare various prediction methods. The task of the predictor is to predict 100 missing values of the 5000 artificial time series data. The missing values are divided in 5 blocks – 981 to 1,000; 1,981 to 2,000; 2,981 to 3,000; 3,981 to 4,000; 4,981 to 5,000.

CATS benchmark uses two criterions E_1 and E_2:

$$E_1 = \frac{\sum_{t=981}^{1000}(e_t - \hat{e}_t)^2}{100} + \frac{\sum_{t=1981}^{2000}(e_t - \hat{e}_t)^2}{100} +$$
$$+ \frac{\sum_{t=2981}^{3000}(e_t - \hat{e}_t)^2}{100} + \frac{\sum_{t=3981}^{4000}(e_t - \hat{e}_t)^2}{100} + \qquad (3)$$
$$+ \frac{\sum_{t=4981}^{5000}(e_t - \hat{e}_t)^2}{100}$$

$$E_2 = \frac{\sum_{t=981}^{1000}(e-\hat{e})^2}{80} + \frac{\sum_{t=1981}^{2000}(e-\hat{e})^2}{80} +$$
$$+ \frac{\sum_{t=2981}^{3000}(e-\hat{e})^2}{80} + \frac{\sum_{t=3981}^{4000}(e-\hat{e})^2}{80} \tag{4}$$

Where e is the real value of the signal, $\hat{e}$ is the predicted value and t is the time step.

Model Predictive Control

The selected artificial networks were tested in the modeling and prediction of nonlinear system of two interconnected ball tanks. Involving usual simplifications, mathematical model of the system can be defined:

$$\pi h_1(d_1 - h_1)\frac{dh_1}{dt} + q_1 = q_{1v} \tag{5}$$

$$\pi h_2(d_2 - h_2)\frac{dh_2}{dt} - q_1 + q_2 = q_{2v} \tag{6}$$

Where d_j are tanks diameters, h_j are levels of liquid in the tanks, q_j are output flow rates and q_{jv} are input flow rates (j denotes number of tank). Input flow rates depend on the liquid levels:

$$q_1 = k_1\sqrt{|h_1 - h_2|} \tag{7}$$

$$\text{if } h_1 - h_2 < 0 \text{ then } q_1 = -q_1 \tag{8}$$

$$q_2 = k_2\sqrt{h_2} \tag{9}$$

Where k_1 and k_2 are constants describing the pipes. Initial conditions of equations (5) and (6), that were obtained from steady state, are $h_1(0)=1{,}5$m, $h_2(0)=1{,}3$m, $q_{1v}(0)=0.38$m^3/min, $q_{2v}(0)=0.19$m^3/min. System is regarded as SISO – controllers task is to control liquid level in the first tank ($h_1=y$) by influencing input flow rate q_{1v} (u). Second input flow rate remains constant.

In order to test the adaptivity of the predictors, parameters d_1 and d_2 were rapidly changed during simulation at time t=100min. This change is physically unfeasible and can be done only in the simulation. These parameters were chosen with the intention of better demonstration of the adaptivity feature.

The controller uses classical MPC objective function (Camacho and Bordons 1995):

$$J = \sum_{i=N_1}^{N_2}[w(k+i) - \hat{y}(k+i)]^2 +$$
$$+ \rho \cdot \sum_{i=1}^{N_u}[u_t(k+i-1) - u_t(k+i-2)]^2 \tag{10}$$

where N_1, N_2 and N_u define horizons over which the tracking error and the control increments are evaluated (usually $N_2 \geq Nu$). The u_t variable is the tentative control signal, w is the desired response and $\hat{y}$ is the network model response. The parameter ρ determines the contribution that the sum of the squares of the control increments has on the performance index.

There is usually assumed that after a certain interval $N_u < N_2$ there is no variation in the proposed control signals, that is:

$$\Delta u(k+i) = 0 \text{ for } i \in \langle N_u, N_2 - 1\rangle \tag{11}$$

This is equivalent to giving infinite weights to the changes in the control from a certain instant. This approach is adopted in this paper.

In addition to visual comparisons of control courses, quadratic criterions were also used to evaluate individual control courses. Two quadratic criterions were used to describe behavior of individual controllers.

$$S_y = \sum_{j=0}^{2000}(w_j - y_j)^2 \tag{12}$$

$$S_u = \sum_{j=0}^{2000}(u_j - u_{j-1})^2 \tag{13}$$

The symbol w_j stands for the desired value of the controlled variable (the level of liquid) in the step j, y_j is the controlled variable in the step j, u_i is the control signal in the step j. The limits $j=0$ and $j=2000$ corresponds to the simulation time 0 min and 200 min (because of the sampling rate 0,1 min).

The S_y criterion is based on the control error and thus represents the tracking performance of the controller. The criterion S_u is based on changes of control signal and represents the controller demands on the actuators.

SIMULATIONS AND RESULTS

All simulations were performed using MATLAB/Simulink.

CATS Benchmark

By many experiments it was discovered that ideal (sufficient) structure for MFFNN is 5 neurons in input layer (zero layer), 10 neurons in hidden layer and 1 output neuron. Furthermore, it was found out that number of training epochs has key influence on the prediction quality. Thus, it was decided to explore this influence. There were created, trained and tested 100 networks for each of ten different maximum numbers of training epochs (MTE) from the interval 25-250. In the tables 1-3 are presented averages of criterions E_1, E_2, final global error FGE, real number of training epochs RTE, prediction time PT and train time TT.

As well as in case MFFNN, ADALINE has important training parameter – learning rate (LR). The influence of this parameter was observed in the interval 10^{-2}-10^{-11}. And, again, it was done 100 simulations for each value of learning rate. The average values of criterions are presented in the table 3. The structure of *adaline* contained 5 neurons in the input (zero) layer and 1 neuron in the output layer.

Table 1: CATS Benchmark of the *mffnntt*

MTE	E_1 (10^4)	E_2 (10^4)	FGE (10^{-4})	RTE	PE (s)	TT (s)
25	3,02	2,93	52,65	25,00	0,41	7,04
50	2,44	2,44	66,92	50,00	0,41	14,07
75	2,17	2,12	10,55	75,00	0,41	21,11
100	1,76	1,61	6,89	100,00	0,41	28,26
125	1,60	1,51	5,98	124,92	0,41	35,13
150	1,49	1,38	5,70	149,69	0,41	42,06
175	1,48	1,38	5,54	171,19	0,41	48,05
200	1,46	1,34	5,43	191,82	0,41	53,73
225	1,46	1,34	5,34	208,67	0,41	58,68
250	1,52	1,41	5,32	219,60	0,41	62,17

Table 2: CATS Benchmark of the *mffnntp*

MTE	E_1 (10^4)	E_2 (10^4)	FGE (10^{-4})	RTE	PE (s)	TT (s)
25	55,61	55,27	34,56	25,00	0,40	7,28
50	9,98	10,62	13,66	50,00	0,40	14,10
75	5,80	6,05	8,67	75,00	0,40	21,11
100	2,37	2,42	7,08	100,00	0,40	28,02
125	3,51	3,48	6,36	124,75	0,40	34,88
150	4,06	3,93	5,76	149,13	0,40	41,72
175	1,69	1,61	5,40	173,58	0,40	48,51
200	2,21	2,17	5,28	195,21	0,40	54,52
225	2,22	2,16	5,14	213,43	0,40	59,11
250	2,02	2,02	5,12	225,35	0,40	62,24

As can be seen from tables 1-3, the best results from the point of view of criterions E_1 and E_2 were obtained by *mffnntt*. The *mffnntp* has slightly worse results, but not very significantly. The highest values of the criterions were obtained by *adaline*. However, the computational times (training time TT and prediction time PT) were shortest in case of *adaline*.

As can be seen from table 1, the best CATS predictor from the tested group is *mffnntt* trained approximately 200 epochs. From the table 3 it can be concluded that it is useless to set lower learning rate (LR) than $1 \cdot 10^{-4}$.

Table 3: CATS Benchmark of the *adaline*

LR	E_1	E_2	FGE	RTE	PE (s)	TT (s)
10^{-2}	$7,59 \cdot 10^{42}$	$8,97 \cdot 10^{42}$	0,17	1	0,55	0,13
10^{-3}	$4,99 \cdot 10^{13}$	$5,75 \cdot 10^{13}$	0,17	1	0,62	0,16
10^{-4}	$2,50 \cdot 10^4$	$2,66 \cdot 10^4$	0,17	1	0,51	0,09
10^{-5}	$2,50 \cdot 10^4$	$2,46 \cdot 10^4$	0,17	1	0,55	0,14
10^{-6}	$2,51 \cdot 10^4$	$2,45 \cdot 10^4$	0,17	1	0,69	0,21
10^{-7}	$2,51 \cdot 10^4$	$2,45 \cdot 10^4$	0,17	1	0,38	0,14
10^{-8}	$2,51 \cdot 10^4$	$2,45 \cdot 10^4$	0,17	1	0,39	0,16
10^{-9}	$2,51 \cdot 10^4$	$2,45 \cdot 10^4$	0,17	1	0,50	0,14
10^{-10}	$2,51 \cdot 10^4$	$2,45 \cdot 10^4$	0,17	1	0,46	0,11
10^{-11}	$2,51 \cdot 10^4$	$2,45 \cdot 10^4$	0,17	1	0,51	0,13

Model Predictive Control

Controller parameters were set to: ρ=0,8; N_1=1; N_2=20 and N_u=5. All predictors (ANNs) used 5 recent values of control signal and 5 recent values of output signal for prediction. Thus, the ANNs had ten inputs and one output.

The structure of *mffnntt* and *mffnntp* consisted 10 neurons in input layer (zero layer), 25 neurons in hidden layer and 1 output neuron. The structure of *adaline* contained 5 neurons in the input (zero) layer and 1 neuron in the output layer.

Due to satisfactory control performance during beginning of experiment all predictors were trained off-line from off-line prepared data. However, the ADALINE predictor was adapted during its function at each time step. Simulation results are shown in the figures 3-5 and table 4.Controller uses sampling rate 0,1 min and Levenberg-Marquart method was used as the optimization algorithm of the (10).
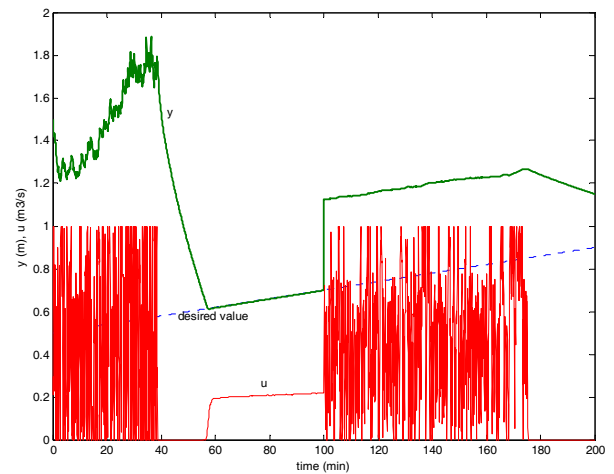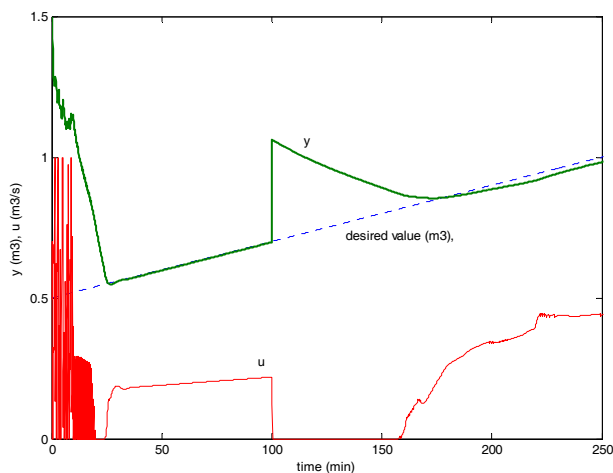

Figure 3: Control Using *mffnntt*

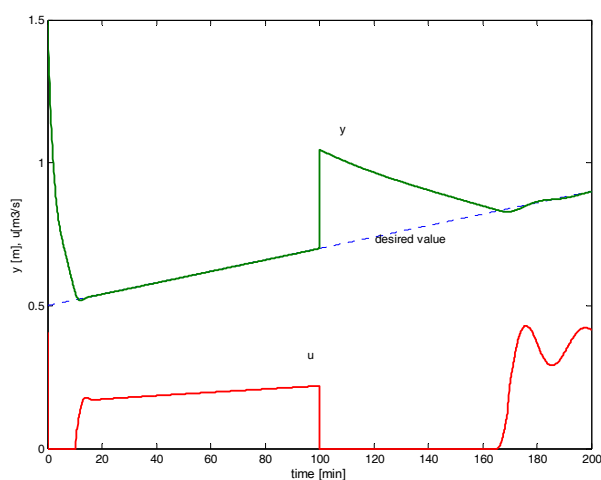Figure 4: Control Using *mffnntp*

Figure 5: Control Using *adaline*

Table 4: Comparison of the Predictors in MPC

Predictor of the controller	Criterion S_y	Criterion S_u
mffnntt	577,39	140,51
mffnntp	92,84	15,22
adaline	40,23	0,22

As can be seen from figures 3-5 and the table 4, the best results were gained using *adaline* predictor. The predictors based on MFFNN lead to oscillations of control signal. This behavior could be reduced by changing controller parameters (especially increasing ρ parameter). Furthermore, from the courses after change of system parameters (time=100min) it can be concluded that MFFNN based predictors do not provide natural adaptive prediction. However, ADALINE from its structure offers adaptivity, as was depicted in the figure 5. What is more, ADALINE has very simple structure thus the computational demands of the controller are notably low.

CONCLUSION

From the simulations it can be stated that selection of suitable predictor based on artificial network must be done very carefully. For various predicted systems has to be used different predictors. The multilayer feed-forward neural networks provided best results in the CATS benchmark of predictors. Nevertheless, in the model predictive control with the same types of artificial neural networks MFFNN did not offered applicable results. On the other hand ADALINE in the control task showed impressive results, including control quality, fast training and prediction times and adaptivity.

ACKNOWLEDGEMENT

REFERENCES

Arbib, M. A. 2002.*The Handbook of Brain Theory and Neural Networks*. The MIT Press, Cambridge, MA.

Camacho, E.F. and C. Bordons. 1995. *Model Predictive Control in the Process Industry*. Springer – Verlag, Berlin.

Chu J.Z.; Tsai, P.F.; Tsai, W.Y.; Jang, S.S.; Wong, D.S.H.; Shieh, S.S.; Lin, P.H.; and S.J. Jiang. 2003. "An Experimental Study of Model Predictive Control Based on Artificial Neural Networks". In *Proceedings of 7th International Conference on Knowledge-Based Intelligent Information and Engineering Systems* (Oxford, UK, Sep 3-5), Springer, Oxford, UK, 1296-1302.

Demuth, H. and M. Beale. 2002. *Neural Network Toolbox for Use with MATLAB*. The Mathworks, Inc., Natick, MA.

Freeman, J.A. and D.M. Skapura. 1991. *Neural Networks Algorithms, Applications, and Programming Techniques*. Addison Wesley Publishing Company, Reading, MA.

Hagan, M.; Demuth H.; and O.D. Jesus. 2002. "An Introduction to the Use of Neural Networks in Control Systems." *International Journal of Robust and Nonlinear Control*, No.12 (Nov), 959-985.

Hussain, M.A. 1999. "Review of the application of neural networks in chemical process control – simulation and online implementation." *Artificial Intelligence in Engineering*, No.1 (Jan), 55-68.

Kanjilal, P.P. 1995. *Adaptive prediction and predictive control*. P. Perengrinus Ltd., London.

Lendasse, A.; Oja, E.; and O. Simula. 2004. "Time Series Prediction Competition: The CATS Benchmark". In *International Joint Conference on Neural Networks* (Budapest, Jul 25-29), IEEE, 1615-1620.

Leshno, M.; Lin, V.Y.; Pinkus, A.; and S. Schocken. 1993. "Multilayer feedforward networks with a non-polynomial activation can approximate any function." *Neural networks*, No. 6, 861-867.

Liu, G.P. 2001. *Nonlinear identification and control – A neural network approach*. Springer - Verlag London Ltd., London.

Meszaros, A.; Rusnak, A.; and M. Fikar. 1999. "Adaptive Neural PID Control – Case Study: Tubular Chemical Reactor," *Computers and Chemical Engineering Supplement*, No.1999, 847-850.

Rumelhart, D. E.; Hinton, G. E.; and R. J. Williams. 1986. "Learning internal representations by error propagation". In *Parallel distributed processing: explorations in the microstructure of cognition, vol. 1: foundations*, D. E. Rumelhart and J. L. McClelland (Eds.). MIT Press, Cambridge, MA, 318-362.

Vandewalle, J. Suykens, J. De Moor, B. and A. Lendasse. 2007. "State-of-the-art and Evolution in Public Data Sets and Competitions for System Identification, Time Series Prediction and Pattern Recognition". In *32nd International Conference on Acoustics, Speech, and Signal Processing* (Honolulu, HI, Apr 15-20). Hawaii Convention Center in Honolulu, HI, 1269-1272.

Verdurmen R. E. M. and P. de Jong. 2003. "Optimising product quality and process control for powdered dairy products". In *Dairy processing: Improving quality*. G. Smith (Ed.). Woodhead Publishing Limited, Cambridge, 333-365.

Widrow B. and M. E. Hoff, Jr. 1960. "Adaptive switching circuits". In *IRE WESCON Convention Rec.* (New York). IRE, New York, 96–104.

AUTHOR BIOGRAPHIES

DAVID SAMEK was born in Zlin, Czech Republic and went to the Brno University of Technology, where he studied automation and control and obtained his degree in 2000. Then he studied post gradual in Tomas Bata University in Zlin and obtained Ph.D. degree in 2005. Since 2001 he has been working at Tomas Bata University in Zlin as a senior lecturer. His field of interest is focused on artificial networks and their applications. His e-mail address is : samek@ft.utb.cz and his Web-page can be found at http://www.ft.utb.cz/czech/uvi/.

PETR DOSTAL studied at Institute of Chemical Technology in Pardubice, where he studied automation of chemical processes and obtained his masters degree in 1968 and Ph.D. degree in 1976. He has been working Nowadays he works as a professor and head of Department of Process Control in Tomas Bata University, Faculty of Applied Informatics. His research is focused on field of interest is focused on artificial networks and their applications. His e-mail address is : dostalp@fai.utb.cz and his Web-page can be found at http://www.fai.utb.cz/.

Modelling and Predictive Control of Inverted Pendulum

Petr Chalupa
Vladimír Bobál
Department of Process Control
Tomas Bata University in Zlin
Mostni 5139, 760 01, Zlin, Czech Republic
E-mail: chalupa@fai.utb.cz

KEYWORDS

Modelling, State space model, Inverted Pendulum, Predictive control.

ABSTRACT

The paper is focused on creating a model of Inverted pendulum system and subsequent usage of this model to design a predictive controller of inverted pendulum system. The model is obtained on base of mathematical physical analysis of the system. Unknown parameters of the model are obtained from real-time experiments on the PS600 Inverted pendulum system. The model is designed in MATLAB/Simulink environment. The model was created with respect to most nonlinearities contained in the system. Nonlinearities are caused by fundamental principles of the system and by friction between individual parts of the system. Thus, the model is highly non-linear and therefore linearization around working point was performed and continuous linearized model was calculated as well as its discrete version. The discrete linear model was used to design predictive controller which was also verified by real time experiments.

INTRODUCTION

Most of current control algorithms are based on a model of a controlled plant (Bobál et al. 2005). It is obvious that some information about controlled plant is required to be able to control output of the plant or to investigate its properties and behaviour. Two basic approaches of obtaining plant model exist: the black box approach and the mathematical-physical analysis of the plant.

The black box approach to the modelling (Liu, 2001) is based on analysis of input and output signals of the plant. The main advantages of this approach lies in the possibility of usage the same identification algorithm for wide set of different controlled plants. Also, the knowledge of physical principle of controlled plant and solution of possibly complicated set of mathematical equation is not required. On the other hand, model obtained by black box approach is generally valid only for signals it was calculated from. For example, if step response was used to obtain the model, this model should not be used for analysis of the system behaviour when high frequency changes of input signals are

applies. The mathematical-physical analysis of the plant and following derivation of the relations between plant inputs and outputs provides general model which can be valid for whole range of plant inputs and states. Contrary, there is usually a lot of unknown constants and relations in the model description when performing mathematic-physical analysis.

The second method, mathematical-physical analysis, is used in this paper. The goal of the work was to obtain a mathematical model of the PS600 Inverted Pendulum System (Amira, 2000), to design the model in MATLAB-Simulink environment and use this model for a design of model predictive controller. The PS600 laboratory equipment was developed by Amira Gmbh, Duisburg, Germany and serves as a real-time model of unstable, highly nonlinear system.

When the model of the controlled system is known the problem of selecting an appropriate control synthesis arises. Many successful control techniques have been developed in past decades. One of them is model predictive control (MPC) (Camacho and Bordons, 2004). Contrary to most other approaches, MPC uses not only current and previous values of control circuit signals but also future values of reference signal. Future course of reference signal is known in many applications and thus can be used in controller synthesis.

The scheme of a simple control circuit with self-tuning predictive controller is shown in Figure. 1. Note that the reference signal is marked as w(t), which means that the course of reference signal is sent to the controller, not only the current value w(k).
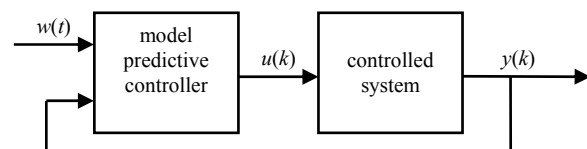

Figure 1: Control circuit with model predictive controller
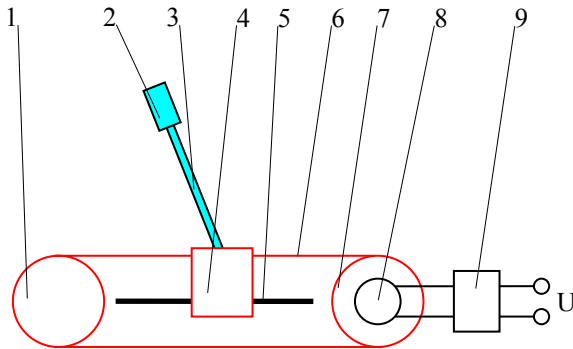
INVERTED PENDULUM SYSTEM

The PS600 inverted pendulum system is shown in figure 2. The main parts of the system are cart driven by

servo amplifier and the pendulum rod attached to the cart.

Figure 2: Photo of PS600 Inverted Pendulum system

A simplified scheme of the inverted pendulum system is shown in figure 3. This scheme was used for mathematical physical analysis of the system.

1 – guide roll   6 – transmission belt
2 – pendulum weight   7 – drive wheel
3 – pendulum rod   8 – DC motor
4 – cart   9 – servo amplifier
5 – guiding bar

Figure 3: Scheme of inverted pendulum

The PS600 Inverted Pendulum is designed as a system with one input and two measured outputs – SITO (single input two outputs). The input of the system is a control voltage of the servo amplifier (U) and the outputs are cart position and angle of pendulum rod. Both outputs are measured by incremental encoders (Amira, 2000).

Systems allows different control objectives with various difficultness of control design. The most common cases are:

- control of cart position with pendulum acting as a disturbance
- control of cart position with stabilization of the pendulum in the stable equilibrium position (ie. pendulum underneath the guiding bar)
- stabilization of the pendulum in the upright position

- control of the cart position with pendulum in upright position.

MATHEMATICAL MODEL

Forces and moments acting in the system were analysed using figure 4 where φ represents the angle of pendulum rod, M_0 and M_1 stands for the weight of the cart and pendulum respectively, l_S is a distance between centre of gravity of the pendulum and the centre of rotation of the pendulum and g is the gravity acceleration constant. Symbol F represents the force produced by the DC motor.
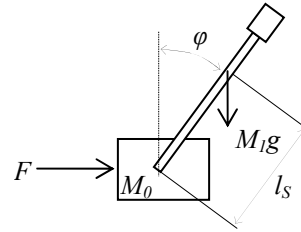

Figure 4: Analysis of inverted pendulum

It is obvious that the position and dynamics of the pendulum affects the cart. This affect is described by a force which can be divided into horizontal and vertical components. The horizontal component of the force is:

$$H = M_1 \frac{\partial^2}{\partial t^2}\left(r + l_S \sin \varphi\right) \tag{1}$$

where r represents position of the cart. The vertical component is:

$$V = M_1 \frac{\partial^2}{\partial t^2}\left(l_S \cos \varphi\right) \tag{2}$$

The motion equation of the cart can be written as follows:

$$M_0 \frac{\partial^2 r}{\partial t^2} = F - H - F_r \frac{\partial r}{\partial t} \tag{3}$$

where F_r represents constant of a velocity proportional friction of the cart. According to the angular momentum conservation law, the rotary motion of the rod about its centre is described as:

$$\Theta_S \frac{\partial^2 \varphi}{\partial t^2} = V l_S \sin \varphi - H l_S \cos \varphi - C \frac{\partial \varphi}{\partial t} \tag{4}$$

where Θ_S represents the inertia moment of the pendulum rod with respect to the centre of gravity and C denotes the friction constant of the pendulum.

Substituting equations (1) and (2) into equations (3) and (4) leads to the description of behaviour of the system by set of two nonlinear differential equations:

$$Mr'' + F_r r' + M_1 l_S \varphi'' \cos\varphi - M_1 l_S \left(\varphi'\right)^2 \sin\varphi = F \quad (5)$$

$$\Theta\varphi'' + C\varphi' - M_1 l_S g \sin\varphi + M_1 l_S r'' \cos\varphi = 0 \quad (6)$$

where following abbreviations were used:

$$\Theta = \Theta_S + M_1 l_S^2$$
$$M = M_0 + M_1$$

The input of the system is represented by a force F. This force is proportional to the control voltage of servo amplifier.

The model described by equations (5) and (6) was created in the MATLAB/Simulink environment as a standalone block. This approach can be used in simulations or as a part of rapid prototyping of a controller. The Simulink scheme is shown in figure 5.
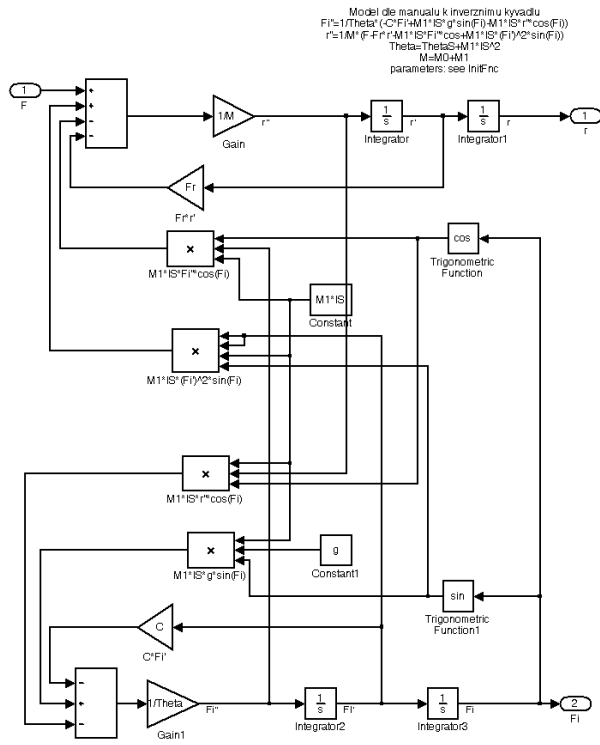
Figure 5: Simulink scheme of the nonlinear model

State description of the model

Equations (5) and (6) describe behaviour of the inverted pendulum. To obtain state space description of the system, following state vector was introduced:

$$\boldsymbol{x} = \begin{bmatrix} r \\ r' \\ \varphi \\ \varphi' \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \quad (7)$$

Then, the dynamics of the system can be described by a vector function

$$\boldsymbol{x}' = \begin{bmatrix} x_1' \\ x_2' \\ x_3' \\ x_4' \end{bmatrix} = \boldsymbol{f}(\boldsymbol{x}, F) = \begin{bmatrix} x_2 \\ f_2(\boldsymbol{x}, F) \\ x_4 \\ f_4(\boldsymbol{x}, F) \end{bmatrix} \quad (8)$$

Functions f_2 and f_4 can be derived from equations (5) and (6):

$$x_2' = r'' = \frac{1}{M_1^2 l_S^2 \cos^2\varphi - M\Theta}\Big[-CM_1 l_S\varphi'\cos\varphi +$$
$$+M_1^2 l_S^2 g\sin\varphi\cos\varphi + \Theta F_r r' - \Theta M_1 l_S\left(\varphi'\right)^2\sin\varphi - \Theta F\Big] \quad (9)$$

$$x_4' = \varphi'' = \frac{1}{M_1^2 l_S^2 \cos^2\varphi - M\Theta}\Big[MC\varphi' - MM_1 l_S g\sin\varphi -$$
$$-M_1 l_S F_r r'\cos\varphi + M_1^2 l_S^2\left(\varphi'\right)^2\sin\varphi\cos\varphi + M_1 l_S F\cos\varphi\Big] \quad (10)$$

Outputs of the system are identical to its states r and φ and thus their computation is straightforward.

$$\boldsymbol{y} = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} r \\ \varphi \end{bmatrix} = \begin{bmatrix} x_1 \\ x_3 \end{bmatrix} \quad (11)$$

Model linearization

Most control design techniques are based on linear model of the controlled system. If the mathematical model of the system is nonlinear, it has to be linearized around some appropriate working point (Antsaklis and Michel, 2005). In case of availability of mathematical model of the system, linearization is performed by calculating first two members of corresponding Tylor series (function value and first derivation) .

The state space linear was calculated for working point of unstable steady state (i.e. upright position of the pendulum)

$$\boldsymbol{x}_0 = \begin{bmatrix} r_0 \\ r_0' \\ \varphi_0 \\ \varphi_0' \end{bmatrix} = \begin{bmatrix} x_{10} \\ x_{20} \\ x_{30} \\ x_{40} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (12)$$

Linearized continuous model of the inverted pendulum is calculated according following equation:

$$X' = f(x,0) + \frac{\partial f}{\partial x}\bigg|_{X=0,F=0} \cdot X + \frac{\partial f}{\partial F}\bigg|_{X=0,F=0} \cdot F = AX + BF \quad (13)$$

Computation of individual elements of matrixes *A* and *B* is straightforward using equations (8)-(10).

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & a_{22} & a_{23} & a_{24} \\ 0 & 0 & 0 & 1 \\ 0 & a_{42} & a_{43} & a_{44} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ b_2 \\ 0 \\ b_4 \end{bmatrix} \quad (14)$$

where

$$a_{22} = \frac{\Theta F_r}{R}, \qquad a_{23} = \frac{M_1^2 l_S^2 g}{R}, \qquad a_{24} = \frac{-C M_1 l_S}{R}$$

$$a_{42} = \frac{-M_1 l_S F_r}{R}, \quad a_{43} = \frac{-M M_1 l_S g}{R}, \quad a_{44} = \frac{C M}{R}$$

$$b_2 = \frac{-\Theta}{R}, \qquad b_4 = \frac{M_1 l_S}{R}, \qquad R = M_1^2 l_S^2 - M\Theta$$

The continuous linear model can be used either directly for controller design or can be transformed to its discrete equivalent with respect to sample time used in discrete controller. The second approach was used in design of MPC controller.

IDENTIFICATION OF MODEL PARAMETERS

Initial estimates of model parameters were obtained from the manufacturer of the system (Amira, 2000). This setting of parameters is summarized in table 1.

Table 1: Initial setting of model parameters

parameter	value	unit
M_0	4.0	kg
M_1	0.36	kg
l_S	0.451	m
Θ	0.08433	kg m^2
F_r	10.0	kg s^{-1}
C	0.00145	kg m^2 s^{-1}
k_A	2.25	N/V

Constant k_A represents the gain of the servo amplifier (see figure 3).

Validation of this setting was performed especially by comparison of impulse responses of the real-time system and corresponding impulse responses of the Simulink model. Impulses of duration 0.3 s and various gain were applied to the PS600 system and courses of outputs were recorded and compared to responses obtained by Simulink model.

At first attention was paid to the cart position. The comparison of responses obtained by applying two different impulses is shown in figure 6. It can be seen that the Simulink model does not represent the real-time

plant well. In case of small gain of impulse applied to the Simulink model, the final position of the cart is smaller, while greater impulses lead to smaller position comparing to the positions obtained by real PS600 system.
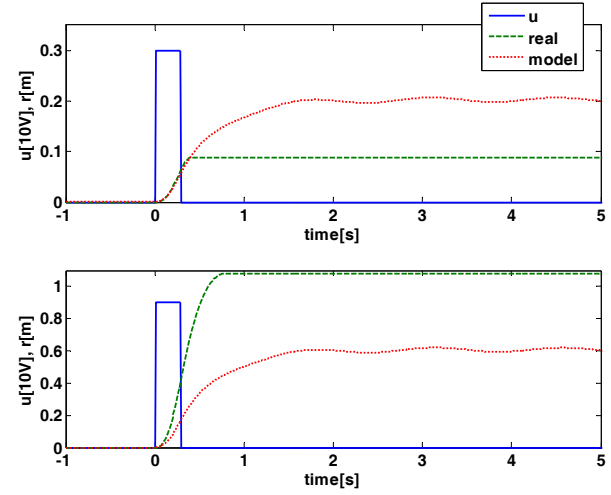

Figure 6: Comparison of impulse responses of PS600 system and its Simulink model (cart position)

By further investigation and comparison of responses, it was found out that given friction constant F_r and servo amplifier gain k_A do not correspond to PS600 and should be changed.

But even changes of other model parameters cannot cover the courses after the end of impulse. The slowing down of the cart is quantitatively different for model and PS600 system. The comparison of cart speed is shown in figure 7.
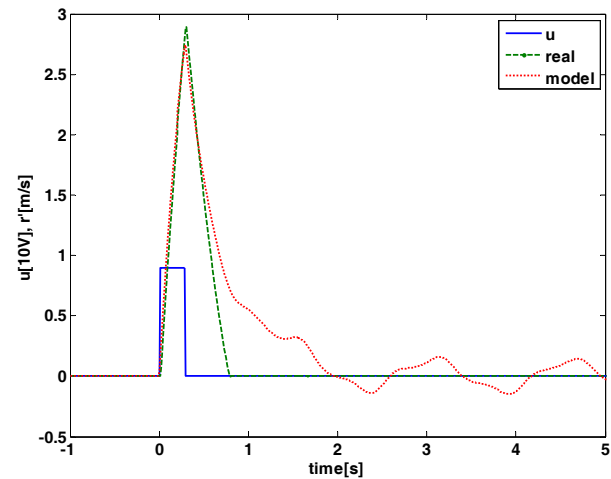

Figure 7: Comparison of impulse responses of PS600 system and its Simulink model (cart speed)

It can be seen that cart of real PS600 slows down and stops after the end of the impulse while oscillations occur in the Simulink model. These oscillations are

caused by swinging of the pendulum and cannot be suppressed by any parameter of the Simulink model.

Non-presence of oscillations in real PS600 is caused by presence of dry (Coulomb) friction between cart and guiding bar. A force invoked by this friction acts against the movement of the cart and can be described by the following equation:

$$F_{rC} = -F_C \cdot \text{sign}(r') \qquad (15)$$

where F_C is a Coulomb friction constant. The Simulink model had to be updated to cover this friction force. Model update is quite simple because the Coulomb friction acts in a similar manner as force proportional friction. The force proportional term in equations (3), (5), (9) and (10) was superseded as follows:

$$F_r r' \rightarrow F_r r' + F_c \, \text{sign}(r') \qquad (16)$$

Investigation of the pendulum rod angle φ was performed in the similar manner as investigation of the cart and led also to update of model parameters. It was found out that the value of the pendulum friction constant C should be updated. Moreover, the distance between centre of gravity of the pendulum and the centre of rotation (l_S) is slightly smaller then the value given by manufacturer.

Updated model parameters of PS600 are summarized in table 2. Changed values are marked by shaded background.

Table 2: Initial setting of model parameters

Parameter	Original value	New value	Unit
M_0	4.0	4.0	kg
M_1	0.36	0.36	kg
l_S	0.451	0.420	m
Θ	0.08433	0.08433	kg m^2
F_r	10.0	6.5	kg s^{-1}
C	0.00145	0.00652	kg m^2 s^{-1}
k_A	2.25	7.50	N/V
F_C		15	N

Due to algebraic loop in the Simulink model and further nonlinearity brought by Coulomb friction, the modelling of the system by terms of Simulink blocks is not suitable. Moreover discontinuity in the Coulomb friction around $r' = 0$ leads to algebraic problems during numeric integration. The PS600 model was created in the form of Simulink s-function to cope with these problems.

Comparison of impulse responses of the PS600 and its improved model is shown in figure 8. It can be seen that courses of signals obtained by improved model are far

closer to real PS600 comparing to initial model (see figure 6). Note that pendulum angle of around π corresponds to stable position of pendulum (i.e. pendulum rod pointing down).
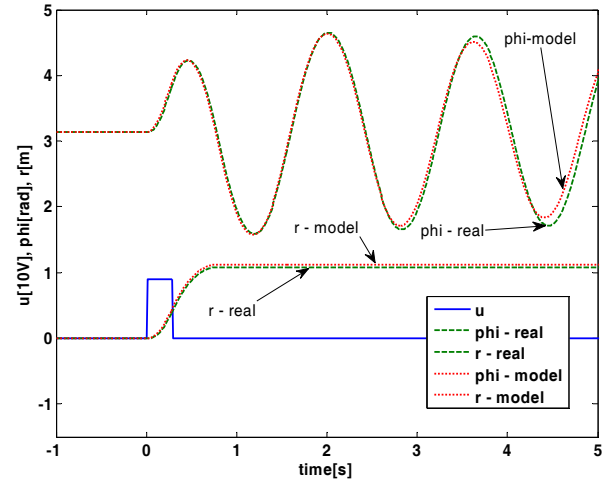

Figure 8: Comparison of impulse responses of PS600 system and its improved Simulink model (cart position and pendulum angle)

MODEL PREDICTIVE CONTROL OF THE PS600

The design of mode predictive controller for PS600 system is based on the mathematical model of the system and its linearized version. Control objectives are given as follows:
1. hold the pendulum rod in the upright position ($\varphi=0$, i.e. $w_2(t)=0$)
2. move the cart according to predefined reference trajectory ($r=w_1(t) = w(t)$)

The PS600 can be considered as single input two outputs system (SITO). The second output of the system (φ) acts as a constrained because it should be close to zero during whole control process.

Generally, the computation of control signal of model predictive controller (MPC) is based on minimization of particular criterion (Kwon and Han, 2005). Usually a quadratic criterion is used (Sunan *et al.*, 2002).:

$$J(k) = e^T Q(k) e + u^T R(k) u \qquad (17)$$

where e is a vector of predicted control errors, u is a vector of future control signal samples and square matrixes Q and R allows to set weighting of individual vector elements. In most cases control sample differences Δu are taken into account instead of u, but due to integrative behaviour of the cart system the simple criterion (17) is suitable for PS600. Control sequence is obtained by minimizing criterion (17). The receding horizon is usually used: only a finite number of future values is used in the criterion and only the first element of the obtained control sequence is applied to the controlled system.

Future outputs of the controlled system, and consequently control errors, are computed on base of the linear model. If the system is described by state space model, future system outputs depend only on current states of the system and future control signals:

$$y = Px(k) + Hu \qquad (18)$$

where P and H are constant matrixes. For SITO system, the vectors y and u have following form:

$$y = \begin{bmatrix} y_1(k+1) \\ y_2(k+1) \\ y_1(k+2) \\ y_2(k+2) \\ \vdots \\ y_1(k+N) \\ y_2(k+N) \end{bmatrix}, \quad u = \begin{bmatrix} u(k) \\ u(k+1) \\ \vdots \\ u(k+N) \end{bmatrix} \qquad (19)$$

Proposed predictive controller simplifies criterion (17) by assigning the same weight to all samples of the same signal – matrixes Q and R are diagonal. Then the predictive control criterion can be rewritten into the following form:

$$J(k) = \lambda_1 \sum_{j=1}^{N} \left[e_1(k+j) \right]^2 + \lambda_2 \sum_{j=1}^{N} \left[e_2(k+j) \right]^2 + \lambda_3 \sum_{j=1}^{N} \left[u(k+j-1) \right]^2 \qquad (20)$$

where

$$e_1(i) = w_1(k+j) - y_1(k+j) = w_r(k+j) - r(k+j)$$
$$e_2(i) = w_2(k+j) - y_2(k+j) = -\varphi(i)$$
$$u(i) = F(i)$$

where N is prediction (control) horizon and λ_1, λ_2 and λ_3 are weights. Process of minimizing of the criterion can be rewritten to a quadratic programming problem:

$$J(k) = u^T M(k) u + g(k) u \qquad (21)$$

where M and g are matrix and vector derived from weights and model parameters. Quadratic programming problem is usually solved numerically. This allows further constraints to be applied to vector u.

Simulation and real time experiments

This section contains several results of control of the PS600 system. A Simulink control circuit used for both simulations and real time measurements is shown in figure 9. The circuit contains block for compensation of Coulomb friction. The fiction F_C=15N corresponds to

input signal of 2V (see table 2). Therefore control signal is always outside the interval of (-2V, 2V) except special case of u=0V which occurs at the beginning of control courses.

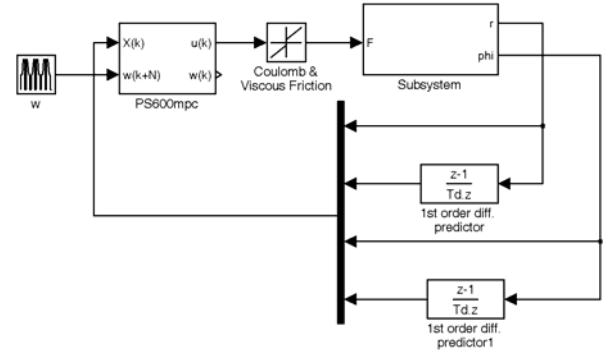Figure 9: Simulink control circuit used for simulation and real time experiments

Precise knowledge of current states of the control system ($X(k)$) is crucial for the MPC controller. Especially is case of unstable controlled system as PS600. Different approaches of reconstruction of immeasurable states exist. A very simple state reconstruction method is used in the scheme in figure 9. Two states are measured directly as system outputs (r,φ) and their derivatives are to be reconstructed. A two point difference is used to compute derivatives as shown in figure 9.

First control courses were obtained by simulation and are presented in figure 10. Prediction horizon of N=25 samples and sample time of T_0=0.04s were used. Weights were set to λ_1=10, λ_2=1 and λ_3=0.001.
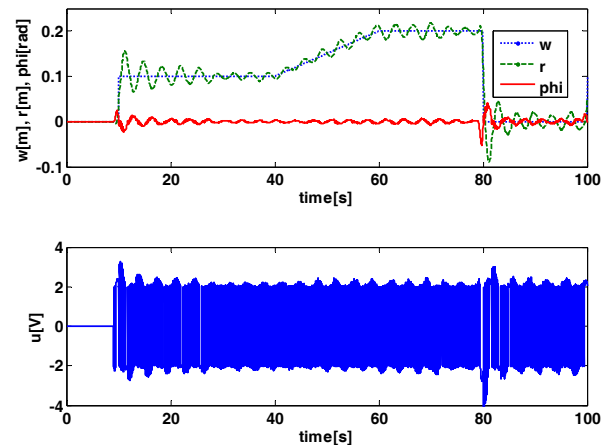

Figure 10: Simulation results of predictive control of PS600

Control of real plant with the same settings is depicted in figure 11. It can be seen that reference tracking is slightly worse than in case of figure 10.
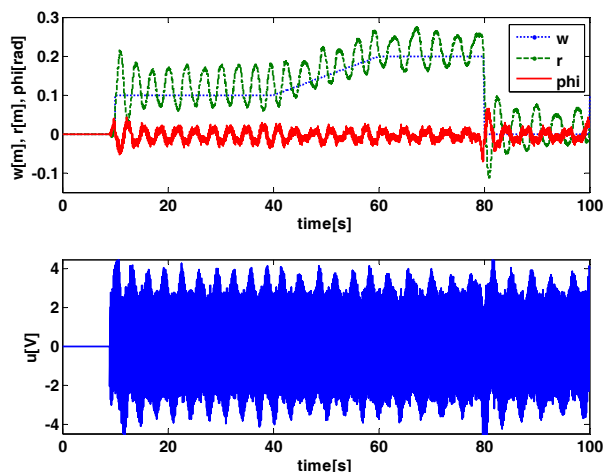
Figure 11: Real time control of PS600

An affect of different setting of weight of cart position error is shown in figure 12. In this case, all settings remained the same as in previous cases except $\lambda_1=50$. It is obvious that higher value of λ_1 leads to better reference tracking of cart position but it can be observed that pendulum angle is also smaller than in the previous case.
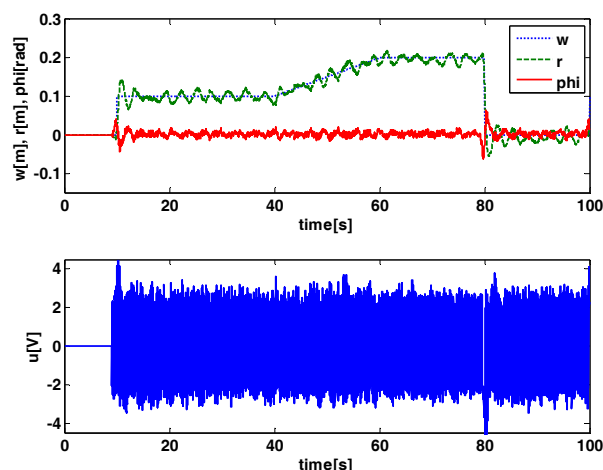

Figure 12: Real time control of PS600 ($\lambda_1=50$)

CONCLUSION

A mathematical mode of PS600 inverted pendulum was derived and its linearized version was calculated. Model parameters were investigated and their values were refined to obtain better correspondence between model and real equipment. Coulomb (dry) friction was also included into the model.

A model predictive controller was designed on base of a linearized discrete model. The controller was verified in both simulations and real time experiments.

Future work is to be focused especially to improvement reconstruction of immeasurable states.

REFERENCES

Amira. 2000. *PS600 Laboratory Experiment Inverted Pendulum.* Amira GmbH, Duisburg.

Antsaklis, P. J. and A. N. Michel. 2005. *Linear Systems.* Birkhauser, Boston.

Bobál, V.; J. Böhm; J. Fessl and J. Macháček. 2005. *Digital Self-tuning Controllers: Algorithms, Implementation and Applications.* Springer - Verlag London Ltd., London.

Camacho, E. F. and C. Bordons. 2004. *Model Predictive Control.* Springer-Verlag, London.

Kwon, W. H. and S. Han (2005). *Receding Horizon Control.* London: Springer-Verlag.

Liu, G. P. 2001. *Nonlinear identification and control – A Neural Network Approach.* Springer - Verlag London Ltd., London.

Sunan, H., T. K. Kiong and L. T. Heng (2002). *Applied Predictive Control.* London: Springer-Verlag.

AUTHOR BIOGRAPHIES

PETR CHALUPA was born in Zlin, Czech Republic in 1976. He graduated from Brno University of Technology in 1999. He obtained his Ph.D. in Technical Cybernetics at Tomas Bata University in Zlin in 2003. Nowadays he works as a researcher at Centre of Applied Cybernetics at Tomas Bata University in Zlin. His research interests are adaptive and predictive control of real-time systems. You can contact him on email address chalupa@fai.utb.cz

VLADIMÍR BOBÁL was born in Slavičín, Czech Republic in 1942. He graduated in 1966 from the Brno University of Technology. He received his Ph.D. degree in Technical Cybernetics at Institute of Technical Cybernetics, Slovak Academy of Sciences, Bratislava, Slovak Republic. He is now Professor in the Department of Process Control, Faculty of Applied Informatics of the Tomas Bata University in Zlín. His research interests are adaptive control systems, system identification and CAD for self-tuning controllers. You can contact him on email address bobal@fai.utb.cz

VR / AR
in
Modelling and Simulation

A DEFORMABLE SURFACE MODEL FOR SOFT VOLUME SIMULATION

Sylvester Arnab
Vinesh Raja
Warwick Manufacturing Group
University of Warwick
CV4 7AL, Coventry, UK
E-mail: s.arnab@warwick.ac.uk

KEYWORDS

Surface model, Physically Based Modelling, Mass Spring System, Deformable Model

ABSTRACT

This paper introduces a surface alternative to the volume model commonly employed in soft object simulation. This is inspired by a significant interest in employing the surface data due to its simplicity. However, there are issues in properties estimation and volume preservation. Therefore, the aim of the ongoing research includes exploring the potential of a surface mass spring model with shape-preserving springs for volume simulation. The approach illustrates the possibility of deriving the spring stiffness at runtime based on the real material properties as well as achieving a global shape deformation despite the non-existence of internal volume. The proposed scheme can be further explored to address other material properties.

INTRODUCTION

There has been an increase in the use of soft volume simulation in medical training, where accurate visual and behaviour replications of human organs are of great importance. These organs are geometrically modelled into 3 dimensional volumes based on medical data such as CT/MRI scans. This volume data can then be simulated based on the organ's physical and material properties. To allow tactile interaction with the visual representation, haptic rendering has been introduced. However, there is a significant interest in utilising surface data as an alternative to volume data due to its simplicity. A surface model has the benefit of having a less complex mesh network compared to the volume counterpart, which can be manipulated to emulate soft volume behaviour. The main issue is that volume simulation requires constant volume preservation as well as correct volume behaviour during simulation. Generally, a surface model would collapse under gravity and without the internal volume, determining the correct deformation effect would be a challenge. Figure 1 shows that the haptic interaction on the cylinder surface does not produce the correct global deformation effect on the opposite side. For instance, a blood vessel with

internal pressure from the blood density would produce a deformation effect at the opposite surface relative to the interaction point. The proposed approach explores these issues where, a surface mass spring system with anisotropic shape preserving springs and weighted constraints have been employed.
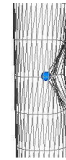

Figure 1: A Surface Model Deformed upon Interaction

RELATED WORKS

To create a volume model using surface data, the surface model can be re-meshed to create internal volume. However, this creates computational overhead during simulation imposed by a more complex volume network. The addition of new artificial springs (Bourguignon and Cani 2000) to the existing volume mesh produces an object that is stiffer than it should be (Hong et al. 2006). There are also attempts that employ surface data which addressed shape preservation but not volume (Nedel and Thalmann 1998; Aubel and Thalman 2000). (Hong et al. 2006) introduced weighted constraints that control the deformation distribution of the muscle instead of using additional springs. However, this method is fundamentally focusing on the local radius of influence of the interaction and is not influenced by the orientation of the interaction force.

A more effective shape preserving method has been embedded into the Mass Spring Systems (MSS) where the springs are placed at the mesh nodes. They are also known as the local shape memory (Marchal et al. 2005) and zero-length springs. These springs have been employed to simulate non-linear skin behaviour of a virtual thigh (Mendoza et al. 2002; Laugier et al. 2003). (Choi et al. 2005) administered these springs to preserve object shape during simulation when a force is imposed on the nodes. However, the stiffness of the springs was either statistically fine-tuned based on predefined properties (Zhang et al. 2002) or regularly distributed. (Arnab and Raja 2007) extended this

method by extracting the properties of the springs based on radial links (Vassilev & Spanglang 2002; Balaniuk and Salisbury 2003).

Volume behaviour is influenced by the properties estimated for the model. Regular properties distribution is very common where the regular mesh topology (regular node concentrations) is assumed (Gelder 1998; Delingette 1998; Bourguignon and Cani 2000; Brown et al. 2001) or in the case of irregular node concentrations, the topology is modified to be as regular as possible (Deussen et al. 1995; Bielser 2003). However, when a portion of the surface model is refined, the regular topology becomes irregular. Consequently, the properties require re-estimation within the refined area. (Choi et al. 2005) attempted properties re-estimation after surface refinement based on the topology but the behaviour patterns between the coarse and the refined area do not coincide. The behaviour is improved by our previous method discussed in (Arnab and Raja 2007) where these patterns achieve a higher level of co-incidence.

THE PROPOSED SCHEME

The scope of dynamic behaviour in this research is within a constrained space such as a human breast fixed on a static body (figure 2). The model is constructed from the surface mesh and the dynamic behaviour is achieved by employing the surface MSS with volumetric supports.
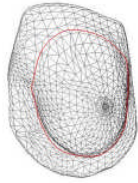

Figure 2: A Breast Model (within the drawn boundary) with Convex Shape

To address the issue of properties estimation as well as the non-existence of inner volume for the proposed surface model, the object's internal volume represented by the surface elements has to be considered. Initial discretisation of volume is required to estimate the properties based on the real material properties. The assumption is that the initial shape of object is convex with known centre of object. This scheme extended the approach described in (Arnab and Raja 2007).

Surface Mesh Topology

The surface mass spring model is based on the surface mesh topology where the springs are represented by the edges of the triangular elements (figure 3). For instance, the edge that connects the nodes with mass m_i

and m_j is the spring with stiffness k_{ij}. For a spring link ij as in figure 2, the internal force $\boldsymbol{F_{ij}}$ is

$$F_{ij} = K_{ij}\left(\left\|p_j - p_i\right\| - l_{ij}\right)\frac{p_j - p_i}{\left\|p_j - p_i\right\|} \tag{1}$$

, where i and j denote the indexes for the 2 nodes, $\|p_j - p_i\|$ is the magnitude of the displacement of the current state of the spring link ij, l_{ij} is the rest length of the spring link, and K_{ij} is the stiffness (spring) coefficient of the node pair. The system of equations is solved through the time using numerical integration techniques.
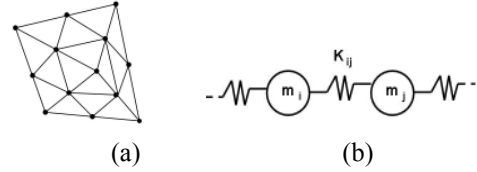

(a)   (b)

Figure 3: Surface Mesh Topology: a) the nodes or vertices connected by edges (b) a pair of nodes or masses connected by a spring

This spring network will not be able to preserve volume behaviour during deformation due to the non-existence of the inner volume mesh. Therefore, inner supports, which are based on the concept of shape preserving springs, are introduced to provide the volumetric reinforcement.

Volumetric Support

In order to preserve the rest shape of the object, the inner volume support (figure 4) has to be defined for the mass at the nodes. The volume spring works in a similar way as the surface spring but the rest length of the spring is zero. The springs preserve the object shape at equilibrium as they act as virtual anchors.
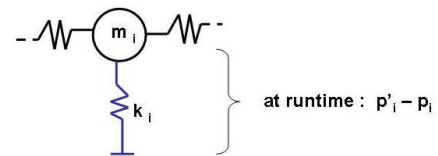

Figure 4: An inner spring stretched during simulation

Based on Equation (1) but rest length is equal to zero, the reaction force at inner spring will only affect mass at node $\boldsymbol{i}$,

$$F_i = K_i\left(\left\|p'_i - p_i\right\|\right)\frac{p'_i - p_i}{\left\|p'_i - p_i\right\|} \tag{2}$$

, where K_i is the stiffness of the inner spring at node i, p'_i is the new position of node i at runtime and p_i is the anchored position of node i (figure 4). In order to not only preserve the object shape but also maintain a constant object volume during simulation, the concept of these shape preserving springs has been extended.

Volumetric Discretisation

The relationship of the surface elements to the global object properties is defined and employed to extract the material properties for each of the surface elements of a mass spring mesh. Explicit inner volume approximation is carried out for the underlying surface mesh. Initially, the relationship of the surface nodes relative to the centre of object (Vassilev & Spanglang 2002; Balaniuk & Salisbury 2003; Arnab & Raja 2007) is considered (figure 5).
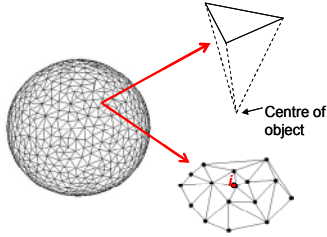

Figure 5: Volumetric Discretisation based on Radial Links

If the mesh is of irregular topology, the triangles are of different sizes. Therefore, different properties should be distributed to the individual triangles. The obvious issue here is the need to correctly distribute the properties from the neighbouring triangles to the member nodes. For instance, mass at node *i* is determined by manipulating the masses at the neighbouring triangles of which it is a member. (Bourguinon and Cani 2000) and (Bielser 2003) uniformly distributed mass to the respective nodes based on the barycentric relationship. In the case of a triangle, a third of the triangle mass is distributed to the nodes. Each node accumulates the mass contributions from the neighbouring volume represented by the triangle.

However, since the mass at the nodes are not known, the barycentric position of the triangle is determined by assuming uniform mass at the nodes. This is of course incorrect if the nodes are of different individual distances to the object centre. Hence, the proposed volume discretisation is based on the stress and strain relationship of a material instead. This relationship has been employed to extract the elasticity modulus of a real material and is based on a force parallel to the normal of the cross-sectional area. In MSS, the internal and external forces will directly influence the dynamic behaviour of the nodes and not the triangular elements. Therefore, as illustrated in figure 6, the new distance vector with length L'_i, relative to the object centre, the initial length L_i and the surface normal at the node, can be derived.
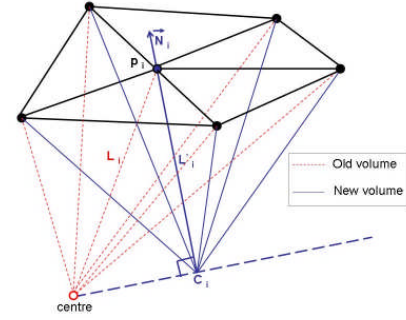

Figure 6: Volumetric Discretisation based on the Neighbouring Triangles Relative to the Normal at p_i

The new volume is based on node p_i relative to the new centre point c_i and the other nodes or vertices of the triangular element. Hence, the properties of a node are directly influenced by its normal and the volume relative to its normal. This discretisation supports the need for the volume springs at the nodes to be influenced by the orientation of force relative to the normal. Similar discretisation and volume calculation will be carried out for the other nodes relative to their respective normals and neighbouring triangles. The resulting discretisation represents the mass and the material properties within the volume.

Mass Estimation

To support both irregular topology and objects with non-uniform shape, the mass distribution is based on the volume discretisation in figure 5. The total mass of the virtual model has to replicate the total mass of the real object. Therefore, the estimation method has to take mass preservation into account. This has to be true even when the mesh topology is modified. Therefore, the mass at node *i* is estimated as

$$m_i = \frac{\sum_{i \in t} v_i}{\sum_{i=0}^{n} v_i} M$$

, where the estimation ratio for each node is the total volume under the neighbouring triangular elements *t* (of which node *i* is a member) divided by the total volume for all *n* nodes, and *M* is the object mass. This method can be used to re-estimate mass upon any topological modifications.

The Estimation of Spring Stiffness

The surface spring stiffness is treated like a membrane which behaviour mimics the surface of an object. Therefore, the spring stiffness is distributed based on an estimation algorithm which has been commonly employed to estimate the behaviour of a membrane with irregular mesh topology (Gelder 1998; Maciel et. al 2003; Lloyd et al. 2007; Arnab and Raja 2007)

The volume spring stiffness defines the level of elasticity of virtual space or volume represented by the

node, which is also based on the explicit volume discretisation. Therefore, the value of the stiffness is influenced by the normal at the node. The elasticity of the stiffness can be described based on the elasticity, shear and bulk modulus. These characteristics are considered when the estimation method is derived in order to address the issue of the single dimensionality of the existing spring and to extend its role from not only to preserve the object shape at equilibrium but also to dynamically maintain the object volume during simulation.

The behaviour of the spring stiffness at runtime depends on the orientation of the acting force along each spring. Based on this relationship and the fact that the properties at the node i have been extracted based on its normal, figure 7 illustrates the orientation of the force F_i relative to its normal N_i. The stiffness of the volume spring K_i at each node i is derived at runtime:

$$K_i = \left[\left\| \vec{N_i} \cdot \vec{F_i} \right\| \quad 1 - \left\| \vec{N_i} \cdot \vec{F_i} \right\| \right] \begin{bmatrix} K_E \\ K_G \end{bmatrix}_i$$

, where $\left\| \vec{N_i} \cdot \vec{F_i} \right\|$ is the dot product of the normal unit vector and the acting force unit vector at node i, K_E and K_G are the spring stiffness based on linear elasticity modulus and shear (rigidity) modulus respectively.
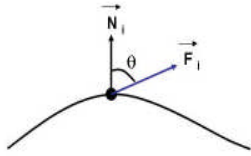

Figure 7: Force Vector relative to the Normal Vector

The stiffness dimension based on the 2 extremities is initialised before the simulation. The dimension is defined as

$$\begin{bmatrix} K_E \\ K_G \end{bmatrix}_i = E \sum_{i \in t} \frac{V_t}{{L'_i}^2} \begin{bmatrix} 1 \\ 1/2(1-v) \end{bmatrix}$$

, where E is the Young's Modulus and v is the Poisson Ratio of the material. The anisotropic extraction illustrates the possibility of employing multi-dimensional inner spring stiffness.

Volume Behaviour

In order to preserve volume during simulation, the spring dimension can be extended to address other properties such as bulk elasticity as a factor against volume variation during simulation. Bulk stiffness K_B at node i is:

$$K_{Bi} = \frac{E}{3(1-2v)} \sum_{i \in t} \frac{V_t}{{L'_i}^2}$$

Volume displacement during simulation can be derived based on the volume calculation employed in

(Hong et al. 2006). This calculation will be correct even when the surface becomes concave during simulation. Therefore, force at node i along its normal unit vector at a time step without any external force interaction on the surface is

$$F_i + = K_{Bi} \Delta V w_i \vec{N_i}$$

(3)

where, ΔV is the volume displacement, and w_i is the weighted constraints that control the distribution of the volume penalty force. The constraint is generally set to 1, which means the volume change affects all nodes equally but constrained by their respective bulk stiffness. This arrangement produces a global deformation as shown in figure 1.

Therefore, in order to correctly distribute the interaction force effect to the object surface, the weighted constraints have to be correctly distributed based on the interaction radius of influence where, the sum of the constraints is equal to the number of nodes. The radius of influence (Hong et al. 2006) has been modified and the interaction force orientation is introduced as the correction factor. If the surface nodes are within the radius of influence r, weight at node i is

$$w_i = \left(\frac{p_i - p_f}{\left\| p_i - p_f \right\|} \bullet \vec{F_f} \right) \left(cos \left(\frac{\left\| p_i - p_f \right\|}{r*2} * \Pi \right) \right)$$

(4)

where, p_i and p_f are the position vector of node i and the position of where force is imposed respectively. Since the sum of weighted constraints is equal to the number of nodes (Hong et al. 2006), the constraint values have to be normalised.

Upon interaction, the direction of the volume penalty force at node i depends on the position of node i relative to the point of interaction. Therefore, based on Equation (3) and (4), the penalty force at node i is

$$F_i + = K_B \Delta V w_i \frac{p_i - p_f}{\left\| p_i - p_f \right\|}$$

EXPERIMENTAL FINDINGS

The framework for the empirical experiments has been implemented on top of Microsoft Visual C++, OpenGL and OpenHaptics platforms. Phantom Desktop haptic device has been employed to provide the interaction and the desktop PC has the specification of Intel Pentium 4, 2.40 GHz and 1 G RAM. To evaluate the estimation method, four different distribution schemes have been compared:

- **Scheme A :** Uniform mass and inner stiffness
- **Scheme B :** Irregular mass and regular inner stiffness
- **Scheme C :** Irregular mass and inner stiffness (single stiffness dimension)

- **Scheme D :** The Proposed Scheme

Further comparisons are carried out against the Finite Element Model (FEM) and volume MSS (VMSS). The elasticity modulus is re-extracted from the model to evaluate if the proposed model emulates the same material behaviour.

Properties Estimation

The stiffness of the springs is based on the real elasticity properties of the object material. Therefore, to evaluate the proposed properties estimation scheme (Scheme D), the Young's modulus (E) can be derived from the strain and stress relationship of the proposed deformable model and compared to the original E. The same experiment is repeated for other surface mesh complexities. The values (Table 1) demonstrate that the deformable surface model can closely emulate the volume material behaviour.

Table 1: The Young's Modulus (E) extracted from the model when a constant force is imposed on the surface of a cube

Model (complexity)	Cube (602 nodes)	Cube (1352 nodes)	Cube (2402 nodes)
E (100 Pascal)	100.07	98.7	98.7
E(3.25 Kpa)	3.30	3.17	3.11

To evaluate the re-estimation of properties, a local surface area is refined and the properties are then re-estimated. A constant force is imposed on a node and the displacement data is collected. The displacement patterns within the coarse and the refined area are compared (Choi et al. 2005; Arnab and Raja 2007). If the patterns are identical, the deformation behaviour is preserved despite the change in the mesh topology. The displacement behaviours are studied where 2 values are analysed:

i) The standard deviation between the 2 patterns determines their level of co-incidence. The smaller the standard deviation, the more identical the patterns are
ii) The mean deviation of the 2 patterns represents the error in behaviour after refinement. The least value indicates the least deviation from the original behaviour

Figure 8 describes the findings based on a sphere model with irregular mesh topology to illustrate the concept.
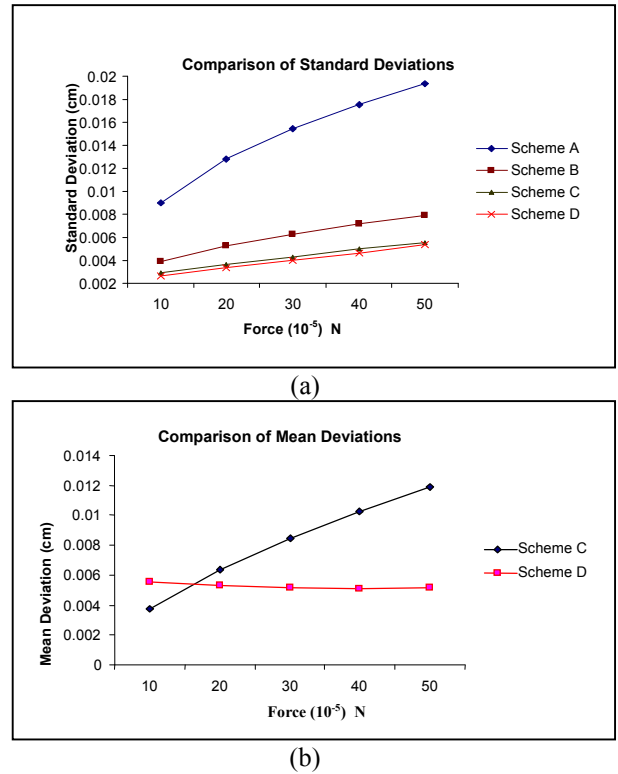

(a)

(b)

Figure 8: Properties Re-estimation Test (E=10 Pascal): (a) The standard deviation of the patterns (C and D denotes the most co-incidence) (b) Average displacement deviation comparison, where D maintains the least error

The standard deviations comparison in 8(a) shows that both scheme C and D produce high level of coincidence between the displacement patterns. However, scheme D maintains a minimal displacement deviation for any magnitude of force compared to scheme C where its deviation increases with force (8 (b)). Hence, the analysis concludes that the proposed scheme D preserves the properties and the local dynamic behaviour with minimal standard deviation when a local area is refined. Tests have been carried out on models with various mesh complexities and they also draw similar conclusions

Shape and Volume Preservation

To evaluate volume preservation during simulation, the global behaviour is observed. Figure 9 shows that the complete irregular scheme improves the surface behaviour with gravity.
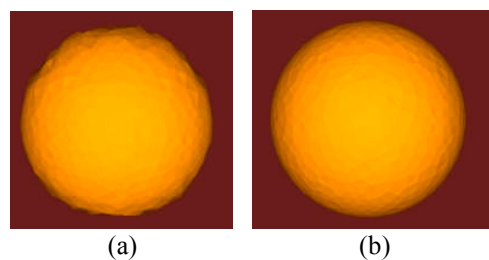

(a)   (b)
Figure 9: Gravity test: (a) Scheme B (b) Scheme D

When a constant force is imposed on the surface, the object shape changes and deforms. The analysis concludes that the irregular scheme preserves the object volume with the least volume deviation as illustrated by the average deviation percentage comparison in figure 10.
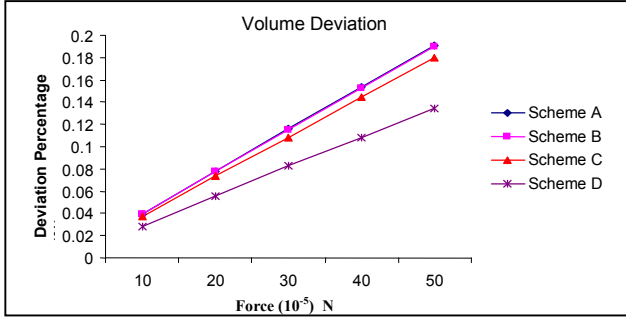

Figure 10: The percentage of the average volume deviation

Global Deformation Effect

To produce a more realistic global deformation based on the interaction force radius and orientation of influence, the weighted constraints are manipulated at runtime. When these constraints are set to 1 (figure 11 (a)), the global deformation as in figure 11 (a) is produced. Upon interaction, the weighted constraint at each node is changed during simulation in regards to the radius and the interaction force orientation at runtime. Consequently, figure 11 (b) shows that at runtime, the automatic constraint derivation based on the force orientation and radius of influence produces a more realistic behaviour.
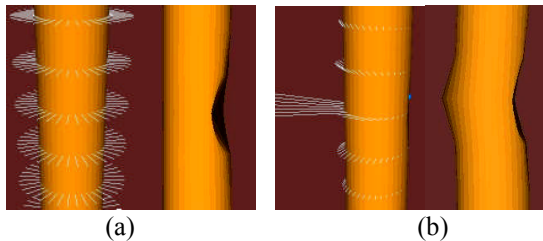

Figure 11: The white lines reflect the weighted constraints at the nodes (a) constraints are uniform (b) constraints extracted at runtime

Models such as cubes with different surface mesh complexities have been compared with the FEM model as analysed by SolidWorks/Cosmos. A shear force is imposed on the top surface while the opposite surface is fixed. When compared with FEM, the shape produced is similar with minor deviation (figure 12). The displacements of the selected nodes from the same location on both models also indicate minimal and subtle deviation.
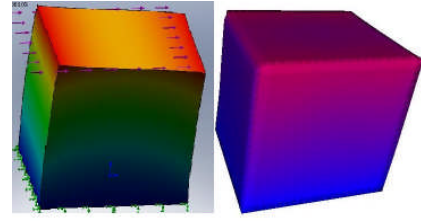

Figure 12: Shape comparison (a) FEM (b) Proposed Scheme

Performance

A simple performance test has been carried out to illustrate that the surface data reduces both mesh and computational complexity. Table 2 shows that the average frame per second (FPS) achieved by the VMSS and the proposed surface model.

Table 2: Total FPS Comparison

Model (number of surface nodes)	Cube (602 nodes)	Cube (1352 nodes)	Cube (2402 nodes)	Sphere (1000 nodes)
VMSS	77	37	20	47
Scheme D	80	44	25	77

CONCLUSIONS

This paper has illustrated the feasibility of employing a surface model to simulate soft solid. The proposed scheme introduces volumetric discretisation based on the node normal, the anisotropic extraction of inner spring relative to force orientation as well as weighted constraints to control global deformation effect. Local deformation behaviour is preserved regardless of the mesh resolutions and global deformation effect is achieved despite of the non-existence of internal volume. It displays similar behaviour with the FEM model and produces minimal deviation from the original elasticity modulus. Furthermore, the computational performance is better than the volume counterpart. Further evaluations such as the perception test will be carried out to verify the visual and haptic feedback experienced by the users. The scheme is currently explored in the ongoing research in breast simulation (figure 13).
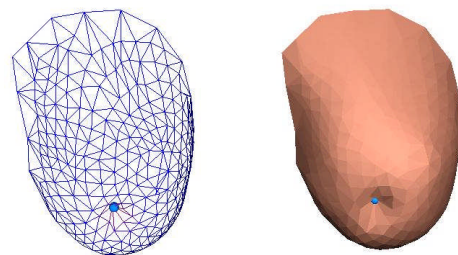

Figure 13: A Breast Model Deformed Upon Interaction

REFERENCES

Arnab S.; Raja V. 2007, Irregular Surface Mesh Topology for
Volumetric Deformable Model, The 4th INTUITION International Conference and Workshop, 4-5 October 2007 , Athens, Greece

Aubel A. and Thalmann D. 2000, Realistic Deformation of Human Body Shapes, Proc. Computer Animation and Simulation 2000, Interlaken, August 2000, pp. 125-135.

Balaniuk R. and Salisbury K. 2003, Soft Tissue Simulation using the Radial Element Method, IS4TM-International Symposium on Surgery Simulation and Soft Tissue Modelling, France, June 12-13 2003

Bourguignon D. and Cani M. P. 2000. Controlling Anisotropy in Mass-Spring Systems, Computer. *Animation and Simulation 2000*, 113-123

Bielser D. 2003. A Framework for Open Surgery Simulation, *Doctor of Technical Sciences Thesis*, Swiss Federal Institute of Technology, ETH, Zurich, 2003

Brown J.; Sorkin S.; Bruyns C. 2001. Real Time Simulation of Deformable Objects: Tools and Application*, In Comp. Animation*, 2001

Choi Y.; Hong M.; Choi M.; Kim M. 2005. Adaptive Surface-deformable Model with Shape-preserving Spring, Journals of Computer Animation and Virtual Worlds, Comp. Anim. Virtual Worlds 2005, number 16, pages 69-83

Delingette H. 1998. Towards Realistic Soft Tissue Modeling in Medical Simulation, *Proceedings of the IEEE: Special Issue on Surgery Simulation*, 512-523, April 1998

Deussen O.; Kobbelt.; Tucke P. 1995. Using Simulated Annealing to Obtain Good Nodal Approximations of Deformable Objects, *Computer Animation and Simulation '95*, Springer-Verlag, 1995

Gelder A. V. 1998. Approximate Simulation of Elastic Membranes by Triangulated Spring Meshes, Journal of Graphics Tools 1998, 3(2), pages 21-41

Hong M.; Jung S.; Choi M.; Welch S. 2006. Fast Volume Preservation for a Mass-Spring System, *IEEE Computer Graphics and Applications,* September/October 2006, 83-91

Laugier C.; Mendoza C.; Sundaraj K 2003. Towards a Realistic Medical Simulator using Virtual Environments and Haptic Interaction, Robotics Research: The Tenth International Symposium, Springer Tracts in Advanced Robotics Volume 6/2003

Lloyd B. A.; Szekely G.; Harders M. 2007, Identification of Spring parameters for Deformable Object Simulation, IEEE Transactions on Visualisation and Computer Graphics, Vol. 13, No. 5, Sept/Oct 2007

Maciel A.; Boulic R.; Thalmann D. 2003, Towards a Parameterization Method for Virtual Soft Tissues Based on Properties of Biological Tissue, In International Symposium on Surgery Simulation and Soft Tissue Modeling, 2003

Marchal M.; Promayon E.; Troccaz J. 2005. Simulating Complex Organ Interactions: Evaluation of a Soft Tissue Discrete Model, ISVC 2005, LNCS 3804, pages 175-182, © Springer-Verlag Berlin Heidelberg 2005

Mendoza C.; Sundaraj K.; Laugier C. 2002. Issues in Deformable Virtual Objects Simulation with Force Feedback, *International Advanced Robotics Program (IARP): International Workshop on Human Robot Interfaces*, Rome - Italy, 2002.

Nedel L. P. and Thalmann D. 1998. Real-Time Muscle Deformations Using Mass Spring Systems, Proc. Computer Graphics Int'l, IEEE Press, 1998, pages 156-165

Vassilev T. and Spanlang B. 2002, A Mass-Spring Model for Real Time Deformable Solids, Proceedings of East-West Vision 2002, pp. 149-154, Graz, Austria, September 12-13, 2002

Zhang J.; Payandeh S.; Dill J. 2002. Haptic Subdivision: an Approach to Defining Level-of-Detail in Haptic Rendering, Proceedings of the 10 Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, Orlando, FL, March, 2002, pages 201-208

AUTHOR BIOGRAPHIES

SYLVESTER ARNAB obtained an Honours Degree in Computer Science at The University of Manchester in 1998 and a masters degree in Distributed Multimedia Systems at the University of Leeds in 2001. He has worked at University Malaysia Sarawak (UNIMAS) in Malaysia as a lecturer and researcher for 5 years. He is currently reading a doctorate degree at the University of Warwick. His research interest is in deformable models for medical simulations. His e-mail address is: s.arnab@warwick.ac.uk

VINESH H. RAJA is currently a professorial fellow at the University of Warwick and is currently leading the Informatics Research group within the Warwick Manufacturing Group. His research interests are applications of ICT in healthcare sector. He focuses on augmenting and extending everyday, learning and work activities with interactive technologies that move 'beyond the desktop'. This involves designing enhanced user experiences through appropriating and assembling a diversity of technologies including haptics, handheld and pervasive computing. The main focus of his research is not the technology per se but the design and integration of the digital representations that are presented via them to support social and cognitive activities in ways that extend current capabilities.His e-mail is: raja_v@wmgmail.wmg.warwick.ac.uk.

VIRTUAL REALITY: THE WAY AHEAD IN INDUSTRIAL SAFETY

Konstantinos Loupos
Institute of Communication and
Computer Systems (ICCS)
9 Iroon Polytexneiou Street
15773, Athens,
Greece

Panagiotis Psonis
Institute of Communication and
Computer Systems (ICCS)
9 Iroon Polytexneiou Street
15773, Athens,
Greece

Dr Angelos Amditis
Institute of Communication and
Computer Systems (ICCS)
9 Iroon Polytexneiou Street
15773, Athens,
Greece

KEYWORDS

Virtual Reality, Virtual Environment, Industrial Safety, Simulation.

ABSTRACT

This paper focuses on the application of Virtual Reality (VR) technologies and Virtual Environments (VEs) on the industrial field regarding safety. The issues of the industrial field are initially presented, focusing on the safety tasks currently performed in the sector. The Virtual Reality technology's innovative character is then presented, focusing on how this technology can by itself aid the safety processes or in combination with other novel technologies. The latter approach provides significant aids to methodologies that are currently used towards industrial safety and the reduction of accidents. Under this scope, this paper will refer to training, safety management, risk assessment and accident investigation as the four main safety actions currently performed in industrial fields. The paper closes by presenting in general the applicability and innovative character of VR applied to the aforementioned four safety actions as well as presenting a concrete use case example.

INTRODUCTION

Significant amounts of resources are nowadays being spent from the industrial sector in order to reduce the causes and finally actual industrial accidents that if occurred could cause property damages, human injuries and even fatalities. Therefore industrial safety can be characterized of particular importance especially when talking about hazardous material environments (chemical, petrochemical, nuclear, etc). There are always several reasons for an accident to occur; however, it may need some thought and help to figure out what has gone wrong. Once the reason has been found though, it is possible to ensure future incidents' prevention. Some actions that can be mentioned inside the framework of industrial safety are those of training, risk assessment, accident investigation and safety management. These actions can include involvement of the actual plant operators, plant managers, emergency teams or control room operators. Additionally these actions can provide particular assistance to plant

managers to define the impact of certain decisions on individual (or group of) operators' work.

The Industrial Situation

There are a series of safety actions currently executed in industrial plants regarding Risk Analysis, Accident Investigation, Safety Management and Training. It can be said that the Risk Analysis and Accident Investigation can be divided into Hazard Identification, Consequence Assessment, and Accident Investigation and Accident Modelling. On the other hand, Safety Management can be defined as the control of risks that have been identified through risk assessment and on-going learning from incidents/accidents and is a more complicated task including audit, accident prevention and mitigation, and emergency response. Furthermore, a training system involves the design of curriculum, training materials and facilities (hardware, software, simulators, instructor facilities, courseware for students and instructors) aiming at having trainees develop knowledge, skills and behavioural practices to perform their work, meeting operational goals (Childs, 1999).

What have been described above are the related industrial safety actions that are usually performed in order to increase industrial safety. As can be seen these are quite complicated and time consuming tasks that up to now (with minor exceptions) have been performed requiring strong human intervention. Later in this publication, description of the issues currently present during the execution phase of these processes has been included.

Issues In Performing Safety Activities

Some of the issues that have been raised as significant obstacles and needs during the execution of the aforementioned safety actions can be summarized below:

- Need for support of training and test of resources with regard to minor/medium emergency response scenarios.
- Representing shared and mutually know tasks and norms / good practice.
- Coordinating between the safety management functions.
- Comparing results against "best practice" instead of standards.

- Lack of training of personnel.
- Insufficient awareness of risks and consequences among staff.
- Training involving real time simulation: fire fighting simulation; emergency response training.
- Support of accident analysis and what-if analysis.
- Need for simulating accident scenario with simulation of static and dynamic processes.
- Time constraints.

VIRTUAL REALITY AS AN ENABLING TECHNOLOGY

Concerning the identified aforementioned needs, various systems can be designed supporting a number of technologies and scientific fields from Human factors related issues up to technologies like Communications, networking, visualization, simulation, Human Machine Interfaces and others. Amongst them, one of the most important technologies to be used is Virtual reality. Recently VR and related tools have started to be used, mainly on a research level with promising results raising the interest of industry, governmental bodies and the society.

Virtual Reality technologies as a rapidly evolving and diversifying field can be regarded as one of the key-technologies regarding visualization and is currently used in a wide area of applications such as archaeology, entertainment, architecture, automotive, industry etc. Its increasing usage in various fields has not only aided in major improvements but it has also contributed in the identification of issues and needs in different application domains. Additionally the usage of VR technologies has been proven ideal as an effective design and training tool thanks to the immersiveness (the sense of "being there") and interactivity features that it supports.

Technology Insight

The importance of Virtual Reality in systems promoting industrial safety can easily be seen if we see VR as an integration point for the technologies and equipment used. On the one hand, the VR system needs to interface with the given hardware including projection systems, tracked devices etc. in order to interact with the users. On the other, lies the application level in which the VR system is responsible for handling the 3D data and coordinating a real-time virtual environment, implementing the logic and behaviour of the elements constituting the virtual world and more.

The innovative character of VR in the specific field stems from the fact that it can effectively address the issues pointed out in the previous section with a low cost. The problem of simulating dynamic accident scenarios in which participants can be trained for emergency responses and enhance their awareness of risks and consequences is reduced to the task of modelling the virtual environment. From that point on the VR system can recreate the environment and allow the users to interact with it freely. A key element, however, in the success of such a system especially when time constraints are applied and evaluated, is its immersiveness. This relies greatly on the input and display technologies used.

One of the characterizing features of virtual reality is stereoscopic vision. By feeding each eye of the viewer with a different image the spectator is able to better perceive the dimension of depth thus enhancing the system's realism. A serious drawback of this technology is the need for specific projection systems (polarized projection, laser projection etc.) that usually occupy large spaces and special glasses that need to be worn by the user and may prove cumbersome. Recent advances, however, have brought forward a new breed of auto-stereoscopic monitors that occupy the same size as normal TFT panels and produce stereoscopic images without the need of any extra equipment. The ease of use and installation will naturally improve the overall experience and allow for further penetration of VR technologies into industrial environments. Further to this, however, one can easily imagine exploiting such technology in the form of VR-enabled handheld PCs. The applications of such a device in real-life field operations are endless since the operators will be able to receive real-time 3d visualizations of information otherwise impossible to perceive (e.g. the temperature of a component). This will essentially lead to more efficient on-site decision making and lead to safer working environments.

On the other hand, equally important are the interaction devices used to manipulate the virtual world. Providing the user with natural and intuitive means of performing familiar activities is a crucial point towards achieving a truly immersive environment. Advances in tracking technologies (magnetic, optical etc.) have offered a lot to this goal. A wide variety of devices can be found today at affordable costs that allow for head, hands or even whole body tracking. Through such equipment (3D mice, data gloves etc.) it is possible to enhance the user's experience by allowing him/her to employ the same body movements as in real life in order to perform a specific activity in the virtual world. For example, using a pair of data gloves, the user could grab and rotate a virtual wheel as he/she would do in reality. In the same manner he/she could press a button by virtually touching it. Moreover, by tracking the body and/or head position of the user it is possible to update on the fly the viewing perspective of the virtual environment thus creating the illusion that the screen is actually a window to the real world. As research in this field progresses, such devices have grown to become

extremely accurate and non-invasive rendering their usage more than practical.

COMBINED TECHNOLOGIES INNOVATION

VR technologies are geared towards creating realistic and immersive virtual environments with which users are allowed to interact freely. However, for all but the simplest applications, this is not enough. A simulation, especially one aimed at industrial safety, will require collaboration with a large range of external applications in order to implement features such as environmental rules or scenario logic. Apart from that, it is important to seek for alternate technologies that can be combined with VR and possibly expand its effectiveness or fields of application.

External Simulators

When dealing with training, it is usually the case that we simply require from the trainees to follow a certain procedure by performing successive actions such as operating various plant components. However, when the simulation is conducted to drive risk assessment or accident investigation research it is mandatory to integrate the plant's behaviour in the virtual environment. Otherwise, it would be impossible to accurately recreate a potential accident case based on the participants' actions. Such functionality can be provided by external industrial simulators that are available in the market and allow for real-time, precise and reliable simulation of a plant's general product flow as well as individual component mechanics. On the other hand, as this type of software is designed to reproduce the "internal" operation of an industrial site, it fails to provide data regarding the consequences of an accident towards the environment. Imagine for example the case of a petrochemical plant in which a gas leakage has been identified. Although a plant simulator would identify it and update the related components accordingly (e.g. pressure drop in a pipe), no information can be produced regarding the emitting of gas to the environment.

Such data, however, is important in case one wishes to train the participants in safety critical situations or simply investigate the possible impact of a specific accident case. Therefore there rises the need for other types of simulation models such as gas dispersion models, fire spreading models etc. Through the integration of such software in the VR system, it is possible for example as soon as a gas leakage occurs, to have a realistic growth of the gas cloud and accurately trigger the on-field gas detectors which in turn will signal certain actions to be performed on behalf of the participants. In failure of the appropriate measures being taken, the gas cloud will expand uncontrollably and in case it reaches a source of high temperature, it could possibly ignite.

Depending on the nature and purpose of the simulation, a number of other assisting software packages could be integrated following the same principle. In order to achieve that, however, it is of paramount importance to ensure during the design phase of the VR system that maintains a modular character and easily allows for cooperation with external software. Such cooperation could be achieved in one of two ways depending on the available computer power as well as whether we wish to maintain a distributed or local character for the VR system. As a first option, the collaborating software could be directly incorporated in the system's codebase in the form of a software library. This solution ensures uninterrupted and latency-free data exchange although significantly raising the amount of required resources by the system. On the opposite side, the external modules could be used as standalone applications distributed in other machines. This requires that a network communication is established between the various modules and the main VR system and interfaces are provided for the data exchange.

Rule-Based Systems

Up until now, we have discussed the need for external simulators that will realistically implement the behaviour of the virtual environment thus increasing its immersiveness for the participants as well as its credibility towards the safety analysts. However, this environment is static in the sense that the plant's behaviour for example remains the same during different simulation sessions in which many details may differ such as the number of participants, objectives and others. This scenario logic can be implemented through a set of rules that define the consequences when certain criteria are met.

This need can be covered by another external module developed using rule-based technologies. Specifically it is designed to hold an internal set of attributes representing the parameters to monitor. Furthermore it allows the user to import a set of pre-defined rules that specify what actions the system should trigger when specific state changes occur. Therefore, when the VR system reports some change of the monitored attributes, the module checks if a rule applies to the new state and reacts accordingly. The communication of this module with the main VR system follows the principles put out in the previous paragraph. In this case, however, integrating the rule-based module in the main system appears to be the preferred solution as it is expected to be very lightweight in comparison to one of the simulators mentioned above.

Novel Interaction Devices

It is evident that in such simulation environments as the ones discussed here, time is a key notion. Often, sessions are run simply to measure the time that a

participant required in order to complete a task or set of tasks. Such measurements, however, are prone to errors due to the simple fact that a virtual environment, regardless how close, never feels 100% real. Naturally, much effort is being put into reducing such distorting factors to a minimum by increasing the immersiveness of the environment. Towards this goal, a very crucial role is played by the users' interaction devices some of which were shortly presented earlier.

It is important to always keep an eye for new interaction techniques that could possibly enhance the virtual experience. Fairly recently, Nintendo revolutionized the way we interact with video games by offering a novel input device, the Wiimote and Nunchuk, along with its gaming console, the Nintendo Wii. It consists mainly of a "wand-like" joy-pad, the Wiimote, that apart from the usual buttons is enriched with accelerometers and an IR camera. Using two IR light sources, the Wiimote is able to calculate both its position in space as well as its orientation. An additional device accompanying the Wiimote is the Nunchuk which resembles a 3D mouse bearing two classical buttons and a joystick as well as accelerometers. This is an excellent example of an alternate-purpose device that has lots to offer to virtual reality. It provides tracking functionalities and can be used to recognize hand movements rendering it ideal for tasks such as operating plant components, positioning objects in the environment, issuing hand signals etc. through a more natural and intuitive manner. Furthermore, the classical controls provide an effective means for executing tasks with no real-life counterpart such as controlling a GUI.

Human Factors Technologies

In every industrial environment, it is humans that form the core of the system's uninterrupted and flawless operation throughout its life-cycle. On the other hand, human actions at either the sharp or the blunt end can become a source of risk. It is therefore mandatory for simulation systems that deal with safety issues to integrate human factors components as an indivisible part of the applied methodologies. Such components will be responsible for assessing the way in which performance variability and deviations may occur as a result of individual actions and thus drive the decision making in regards to developing effective countermeasures.

A VR system, as described until now, provides a controlled environment ideal for collecting the human factors related data required to reach safe decisions. However, it is impractical, and in the case of certain data, impossible to manually record the desired pieces of information. Consequently, it becomes necessary to design the VR system in a way that it is able to automatically store a selected range of relative data. This can be achieved through a logging process able to save a time-stamped history of all notable events such as certain actions of the participants, changes in the properties of environment elements etc.

Although the logging process makes sure that all data needed by human factors technologies (or other simulation data) are collected and stored, they still remain in raw, unprocessed form significantly hindering their direct exploitation. In order to compensate for this problem, the VR system can be coupled with one or more specialized post-processing tools whose goal is to receive the log produced by the simulation and present the desired information directly in human readable form thus automating the process yet more. Further to this, a special-purpose database could be designed as a permanent storage for data collected in past simulation sessions. This history of collected information can be used to produce valuable statistical data as well as provide an indication on the validity of current results thus allowing for example to decide on whether the respective session needs to be repeated. Additionally it gives the safety analysts the opportunity to perform validation tests (through the related sessions) and thus test the efficiency of documented procedures in their real case study by comparing the expected with the actual responses (human, plant, etc).

VR AS A TOOL TO ACHIEVE INDUSTRIAL SAFETY

There is no doubt that the technological status of Virtual Reality systems has a lot to provide to the general aspect of industrial safety. Presented below are applications of VR systems combining the technological advances towards the aforementioned features implementation and particularly safety related tasks.

VR In Training Simulators

Virtual Reality simulators can provide realistic behaviour of the training task and the behavioural processes required for the particular task, to fulfil aspects of psychological fidelity as published by Sian & Robertson, 1999 but also impose the related psychological stress. At the same time such simulators can cope with the aspects of Physical fidelity meaning the degree to which real-world operational equipment is reproduced. This can be provided by advanced VR training simulators being able to model the virtual world (or training environment) as close as possible to reality in order to attain realistic training conditions. They provide thus the ability for the model to behave as the real equipment does and also react to user actions in the same way so that technicians should to be able to perform in the synthetic environment all those relevant actions they would perform in the real world (Stüring, et al, 2002).

Furthermore this kind of simulators can not only provide advanced visualisation in order to represent the simulated environment as close to reality as possible.

These simulators can also expose the trainee to the full complexity of the task, with not any guidance or feedback at all. The idea of this approach is to foster resourcefulness, problem solving and creativity in the learning process. This could be a very valuable attribute in commissioning, maintenance, or emergency environment where tasks often have to be done for the first time under time or commercial pressure (Stüring, et al, 2002). Additionally, time scaling can also be simulated being an important feature during training activity to help operators familiarizing and getting used with critical and particularly fast events. This can be done with the use of VE systems enabling to adjust reaction times at will of trainees. Other important features that would aim the training session are: zooming/hiding (being able to see things that nobody is allowed to see in reality), colouring (colour pipe paths, routes, flows, etc) and display real-time tags (tag components and their behaviour, especially when combined with process simulators).

VR In Risk Assessment And Accident Investigation

In this safety action, the great potential usage of VR can be divided into events involving technology-related hazards and events involving human-related hazards. One important aspect is the familiarization with the plant layout and equipment design, and thus the visualization and navigation (walk-through) to enhance analysts' and operators' knowledge on the layout of the plant.

Additionally through a VE simulating an industrial plant, we can have real-time response of fully emulated processes that allow analysts to execute the simulation and analyze options ("what if" scenarios). This can of course be combined with calculation of the volume of liquids inside pipes and vessel and thus enable to see things and processes in a way that is impossible in reality (e.g. volume, type, pressure and temperature of liquids flowing in pipes). This can be done simulated as a whole in a VR system or combined with the usage of an external industrial simulator interfacing to it.

Identification of fire and explosion sources (sparks, friction, etc) can also be of particular importance through a Risk Assessment test case. This can also be simulated with a VR system and provide the necessary information to the safety analyst or even the plant manager so that they take the necessary precautions in case of an explosive gas release or similar situations.

Another feature that VR can provide in relation to technology-related hazards are first of all error detection & revitalization. Errors can be detected and classified via a VR system by comparing the simulated result with the documented (or expected one). This can be done by combining VR technologies with task execution and monitoring mechanisms and/or rule-based systems. The analysis and actual classification of the error can be done also by post-processing tools (of the data gathered in the VE) and technologies.

Concerning procedures in general inside an industrial environment and specifically the planning stage, a safety analyst can get useful information from a VR simulation by being able to investigate correctness of planned procedures of maintenance (on single parts or the whole plant). Applying the maintenance procedure on virtual components inserted in the system and then evaluating the functionality after the maintenance, one can get feedback directly from the VR system by showing the consequences in the selected point of interest.

VR In Safety Management & Audit

The main use of VR technologies are in accident prevention and handling of emergencies, helping safety managers identify possible countermeasures and barriers to ensure that the accident will not happen or minimize its consequences. Using VR, different barriers (that could prevent an accident) can be highlighted or removed during the simulation and thus give the ability to analysts to evaluate their effectiveness analyzing the consequences. These barriers can be physical (eg a wall, door, etc), symbolic (eg lighting, alarm, etc) and functional (eg interlock).

Moreover and under the framework of emergency response and awareness of the situation and communication, VR can help the safety analysts to assess how the orders are adapted in handling emergency procedures. In this way the safety analyst can not only select the suitable emergency team and see the results but also trace different approaches depending on the emergency situation. VR can also offer previewing options imposing thinking of a few options, identifying key personnel to bring new information and finally identifying dangers and problems associated with each option. This way, comparing options abilities are provided in terms of a set of criteria. More analytical comparisons can follow the initial approach in terms of a set of criteria such as, time required to implement options, risks associated with options, resource requirements, allocation and re-allocation of responsibilities and so on.

Another important feature that a VR system can provide is the plant familiarization particularly with the location of emergency exits and other important points. This feature is also related to the training as a safety action as well. One of the most important aspects of evacuation planning is the familiarization of operators with the emergency exits, in normal conditions and especially in altered environment (e.g., fumes and dust in the air). Through VR, the trainee can visualize a path that leads them to the first available emergency exit. The versatility of VR technologies can also allow for creating new hazards during emergency situations

already envisioned. For instance, operators may have received certain instructions on how to evacuate an area but the evolution of emergency may create addition hazards which must be detected and avoided by the operators while on the move.

Finally close monitoring of works executed inside the plant, compensating for errors and coordinating the work is another feature of particular importance as the ability to monitor the work and the communicating information is a frequent demand for managing emergencies. Virtual reality can facilitate team monitoring and coordination whenever possible to have instantaneous access to all team positions and communications.

PRACTICAL USE CASE

In this section, we will describe a potential use case that employs a sub-set of the above mentioned technologies in order to enhance the training of new or existing personnel.

The working environment is that of a chemical plant and the targeted personnel are Field Operators (F.Ops) and Control Room Operators (C.R.Ops) who are required to familiarize or gain additional experience with new or existing equipment and procedures. F.Ops are responsible for carrying out tasks on the actual components of the plant such as safety checks or maintenance. On the other hand, C.R.Ops are stationed in a remote control room that allows them to oversee the plant's behaviour and guide the F.Ops in their actions.

A potential and rather frequent scenario would be for a C.R.Op to be alerted by the system for a hazardous situation possibly caused by a malfunction in one of the plant's components. This would trigger a series of protocol events as he/she would have to request from the nearest F.Ops to move into the area and investigate. In turn, the F.Op after approaching the general area would have to act as instructed in order to confirm that the alarm is valid as well as locate its exact source and identify whether the problem is caused indeed by a failure in one of the components. Furthermore, depending on the severity of the situation, a number of steps must be followed so as to safely contain the situation and if possible correct the malfunction. Such a scenario would either end with the problem (temporarily or not) fixed and perhaps some maintenance request issued, or with an escalation of the accident resulting in drastic actions such as plant evacuation and/or emergency shutdown.

Executing training drills based on such a scenario has the obvious benefit that the trainees become accustomed to the emergency processes that must be followed as well as the mechanics of each step. Further to this, however, there is a number of other valuable data to be collected. For example, one can measure reaction times

for critical tasks which could lead to reorganising safety procedures or even reassess the ergonomics aspect of the working environment based on observations under time pressure and stress.

On the other hand, holding such drills in the real-life environment is extremely difficult as many of the requested actions interfere with the plant's product flow. This is both highly dangerous in case a human error causes an actual accident but potentially costly as well since interrupting the product flow could be translated in an immediate loss of hundred thousands or even millions of euro.
Consequently, the use of a virtual environment in which such a training session can be reproduced is an ideal solution. In order, however, for it to produce reliable results that can be extrapolated to real life, it is important to provide both an accurate simulation of the actual workspace as well as an immersive user interface.

A VR system designed for such a task would reproduce a realistic 3D model of the plant's components and surrounding environment in a large screen or projection wall covering the trainees range of sight enabled with stereoscopic vision to enhance the depth perception. The user could control motion with a simple joypad such as the ones used for videogames but also utilise more sophisticated tracked devices (data gloves, head trackers) in order to interact with elements of the environment. In order to guarantee the realistic behaviour of the environment, the virtual environment would further have to be linked to an industrial simulator responsible for calculating in real-time the properties of each of the plant's components. Most industrial sites already use such software products so one needs to build the communication layer between the two software packages. A rules-based system could also be required in order to replicate standard events that cannot be reproduced by the simulator e.g. a worker closing a specific valve when some pressure value reaches a given maximum. Finally, additional components could be needed such as a gas-dispersion model in order to simulate the environmental conditions after an accident has occurred.

Holding training sessions with such a VR installation would produce a wide range of raw data such as the timestamps of a series of events. In order, however, to provide the safety experts with the critical information needed to feed the decision making process, human factors technologies will need to be applied and reform these raw results into high level pieces of information such as the statistics of the occurrence of certain events.

CONCLUSIONS

In this paper the innovative character of Virtual Reality systems has been presented, having as a main focus the industrial safety. The current status concerning the industrial situation has also been

presented indicating the main industrial issues currently present in the field and also the missing gaps in the industrial sector. The VR technologies as an innovative technology concerning industrial safety have also been detailed including a draft presentation of the features that could show particular interest in industrial applications and particularly industrial safety. Then the usage of the aforementioned technologies and innovations has been presented including how the application of VR systems could aid particular processes regarding industrial safety.

REFERENCES

Childs JM, et al. - *Handbook of Human Factors Testing and Evaluation*, 2002

Gramopadhye, A. K., et al. 2002 "Using Training to Improve Performance of Inspectors on the Hangar Floor", *16th Human Factors in Aviation Maintenance Symposium, April 2-4, 2002.*

INTUITION NoE – CRF: "Research Position Paper: Towards Virtual Product Creation: the role of VR in the Automotive & Transport Industry", *21/02/2006, INTUITION NoE WG2.3.*

Jefferson, et al 1995. "Automated hazard identification by emulation of hazard and operability studies", *Proceedings of the 8th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems. Melbourne, pp765-770, Gordon and Breach Science Publishers.*

Loupos K, at al 2007, "VR, HF And Rule-Based Technologies Applied And Combined For Improving Industrial Safety". *HCI 2007 Conference Proceedings volume 6, LNCS_4555, ISBN: 978-3-540-73280-9.*

Loupos K. et al. 2007, "Application Of VR And HF Technologies For Improving Industrial Safety", *NHIBE conference proceedings, "New Horizons in Industry and Education", 30-31 August 2007, Rhodes Island, Greece.*

OSHA – Small Business Handbook, *www.osha.gov*

Shang X, et al. 2006. "Integrating Vr And Knowledge-Based Technologies To Facilitate The Development Of Operator Training Systems And Scenarios To Improve Process Safety", *ESREL 2006 Conference "Safety and Reliability for Managing Risk", 18-22 September 2006, Estoril, Portugal.*

Stüring, et al, 2002. "Learning by doing: If at first you don't succeed….", *4th International Conference on New Educational Environments, ICNEE, Lugano, 2002.*

AUTHOR BIOGRAPHIES

Konstantinos Loupos holds an MSc in Microelectronics Systems Design (University of Southampton UK 2002-2003) with Distinction. His degree thesis was on the setup and programming of an Embedded PC (with PC/104 interface) for testing of ULAs (Uncommitted Logic Arrays). He also holds an MEng in Electronic And Electrical Engineering (University of Manchester 1998-2002). This latter degree thesis has covered the implementation of a cryptographic algorithm in a Smartcard as well as the design of the related PC (SW) interface and has taken place in Intracom S.A. (Greece). The last four years he has been full-time employed in the Virtual Reality team of the Institute of Communication and Computer Systems (ICCS) in Athens, Greece. His main activities are management and administration of EC projects and also VR development of systems and applications. He has participated in various related conferences with publications as the main author in most of them. His scientific interests range in Electronics, Embedded Systems, Smartcards, and Computer Programming.
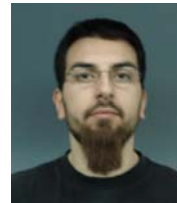
Panagiotis Psonis graduated in 2006 from the Imperial College London (UK) receiving an "M.Sc. in Advanced Computing". The degree thesis involved the research and implementation of contemporary techniques used to enhance realism in computer graphics applications. In 2005 he graduated from the University of Piraeus (Greece) obtaining a "B.Sc. in Computer Science". The final year project required the co-development of a graphical platform that provided the means for comparing AI algorithms. The last two years he has been full-time employed in the Virtual Reality team of the Institute of Communication and Computer Systems (ICCS) in Athens, Greece. His interests lie in the fields of Computer Graphics, Computer Architecture as well as Theory and Design of Programming Languages.

Dr Angelos Amditis has obtained the Diploma in Electrical and Computer Engineering from National Technical University of Athens - NTUA (Greece) in 1992, and a Ph.D. in Electrical and Computer Engineering (Telecommunications) from NTUA (Greece) in 1997. He has been teaching in various courses (communication and computer networks, communication theory etc.) of the Electrical and Computer Engineering Dep. of NTUA and of the Hellenic Naval Academy for the last 5 years. He is employed to ICCS since 1998 and has currently the position of Research Associate Professor. His research interests include model simulation, software and hardware engineering, HMI design and development, Evaluation and consolidation using VE technologies, VR/VEs etc. He is scientific responsible in more than 25 European Research projects, 15 of which in the IST area between them and the VIEW project while he was the coordinator of one other 5th Framework Research project.

ARCHITECTURAL MODELING WITH REUSABLE INTERACTIVE 3D OBJECTS

Jacek Chmielewski
Department of Information Technology
Poznań University of Economics
Poznan, Poland
E-mail: Jacek.Chmielewski@kti.ae.poznan.pl

KEYWORDS

Virtual Reality, modeling, interactive 3D objects, interaction model, metadata, XML.

ABSTRACT

3D modeling is extensively used in many engineering fields, notably in architecture. 3D models of new buildings are built with 3D objects taken from object libraries provided by producers of real objects (i.e. windows, lifts, etc). Such libraries can also include interactive 3D objects that are not only static geometrical representations, but also behave like their real counterparts. Searching for such interactive object in a large library requires object metadata describing object interaction properties. Existing metadata standards are not sufficient for describing object interactions. In this paper a new solution for interactions metadata is presented. The new solution is composed of concepts and tools for modeling an interaction, describing different aspects of an interaction, building XML documents with interaction metadata and using information contained in these documents to search specific objects. The usage of all those concepts and tools is presented in a detailed scenario from the field of architectural modeling.

INTRODUCTION

Modeling techniques based on virtual reality and 3D representations of real objects are widely used in mechanical design and architecture, among others. The time required to build such 3D models can be greatly decreased by the use of libraries of predefined components, which can be simply reused instead of designing them from scratch for every new model. This fact is especially visible in architectural modeling, where different elements of a newly designed building (i.e. windows, doors, lifts, etc) are usually provided by producers of these elements.

3D models of new buildings are often used not only for designing purposes, but also for generating visualizations and publications for marketing purposes. Often, such marketing visualizations include fly-by animations or presentations of specific features. However, it is possible to go a step further and provide potential clients and end-users with 3D immersive and interactive experience, by allowing them to walk through their new office, shop or apartment before it

actually gets built. Such approach can be used not only for demonstration, but also for testing the newly designed space and optimizing some features, like lighting conditions, ergonomics, etc.

3D interactive model requires components (i.e. reusable objects) that are not only geometrical representations of their real counterparts, but are also interactive and behave like real objects. Such interactive objects (Walczak 2006) can be also provided by producers of real elements in a form of object libraries, in the same way as they do it with static objects. However, searching for interactive objects in large object libraries is more complex. With static 3D objects it is enough to have some classification of objects and simple search engine that is capable of using this classification as object semantic description, along with geometry characteristics extracted directly from object 3D geometry. With interactive 3D object there is one additional element that has to be taken into account: object behavior. For interactive presentations, object behavior is usually the most important element, due to the fact that object interaction characteristic is used by designer to decide whether the object is suitable for the new design. In most cases object interaction characteristics cannot be inferred from the object definition. It is required to introduce additional description about the object – metadata. Metadata capable of describing all possible interactions of an object and will allow formulate queries that address specific interaction properties.

BACKGROUND

The topic of multimedia object metadata is well researched and developed. There are a number of metadata standards that can be used to describe object general properties or object format and purpose specific characteristics. The most general metadata standard is Dublin Core, which is a set of fifteen elements designed to foster consensus across disciplines for the discovery-oriented description of diverse resources in an electronic environment. Dublin Core Metadata Element Set (DCMES 2008) is intended to support cross-discipline resource discovery and it does not satisfy every possible declarative need in every discipline. Thus, in most applications it is used in a combination with more advanced, domain-specific metadata standard. DC elements are usually a subset of resource metadata and

are used as a minimal metadata for data exchange and discovery.

Interactive 3D objects usually contain many multimedia components like: images and video as textures, audio, text, complex 3D geometry, therefore the complete description of such objects may be composed of various domain specific metadata. For still images EXIF (EXIF 2002), DIG35 (DIG35 2000) or NISO Z39.87 (Z39.87 2001) can be used. Video can be described with AAF (AAF 2005), MXF DMS-1 (MXF 2004) or P/Meta (P/Meta 2007). Audio components of the interactive 3D object can use MusicBrainz (Swartz 2001) or simple ID3 tags (ID3 2000). However, complex multimedia resources like interactive 3D objects are usually described with the most universal metadata standard for multimedia objects – MPEG-7 (Martinez 2004; ISO/IEC 15938). It addresses not only general and format specific information, but also object semantics, geometry and spatio-temporal composition. MPEG-7 was formulated with audio and video contents in mind and its handling of 3D objects is limited. Nevertheless, in the past few years, some 3D metadata solutions were developed (Lorenz at al. 2006; Pitarello and de Faveri 2006; Bilasco et al. 2005). Especially worth mentioning, is the 3D SEmantics Annotation Model [3DSEAM] (Bilasco et al. 2006) developed by a group from Laboratoire Informatique de Grenoble [LIG], lead by Ioan Marius Bilasco. 3DSEAM extends MPEG-7 localization descriptors with Structural Locator and 3D Region Locator tools, making MPEG-7 the best available metadata solution for 3D multimedia objects.

However, even with such great pool of metadata standards and solutions like MPEG-7 there is no universal and extensible solution capable of describing interaction properties of interactive 3D objects. It is due to the fact that interactive objects change their state according to interaction results and interaction context, and new object parameter values cannot be determined a priori. Existing metadata solutions where designed to handle non-interactive resources and are focused on a single state of the described object. Therefore they cannot be efficiently used for interaction metadata

INTERACTION METADATA

As stated in the previous section, existing metadata solutions like for example MPEG-7 with 3DSEAM extension (Bilasco et al. 2006), can be used for describing object semantics and geometry. However they are not sufficient for describing object interactions. Existing metadata standards for multimedia objects were designed to describe a static or linear content, i.e. still images, documents or movies. In case of interaction descriptions the problem is more sophisticated. An interaction is not a simple 'content'. Interaction influences modification of the content, i.e. the object, but interaction itself is an action based on object behavior. The behavior is represented as an algorithm expressed by a computer program. The interaction metadata does not have to describe the whole algorithm, which may be very complex or confidential. To enable analyzing object interaction properties and chains of subsequent interactions it is enough to describe only the result of execution of such computer program. Existing metadata solutions are not applicable for describing multiple states of an object (i.e. all possible execution results). Therefore there is a need for a new metadata solution.

To deal with this situation, a new approach to interaction metadata is proposed. The proposed approach is based on following assumptions. To be interactive, objects need to have a behavior. Behavior is encoded in a form of a computer program. Computer program of an interactive object has at least one communication interface – an API, which allows exchanging information with the 3D virtual world or other interactive objects. Such API includes a number of functions and attributes that reflect the state of an object. Knowing the API and the computer program it is possible to precisely describe object interaction characteristics and possible object interactions, as well as match compatible objects. Following assumptions stated above, two new concepts are introduced: Interaction Interfaces and Multimedia Interaction Model. These concepts are accompanied with one additional tool called Interaction Metadata Query sub-Language.

Interaction Interface

The Interaction Interface concept is used to describe the interface of an object in terms of functions and attributes related to object interactions. Both elements of the interface are represented by object parameter types of the Interactive Interface and are accompanied with semantic descriptions. The Interaction Interfaces are used in metadata as object interaction characteristic and as a dictionary of parameter types referenced in the Multimedia Interaction Model.

An interaction interface groups object parameter types that have similar semantics, e.g. object parameter types related to object visual characteristics or object parameter types related to object internal components. An object implements a given II if the set of object parameter types is a superset of the set of parameter types of this II.

The proposed solution provides Interaction Interface Definition Rules that describe how to define new object parameter types and how to group them in new interaction interfaces. Single object parameter type definition is composed of an identifier, specification of a data type of values of object parameters of this object parameter type and optional specification of arguments required to evaluate the object parameter function. The technical part of parameter type definition is accompanied with semantic description or relation to an external ontology. While the technical part is required to build metadata, the semantic part is needed to allow search engines properly interpret interaction metadata and match object technical parameters with a semantic query.

Multimedia Interaction Model

The Multimedia Interaction Model is a conceptual model of an interaction described from the point of view of an object. It means that the interaction description is build around object reaction and a context that triggers the reaction. This approach comes from areas of computer science dealing with active environments, especially active databases. Research on the active databases resulted in paradigm called Event-Condition-Action (ECA) (Dayal U. et al 1988), which was introduced in late eighties by the members of the HiPAC project. The semantics of ECA rules are straightforward: when an event occurs, evaluate the condition associated with the event; and, if the condition is satisfied, trigger the action. The ECA rules are used up to date to express active DMBS features and event-reaction oriented features in other systems (Bry and Patrânjan 2005; Papamarkos et al. 2003; Thome et al. 2005; Zhou et al. 2004). The way the ECA paradigm is used in the Multimedia Interaction Model is the following. The Event element corresponds to the trigger of an interaction. The trigger is a change of a selected property of some object or a virtual environment in which the interaction occurs. The Condition element describes the context of an interaction. The context represents a state of selected objects or the virtual environment that is required to allow the action. The Action element matches the object reaction. The reaction describes how the object changes itself in a reply to the event in a given context.

Event

An event is a change of the state of an object. The object is called: trigger object. In the MIM model, the Event component contains a set of preconditions that need to be satisfied to trigger a particular interaction. The preconditions include a specification of a 3D interaction space and a non-empty set of object parameter types. To satisfy preconditions, the event has to occur inside a specified 3D interaction space and object parameters, which values are modified, have to belong to one of specified object parameter types.

Condition

A condition is a set of requirements set on the states of trigger object, the environment and interacting object. A condition needs to be satisfied to trigger an action. In the Condition component of the MIM model, a mathematical description has a form of logical expression. The logical expression represents all condition requirements. Semantic description of a condition may be used as a substitute or an enhancement of the mathematical description. The Condition element of MIM metadata is optional and can be omitted. In such case an interaction will be executed at each occurrence of the event.

Action

An action is a change of the state of an interacting object. In the MIM model, the Action component describes the outcome of an action, a new object state, expressed as a semantic description optionally accompanied by mathematical description represented by mathematical expressions used to calculate new values of object parameters. An action can be composed of an arbitrary number of sub-actions that provide more detailed descriptions. Such approach allows building action descriptions that can be evaluated on different levels of details.

Interaction Metadata Query sub-Language

The Interaction Metadata Query sub-Language goal is to provide easy way to access and use information available in object interaction metadata. IMQL provides a set of operators that can be used in simple keyword comparisons and advanced semantic searches. IMQL syntax relates directly to the MIM model structure and provides references to interaction interfaces of compared objects and parameter types of these objects. Object matching queries can be complex and difficult, especially in the field of interactions. The use of interaction interfaces facilitates object matching queries. The IMQL integrates with interaction interface definitions enabling the use of new object parameter types and value data included directly in the query. IMQL is composed of two parts: first, defining the source of object metadata; second, specifying filtering criteria. The first part can be replaced by results of another search query, while the output of the filtering criteria can be passed for further processing. Such solution allows embedding an IMQL query within larger query formulated in a different query language and performing a complex search by a single search system.

The MIM metadata has the most expression power when combined with more general multimedia metadata standards, like for example MPEG-7. Therefore the IMQL has to be used in conjunction with query languages capable of processing these general metadata standards. As mentioned, the IMQL can be embedded in queries written in other query languages forming a complex query. Complex queries enable search systems to look into standard metadata descriptions and interaction descriptions in a single process.

To provide universality of IMQL embedding rules, an IMQL query is embedded as a function. The imql() function takes object metadata URI and IMQL WHERE clause contents as arguments. The function returns Boolean value equal *TRUE* if given metadata matches the specified IMQL *where_expr*. IMQL function syntax is the following:

$$imql\ (\ object\_metadata\_URI,\ where\_expr\ ) \qquad (1)$$

Embedding IMQL query as a function allows including IMQL in an arbitrary query language: from XML

related XQuery, to RDF query language SPARQL, to SQL used for querying relational databases.

SEARCHING FOR INTERACTIVE OBJECTS

Consider an architecture studio where engineers work to design new buildings. Each building design is composed of many components, like for example walls, windows, doors, lifts, stairs, etc. All these components are virtual models of real objects that will be used during the development of a new building. Some of these components can be interactive and have embedded behavior that makes this object act like its real counterpart. Example of an architectural object with embedded behavior is a model of automatic 'sliding door'. Consider a 'sliding door' model that *opens* itself when someone approaches the door and *closes* itself automatically. It can be *locked* in open or closed state, which means that the door can be permanently closed or permanently open. The actions of door opening and closing are visualized as animations and complemented with appropriate sound. The diagram of such object is presented in Figure 1.
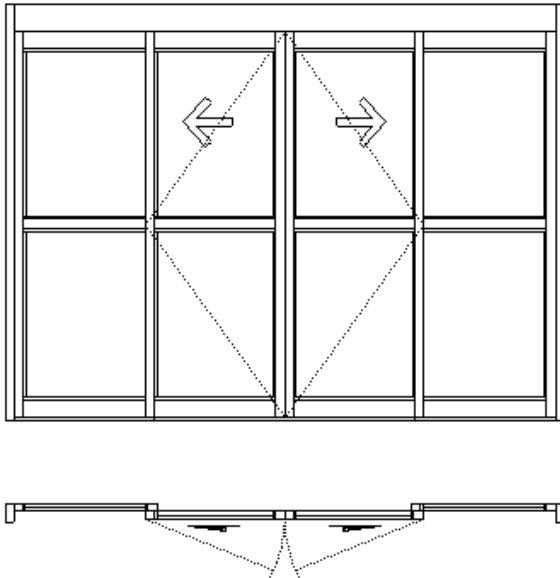
Figure 1: 'Sliding door' diagram

The 'sliding door' virtual model has two interactions: 'open' and 'close'. First interaction is triggered by an object moving in the area monitored by door motion sensor. Second interaction, 'close', is triggered by an object leaving the monitored area. The *trigger object* of both interactions can be any object big enough to exceed the sensitivity limit of the door motion sensor. The 'sliding door' can also be *locked* in the closed or open state. The locked parameter is a representation of object internal state and there is no specific interaction for changing the value of the *locked* parameter.

Parameters used in the 'open' and 'close' interactions include *PositionCenter*, *SizeBoudingBox* and *PlayedSound* from general 'Visual Interaction Interface' [V-II] and 'Aural Interaction Interface' [A-II], and *Closed* and *Locked* parameters from a custom interaction interface called 'ArchItems – Door

Interaction Interface' [AIdoor-II]. The AIdoor-II interaction interface defines the *Closed* and *Locked* parameters as binary flags with Boolean data type. Both, A-II and AIdoor-II interfaces are fictitious and introduced specially for this scenario. V-II is a general predefined interaction interface.

Both interactions of the 'sliding door' object are similar. This example describes in detail the first interaction – 'open'. The Event metadata element of the 'open' interaction contains specification of a parameter whose value change triggers the event, and definition of the 3D interaction space. The event of the 'open' interaction is triggered by a change of the *PositionCenter* parameter of an object located inside the 3D interaction space. The 3D interaction space has a shape of a cone with the apex located above the door and the vertical axis directed down at an angle of 45°. Excerpt of XML document with Event element description is presented in Listing 1.

```
<mim:Event>
  <mim:InteractionSpace>
    <mim:Viewpoint position="0 1.25 0"
          fieldOfView="0.785398" orientation="1 0 0 -2.355"/>
    <mim:Semantics>
      <mpeg7:FreeTextAnnotation>
        Door motion sensor monitoring range
      </mpeg7:FreeTextAnnotation>
    </mim:Semantics>
  </mim:InteractionSpace>
  <mim:Parameter typeID="V-II.PositionCenter"/>
</mim:Event>
```

Listing 1: Excerpt of XML metadata of the 'open' interaction: Event element

The Condition element of 'open' interaction metadata contains a mathematical expression which checks if the size of the trigger object is above the threshold required to execute the interaction. The size parameter value is bound to the S symbol, while the threshold is represented by a numeric value 0.32. The mathematical expression written MathML refers to single components of object size (S) and compares its product with the threshold value.

$$S.x \cdot S.y \cdot S.z >= 0.32 \qquad (2)$$

Other conditions that have to be met to execute the 'open' interaction refer to the state of the 'sliding door'. The door has to be *closed* and *not locked*. These conditions refer to *Closed* parameter value bound to C symbol and *Locked* parameter value bound to L symbol. XML representation of the Condition metadata element is the following.

```
<mim:Condition>
  <mim:Expression>
    <math:apply>
      <math:and/>
      <math:apply>
        <math:eq/>
```

```
    <math:csymbol>C</math:csymbol>
    <math:cn>true</math:cn>
   </math:apply>
   <math:apply>
    <math:eq/>
    <math:csymbol>L</math:csymbol>
    <math:cn>false</math:cn>
   </math:apply>
   <math:apply>
    <math:geq/>
    <math:apply>
     <math:times/>
     <math:csymbol>S.x</math:csymbol>
     <math:csymbol>S.y</math:csymbol>
     <math:csymbol>S.z</math:csymbol>
    </math:apply>
    <math:cn>0.32</math:cn>
   </math:apply>
  </math:apply>
 </mim:Expression>
 <mim:ExpressionArgument>
  <mim:ParameterValue typeID="AIdoor-II.Closed"
       variableBinding="C"/>
 </mim:ExpressionArgument>
 <mim:ExpressionArgument>
  <mim:ParameterValue typeID="AIdoor-II.Closed"
       variableBinding="L"/>
 </mim:ExpressionArgument>
 <mim:ExpressionArgument>
  <mim:ParameterValue typeID="V-II.SizeBoundingBox"
       variableBinding="S"/>
 </mim:ExpressionArgument>
</mim:Condition>
```

Listing 2: Excerpt of XML metadata of the 'open' interaction: Condition element

As mentioned in the previous section, the description of the MIM Action element can be composed of an arbitrary number of sub-actions. The interaction metadata of 'open' interaction uses such complex structure. It consists of descriptions of two parallel actions. First action is an animation of door opening. Second action is a sound that occurs when the door is opening. The animation action is described by a simple semantic description: "Door opens slowly to the sides". The action related with sound includes sound characteristics described by value of parameter taken from a custom Aural Interaction Interface. The value of this parameter is represented in the MPEG-7 AudioSpectrumBasisType. Listing below illustrates the XML representation of the Action element.

```
<mim:Action>
 <mim:SubAction>
  <mim:Parameter typeID="A-II.PlayedSound">
   <mim:Result
        xsi:type="mpeg7:AudioSpectrumBasisType"
        mpeg7:loEdge="62.5"
        mpeg7:hiEdge="8000"
        mpeg7:resolution="1/4 octave">
    <mpeg7:BasisFunctions>
     <mpeg7:Matrix dim="10 5">
        0.26 -0.05 0.01 -0.70 0.44
        0.34 0.09 0.21 -0.42 -0.05
        0.33 0.85 0.24 -0.05 -0.39
        0.33 0.85 0.24 -0.05 -0.39
        0.27 0.83 0.16 0.24 -0.04
        0.27 0.83 0.16 0.24 -0.04
```

```
        0.23 0.83 0.09 0.27 0.24
        0.20 0.83 0.04 0.22 0.40
        0.17 0.81 0.01 0.14 0.37
        0.33 -0.15 0.24 0.05 0.39
     </mpeg7:Matrix>
    </mpeg7:BasisFunctions>
   </mim:Result>
  </mim:Parameter>
 </mim:SubAction>
 <mim:SubAction>
  <mim:Semantics>
   <mpeg7:FreeTextAnnotation>
        Door opens slowly to the sides.
   </mpeg7:FreeTextAnnotation>
  </mim:Semantics>
 </mim:SubAction>
</mim:Action>
```

Listing 3: Excerpt of XML metadata of the 'open' interaction: Action element

Consider a repository of interactive 3D models of door manufactured by a certain company. Each object consists of a 3D geometry, embedded behavior, and MPEG-7 metadata with embedded MIM metadata. Such repository is accessible to architects designing new buildings. Architect specifies door properties that are required by his/her design and looks for doors that match these properties. A simple query could look for interactive 3D objects tagged with 'door' and 'automatic' keywords and reacts to a change of position of other objects. Such query can be formulated in the following way.

```
LET $doc := doc('http://www.kampin.com/doors.xml')
FOR
  $m IN $doc/Mpeg7/Description/MultimediaContent/Multimedia
WHERE (
    $m//KeywordAnnotation/Keyword = 'door'
    AND
    $m//KeywordAnnotation/Keyword = 'automatic'
) AND imql(
    $m//Interactions,
    event.paramtype.id == 'V-II.PositionCenter'
) RETURN $/MediaLocator/MediaURI
```

Listing 4: Sample XQuery query with embedded IMQL query

The architect can also specify requirements on the area of the motion sensor. For example a set of points may be specified that have to be included in the sensor area. Points have to be specified in the X3D SFVector3f format (X3D). For example: furthest point on the ground (0 0 -5) and at half of the door height (0 1 -3), and furthest points located to the sides of the door (-1 1 -1) and (1 1 -1). The previous query extended with such requirement is the following.

```
LET $doc := doc('http://www.kampin.com/doors.xml')
FOR
  $m IN $doc/Mpeg7/Description/MultimediaContent/Multimedia
WHERE (
    $m//KeywordAnnotation/Keyword = 'door'
```

```
        AND
        $m//KeywordAnnotation/Keyword = 'automatic'
) AND imql(
        $m//Interactions,
        event.paramtype.id == 'V-II.PositionCenter'
        AND '0 0 -5' >> event.space
        AND '0 1 -3' >> event.space
        AND '-1 1 -1' >> event.space
        AND '1 1 -1' >> event.space
) RETURN $/MediaLocator/MediaURI
```

Listing 5: Query presented in Listing 4 extended with
3D interaction space requirement

Moreover, also detailed action properties can be included in the query. Assuming that an architect wants to find all sliding doors with a specific opening sound, the query is the following.

```
LET $doc := doc('http://www.kampin.com/doors.xml')
FOR
  $m IN $doc/Mpeg7/Description/MultimediaContent/Multimedia
WHERE (
    $m//KeywordAnnotation/Keyword = 'door'
    AND
    $m//KeywordAnnotation/Keyword = 'sliding'
    AND
    $m//KeywordAnnotation/Keyword = 'automatic'
) AND imql(
    $m//Interactions,
    event.paramtype.id == 'V-II.PositionCenter'
    AND action.semantics = 'slide'
    AND
      action.interactive.paramtype.A-II.PlayedSound.after.value
      =
      '<value xsi:type="mpeg7:AudioSpectrumBasisType"
          mpeg7:loEdge="62.5"
          mpeg7:hiEdge="8000"
          mpeg7:resolution="1/4 octave">
        <mpeg7:BasisFunctions>
         <mpeg7:Matrix dim="5 5">
           0.26 -0.05 0.01 -0.70 0.44
           0.33 0.85 0.24 -0.05 -0.39
           0.27 0.83 0.16 0.24 -0.04
           0.23 0.83 0.09 0.27 0.24
           0.17 0.81 0.01 0.14 0.37
         </mpeg7:Matrix>
        </mpeg7:BasisFunctions>
      </value>'
) RETURN $/MediaLocator/MediaURI
```

Listing 6: Query presented in Listing 4 extended with
sub-action requirements

CONCLUSIONS

The three elements presented in the paper form a complete solution for describing object interaction characteristics and exploiting such descriptions. Although, the approach is particularly useful with architectural modeling, is not specific to any particular domain. With the use of Interaction Interface concept it can be easily employed for describing 3D representations of body parts used in medicine or describing some interactive objects in game development industry. With small modifications, namely definition of new interaction interfaces, the presented solution for building interaction metadata can be even more general. It can be used to describe any interactive entity, even if it is not a 3D object, for example a role in automatic negotiations or a software component.

The domain of interaction metadata for autonomous, reusable interactive 3D objects is in early stages of development. There are many areas open for research. First and most obvious area is research of search engines and comparison algorithms for interaction properties. Second is related with interaction properties descriptors. Some of such descriptors could be calculated directly from the object definition making the metadata creation process faster and more efficient. Moreover, different applications will require different interaction interfaces. Therefore the third area of potential research is related with development of universal interaction interfaces and tools for publishing and discovery of such interfaces. With such tools producers of sliding doors could post libraries of 3D representations of their products that are described with a domain specific interaction interfaces. Making these libraries searchable and allowing for CAD system extensions that connect with object libraries spread through the Internet and automatically find objects best suited for a given design, which could make designer life even easier.

REFERENCES

AAF, *AAF Object Specification v1.1*, 2005, Available: http://www.aafassociation.org/html/techinfo/index.html

Bilasco I.M., Gensel J., Villanova-Oliver M., Martin H., On indexing of 3D scenes using MPEG-7, in *Proceedings of ACM Multimedia 2005*, Singapore, 2005, pp: 471-474

Bilasco I.M., Gensel J., Villanova-Oliver M., Martin H., An MPEG-7 framework enhancing the reuse of 3D models, in *Proceedings of Web3D Symposium 2006*, Columbia, Maryland USA, 2006, pp: 65-74

Bry F. Patrânjan P., Reactivity on the web: paradigms and applications of the language XChange, in *Proceedings of the 2005 ACM Symposium on Applied Computing*, Santa Fe, New Mexico USA, 2005, pp: 1645-1649

Dayal U. et al., HiPAC: a Research Project in Active, Time-Constrained Database Management, *Interim Report, Technical Report XAIT-88-02*, Xerox Advanced Information Technology, June 1988

DCMES: *Dublin Core Metadata Element Set, Version 1.1*, 2008, Available: http://dublincore.org/documents/dces/

DIG35: Digital Imaging Group, *DIG35 Specification, Metadata for Digital Images 1.0*, 2000, Available: http://xml.coverpages.org/FU-Berlin-DIG35-v10-Sept00.pdf

EXIF: Japan Electronics and Information Technology Industries Association (JEITA) Standard, *JEITA CP-3451, Exchangeable image file format for digital still cameras: Exif Version 2.2*, 2002, Available: http://www.exif.org/Exif2-2.PDF

ID3, *ID3 Informal standard, ID3 tag version 2.4.0*, 2000, Available: http://www.id3.org/id3v2.4.0-frames

ISO/IEC 15938, ISO International Standard, *Information technology – Multimedia content description interface* Available: http://www.iso.org/iso/en/prods-services/popstds/mpeg.html

Lorenz B., Ohlbach H.J., Stoffel E.P., A Hybrid Spatial Model for Representing Indoor Environments, in *Proceedings of W2GIS 2006, (LNCS 4295)*, Hong Kong, China, 2006, pp: 102-112

Martinez J. M. et al., *MPEG-7 Overview*, 2004, Available: http://www.chiariglione.org/MPEG/standards/mpeg-7/mpeg-7.htm

MXF: *SMPTE 377M-2004 Television Material Exchange Format (MXF) Standard – File Format Specification*, 2004, Available: http://store.smpte.org/category-s/1.htm

Papamarkos G., Poulovassilis A., Wood PT., *Event-Condition-Action Rule Languages for the Semantic Web*, Workshop on Semantic Web and Databases, 2003

Pitarello F., de Faveri A., Semantic Description of 3D Environments: a Proposal Based on Web Standards, in *Proceedings of Web3D Symposium 2006*, Columbia, Maryland USA, 2006, pp: 85-95

P/Meta: *EBU Metadata Exchange Scheme (EBU P/Meta) 2.0, EBU Tech 3295-v2*, 2007, Available: http://www.ebu.ch/en/technical/metadata/specifications/index.php

Swartz A., *MusicBrainz: A Semantic Web Service*, 2001 Available: http://logicerror.com/musicbrainzArticle

Thome B., Gawlick D., Pratt M., Event processing with an oracle database, in *Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data*, Baltimore, Maryland USA, 2005, pp: 863-867

Walczak K., Beh-VR: Modeling Behavior of Dynamic Virtual Reality Contents, in Proceedings of the 12th International Conference on Virtual Systems and Multimedia VSMM 2006, in: H. Zha et al. (Eds.): *Interactive Technologies and Sociotechnical Systems, Lecture Notes in Computer Sciences 4270,* Springer Verlag Heidelberg 2006, pp. 40-51

X3D Specifications
Available: http://www.web3d.org/x3d/specifications/

Z39.87: NISO American National Standard, *ANSI/NISO Z39.87-2006, Data Dictionary – Technical Metadata for Digital Still Images*, 2001, Available: http://www.niso.org/kst/reports/standards/

Zhou X., Zhang S.,Cao J., Dai K., Event Condition Action Rule Based Intelligent Agent Architecture, *Journal of Shanghai Jiaotong University, Volume 38, Issue 1*, January 2004, pp: 14-17

JACEK CHMIELEWSKI was born in Poznan, Poland and went to the Poznan University of Technology, where he studied teleinformatics and obtained his degree in 2001. After the studies he joined the Department of Information Technology at the Poznan University of Economics. In 2008 he received PhD in the field of multimedia systems. His research interests include multimedia semantics and mobile multimedia. His e-mail address is: `Jacek.Chmielewski@ae.poznan.pl` and his Web-page can be found at `http://www.kti.ae.poznan.pl/jacek_chmiele wski.html`

Human Figure Animation: A Historical Perspective

Stephania Loizidou Himona
Frederick University
Cyprus

Yiorgos Chrysanthou
University of Cyprus
Cyprus

Andreas Loizides
PA College
Cyprus

Keywords

Computer Animation, Human Figure Animation, Simulation, Locomotion, Survey.

Abstract

Computer Animation, has advanced significantly as well as spectacularly over the last two-to-three decades. The interesting field has innovated tremendously moving from first-generation i.e. purely geometric models where motion was developed using kinematic techniques, to second-generation i.e. physical-based models (direct versus inverse dynamics and hybrid systems), towards the development of the third-generation i.e. autonomous behaviour through learning and perception. The simulation and animation of a variety of real-world objects appear with stunning realism. However, the main issue here is not on what we have achieved but rather, based on what we have achieved, what is to follow. The emerging results are expected immensely by researchers in the field, worldwide. The paper delivers a survey on human figure animation and in particular a classification framework of the work done on the control of locomotion.

Introduction

Computer Animation of human figure locomotion is becoming more and more popular in the recent years with virtual humans being used extensively in three-dimensional simulation environments. Such computer animation can be used in a wide range of applications, such as computer games, film industry, multimedia, training, education, medicine, virtual reality, architecture, security etc.

Although a number of different techniques (kinematics, dynamics, biomechanics, robotics, etc) have been used for producing realistic or believable motion, in many ways computer animation of human figures still remains a challenging problem. Walking as a motion is a very challenging control task due to its dynamically unstable nature, which requires continual balance to prevent the human character not only from falling over but also from reaching poses that are out of joint limits. In addition to that, people are very skilled at perceiving the subtle details of human motion and the viewer can easily recognise as well as reason when a motion appears artificial and unnatural.

An "efficient" human locomotion system should be able to model different types of motions and to account for different ways that a person could walk, not to mention different moods, conditions and other sociological as well as psychological states that affect the individual's gait style.

That is, the content (refers to the nature of the action in the movement) as well as the style (denotes the particular way that the action is performed) are the two major factors that should be taken into consideration when dealing with the perception of the human motion. These two factors should be in interaction with each other but at the same time in separation. For example, a system that can synthesise stylistic variations of a given action would be a useful tool. Such a system that applies user-specified styles to pre-recorded motion sequences (as well as extrapolating styles not included in the original data), for a small database, is proposed in (Torresani et al., 2007). Another system that learns to control planar walking gaits of a given style is analysed in (Sharon and van de Panne, 2005).

This paper outlines a comparative study of previous work done in the area of motion control in computer animation focusing on the different methods available together with the solution that each method assisted towards the development of human figure locomotion, i.e. the complex mechanism of human motion, according to the referred application. It aims to be used as a study guide, by covering the relevant literature as well as providing a classification framework. Once the current solutions are identified the paper explores areas of future developments.

Advances in Human Figure Animation

Human figure animation is an interesting as well as a challenging subset of Computer Graphics. It engaged people of different disciplines from the early days. Long before movie cameras were invented, Eadweard Muybridge's photographic sequences of animal (using a still camera he photographed a horse running to prove that when it gallops there are times when it has all four feet off the ground, (Muybridge, 1955)) as well as anthropomorphic locomotion became a point of reference for many researchers.

Studying the activities of human beings as well as their behaviour under different circumstances has become a focal point that has been addressed not only by computer scientists (the use of synthetic actors in 3D simulation environments, e.g. simulations of crowds and human behaviours in virtual reality applications), recently, but originally by psychologists, sociologists, medical practitioners, physicians and many others needless to mention the appli-

cations in the film industry. However, because there is not an easy way for people in one field to know what people in the other fields are working on and thus achieving, there is usually little or no interdisciplinary exchange of ideas and as a consequence results are slowly progressing.

Human Locomotion

Human locomotion is an important activity as well as an interesting problem closely related to man's everyday life. It is a complicated issue, since the articulations contained in the human figure have enormous complexity. The model has a large number of (rotary) joints each with several degrees of freedom (DOF) and the interaction between these joints is controlled by the muscles. Besides, there exist a large number of variations in the movement of individuals, so that exact specification and control of their movements is, in most cases, practically impossible.

The fact that human movement patterns are extremely familiar, (few movements are as familiar and recognizable as human walking and running (Hodgins, 1998)), and the viewer can easily recognize as well as reason when a motion appears artificial and unnatural, complicates the matter further. That is, people are skilled at detecting subtleties in human motion (Hodgins et al., 1998) which makes the animation of human figures an important, but at the same time, a difficult problem in computer animation.
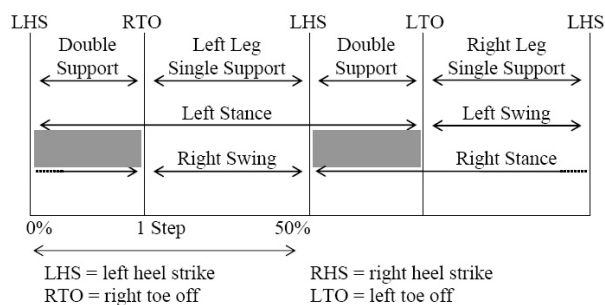
Figure 1: Locomotion cycle for bipedal walking

For simplification, walking is broken up into two phases; the stance and the swing phase, and these are further broken up into sub-phases. Figure 1, describes the locomotion cycle for bipedal walking from biomechanics (sets the start of the walking cycle on the left heel strike event) as described in (Loizidou, 1992). Biomechanics principles to represent walking motion on slopes and around curves has been used by Sun and Metaxas (2001).

Human Figure Animation - Classification Overview

An early classification (Terzopoulos, 1999) tried to distinguish between the different levels of computer animation, and in particular human figure animation, from its beginning towards its then state; geometric versus physically based models. It is believed that a further category has been created since then, namely, the beginning of autonomous behaviour through learning and perception.
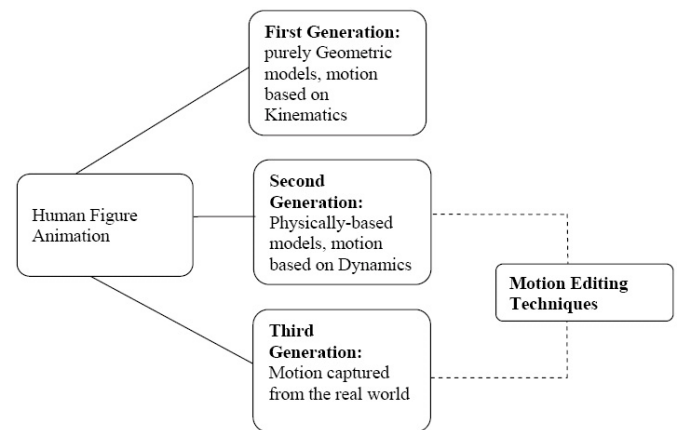
Figure 2: Categorization of human figure animation

Figure 2, illustrates this categorization. The next section describes these three levels, outlining the evolution that progressed from purely kinematic "knowledge-based" methods (1980's) towards approaches that incorporate dynamic simulations to generate motions (1990's) and even towards interactive blending and further tuning of synthetic or captured motion (1997 onwards); see also (Multon et al., 1999). The aim is not on emphasizing on the historical perspective of this classification but rather on describing the intrinsic advantages and/or limitations of each one of them.

First Generation

During the first-generation i.e. purely geometric models, motion was developed using kinematic techniques (direct and inverse kinematics, relied in biomechanical knowledge) which basically helped in keyframing (Zeltzer, 1983), (Lasseter, 1987), (Wilhelms, 1987), (Bruderlin and Calvert, 1996), (Hodgins, 1998), when trying to achieve expressiveness and believable behaviour. Keyframing is basically the technique whereby the animators have to explicitly specify the definition of the key values of the character's DOF at specific instances called the "key-frames". Interpolation of the values of the DOF is then used (e.g. cubic spline techniques) to compute the in-between frames (Perlin, 1995), (Kovar and Gleicher, 2003), (Safonova and Hodgins, 2005).

However, the end result, regardless of the fact that it was easy to produce (low computational cost), was not at all realistic and had nothing to do with its proximity to real life (Wilhelms, 1987). That is, the major drawback of the method concerned the lack of motion realism and therefore researchers had to combine it with appropriate normalization (Boulic et al., 1990), correction techniques to minimize sliding (Boulic, 2003) and foot constraints with inverse kinematics.

Second Generation

Moving towards the second-generation i.e. physical-based models versus inverse dynamics (Vasilonikolidakis and

Clapworthy, 1991) as well as hybrid systems (Loizidou and Clapworthy, 1993)), the motion was improved tremendously, however this was done at the expense of complicated physical laws of motion equations (e.g. Newton's laws, Lagrange mechanics, Featherstone's equations of motion etc) with quite a number of DOF even for a simple representation of e.g. the human body. To simulate human bodies is difficult due to the fact that the human motion is controlled by internal torques applied to the joints. The main problem here is how to specify forces and torques, which actually move the human figure in a way, which the animator might desire (Le et al., 2003).

Although the results have been realistic not much has been done about accounting for feelings, attitude, emotions and behaviour in the body animation. The motion has to be purely directed by the animator but then again the movement had to be predefined with no interesting variations from one animation sequence to the next. The main concern at that point was the realism of the movement with the minimum calculations possible as well as the fastest of the particular application.

Controlling the motion at a high level was basically the main problem. There was not an easy, convenient or even a possible way to specify the form of motion using flexible set of movement commands that could generate a variety of motions. High-level descriptions of commands like "walk happily" or "walk vigorously" where these commands would be translated automatically into the corresponding parametric descriptions of the motion was still under investigation.

To overcome some of the problems stated above, researchers have tried to combine methods from the first and second generations. As an indication, Wilhelms (Wilhelms, 1987) used a hybrid kinematics/dynamics approach where controlling functions, representing positions over time, had to be entered explicitly before starting the simulation, without knowing intuitively what forces or torques would be necessary to produce the desired motion. Girard (Girard, 1987), (Girard, 1991) and Girard & Maciejewski (Girard and Maciejewski, 1985), also used a hybrid kinematics/dynamics algorithm and therefore all the problems associated with the kinematics approach were again present when attempting to control articulated figures. A similar approach has been adopted by Bruderlin and Calvert (Bruderlin and Calvert, 1989) and is discussed by Calvert (Calvert, 1991). Isaacs and Cohen (Isaacs and Cohen, 1987) presented another combination of kinematics and inverse dynamics specifications.

Furthermore, Hahn (Hahn, 1988) presented a method that merged kinematics and dynamics for articulated rigid bodies. He used kinematics to control joint (internal) trajectories and dynamics to model the effects of limb motion (external analysis) and external forces and torques on the body as a whole.

As it can be observed there was no way that using any one of these techniques alone was possible to achieve the required results. Therefore, a combination of two or more of these techniques was necessary according to the specified application for which they had been developed for.

Third Generation

The conventional classification for the third generation involves autonomous behaviour through learning and perception. However, other researchers allow for some other categorization, with the allowance of an extra sub-level which could be introduced between the 2nd and 3rd generation, i.e. motion interpolation which is basically the combination of pre-existing motion data, e.g. through motion capture (Badler et al., 1993), (Hodgins and Wooten, 1995), (Thalmann and Musse, 2007) (which is basically considered as the first stage in the process of producing exploitable animation data (Menache, 2000)), or even motion by example (Arikan and Forsyth, 2002), (Kovar and Gleicher, 2003), (Lerner et al., 2007); so that new motions can be produced.

Motion capture uses magnetic systems or optical (vision-based sensors) systems to record the actions of a real human character in 3D. This data (usually from a database pool) is then used to animate the virtual character. However, it is interesting to note that keeping the size of the database manageable as well as determining when a database has sufficient variety of motions is still an open area requiring further work. The acquisition is not free of problems, magnetic systems suffer from noisy data, while optical systems have problems related to occlusion.

It is rarely the case that captured motions can be used straight away and be effective in every respect. The real and its corresponding mapped simulated character should have exactly the same number of joints, the same segment proportions and the same skinning for the animation to fulfil the plain re-use of the captured motions (the size and the quality of the database resulting from motion capture is the key to success; the database must be large enough so as good transitions can be found (Hodgins and Wooten, 1995) plus accountability of contacts with the environment). That is, it is interesting to look for genericity (any size and any proportions characters) of the versatile walk engine (Boulic et al., 2004). That is why motion editing techniques (blending and warping) need always be applied for modifying and varying captured motions (see Gleicher (Gleicher, 2001) for a survey; as well as Gleicher et al. (Kovar et al., 2002) for a technique on retrieving the best-fitting animation using "point clouds" to estimate the distance of postures).

The lack of flow between different animation clips is also equally important in order to avoid unnatural-looking frozen postures between motions. A system for real-time virtual human idle motion synthesis for the body is proposed (Egges et al., 2004). This is based on motion capture (i.e. existing clips). As a basis for the motion generation, a Principal Component Analysis (PCA) is used to determine dependencies between joints in the various postures. PCA also allows for easy blending and fitting of motions

Level	Method	Characteristics	Applications
1	Geometric models (Kinematic techniques)	- Low computational costs, O(n) - Lack of motion realism	Virtual Reality, Movie Production, Video Games (pre-scripted motion)
2	Physically-based models (Dynamic techniques)	- Realistic motion - High computational costs, $O(n^3)$	Virtual Reality, Art, Biomechanical Simulations
3	Motion Editing (Interpolation methods)	- Low computational costs, O(n) - Large volume of motion database - Realistic trajectories - Hard interactive control	Video Games, Movie Productions, Art, Virtual Reality
4	Autonomous behaviours (Learning and Perception)	- Automatic simulation code generation - High computational costs, $O(n^3)$ - Accounts for dynamic effects	Virtual Reality, Art, Movies, Behaviour simulation, Biomedical simulations

Table 1: Levels of human figure animation

resulting in flexible and realistic animations in real time. The whole approach permits the specification of different databases for different persons, which allows for a personalized aspect in the animations.

Motion warping and motion blending are the two major families of techniques for modifying the captured motion. Motion warping (takes well-known trajectories, e.g. keyframes and modifies them in order to change the motion, thus creating smooth transitions without discontinuity (Witkin and Popovic, 1995), (Torresani et al., 2007)) and motion blending (avoids the jerkiness in the transitions by using a database of captured motions and interpolation techniques in order to generate new motions) are almost always necessary for preserving the fine details of the motion (Witkin and Popovic, 1995), (Bruderlin and Williams, 1995). Furthermore, constraint-based motion editing techniques (Gleicher, 1998), (Lee and Shin, 1999), (Popović and Witkin, 1999), (Rose et al., 1996) are also necessary to avoid e.g. foot sliding.

Analysis

Table 1, summarizes the levels of the models used for computer animation, in particular for human figure animation, giving the advantages and the disadvantages of each together with the application areas that each one is best suitable for.

Each model has its own number of advantages and disadvantages. As an indication, Table 2 compares the main two animation methods that have prevailed in recent years (dynamic techniques versus motion capture methods). From Table 2 one can conclude that dynamic techniques offer more advantages compared to motion capture methods, however this does not restrict researchers to use the latter (growing popularity) due to the fact that it produces realistic results, one of the most (if not the most) valued characteristic (motion capture data is the most common technique in commercial animation because many of the subtle details of human motion are naturally present in the data (Lee et al., 2002)) regardless of its price. In addition, motion capture databases are gradually being built and made available to the wider audience. However, the problems associated with motion capture, see also Table 2, prevent it from being the ideal solution for all applications.

It is thus understood that using one model alone would not solve the specific problem and it is further believed that a combination of the different models from the different levels could solve particular problems and as a result enhance in the creation of autonomous virtual characters capable to react realistically to their environment. Combining, for example, motion capture and dynamic simulation is an emerging area under exploration, in order to retain the advantages of each while avoiding their disadvantages (Zordan and Hodgins, 2002). That is, information about human motion contained in motion capture data coupled with the physical realism provided by simulation, could provide controllable and reactive human motion (Arikan et al., 2005), (Yin et al., 2007), (Sok et al., 2007).

Although the peak of its usage was reported at SIGGRAPH 2007 (6 papers by (Sok et al., 2007), (Treuille et al., 2007), (Cooper et al., 2007), (Chai and Hodgins, 2007), (Safonova and Hodgins, 2007), (Yin et al., 2007)) the idea of combining more than one of the techniques for human figure animation is not new. It originates from the innovative idea of Zordan and Hodgins that use a forward dynamics and a tracking controller to follow human motion data i.e. dynamics with motion capture (Zordan and Hodgins, 1999). At that point, they did not report it as an area of broad potential; their work was basically motivated from the three previous, similar ideas of (i) (Bruderlin and Calvert, 1989) that use simple simulation and kinematics (biomechanical heuristics), (ii) (Witkin and Popovic, 1995) that use motion capture with interpolating joint angles (blending between two sequences of joint angles to create new joint trajectories), (iii) (Rose et al., 1996) that use inverse dynamics and torque minimisation (inverse dynamics between fixed motion sequences).

Furthemore, at that time motion capture equipment was of limited capabilities (limited availability and quality) and on top of that developing robust dynamic controllers that produce natural looking motion was a very hard problem.

Dynamic techniques (force-based simulation of movement)	Motion Capture techniques (pre-recorded movements)
1. Lack of naturalness of the capture movement data (simulating robots)	1. They capture also the naturalness of the movements (humans)
2. Variant movement data (always producing new data)	2. Invariant motion (monotone) sequences (repetitions/reuse); limited to real-world characters
3. High computing requirements (too expensive)	3. No computations to calculate the movement data (pre-recorded data)
4. No mapping needed. No retargeting (new data). Refinement is required	4. Perfect mapping/retargeting/adaptation is required (analyze/modify extract captured data: qualitative analysis)
5. Physically correct forces, not realistic; i.e. physical validity often sacrificed for performance	5. No changing of the fine structure of the original capture data (no forces taken into consideration)
6. The only stage required	6. It can only be used as the first stage
7. Flexible and Interactive motion control of complex mechanical systems. Real-time programming	7. No interactivity
8. Very complicated tasks: specification and solution of the equations of motion, dedicated control techniques (extensive mathematics and physics)	8. Provides motion data for all DOF at a very high level of detail
9. No other equipment necessary	9. Motion capture systems are very expensive
10. "Easy to use" animation system	10. Cables connection restricts actor's range of motion, therefore difficult to record certain movements
11. High degree of reusability, due to motions that can be generated automatically	11. Repetitions of capture sessions (until the desired motion is achieve) can be costly and labour intensive. Therefore editing & processing the motion capture data after it has been collected in order to make it re-usable

Table 2: Dynamic versus Motion Capture methods - (shading indicates an advantage)

Therefore, nobody could really predict then the scale of its today's (and possibly tomorrow's) usage not to mention the capabilities of such a combination.

The objective would be to capture the subtle details of realistic human-like motion while at the same time maintaining the physical realism of the movement. Ideally, virtual worlds with enhanced physics would bring greater realism and expression. The emphasis should be on bringing the real and virtual worlds closer together by bridging their associated strengths.

One such method, although still simple and at its infancy, is discussed by (Zordan et al., 2005), whereby it combines dynamic simulations together with a motion captured database. This system selects the appropriate clip from the motion capture repertoire (the search engine) and once an interaction has happened, computes new motion based on physical conditions (the joint-torque controller). Dynamics is thus used only for a short time burst (e.g. during the effects of a collision) and after that the character finds an available motion from the capture repository, therefore, minimising the computational bottleneck associated with dynamic techniques.

A more simple idea is to combine interpolation with motion capture (Abe et al., 2004), (Safonova and Hodgins, 2007). In such systems, the desired motion is represented as an interpolation of two time-scaled paths through a mo-

tion graph. That is, it retains the natural transitions of motion graphs and the ability to synthesize physically realistic variations provided by interpolation. However, the quality of the results depends largely on the size (Treuille et al., 2007) (only a few minutes (Safonova and Hodgins, 2007)) as well as the quality of the motion database used to construct the motion graph (the appropriateness of the database (Chai and Hodgins, 2007), whereby once again motion capture data is used as a motion prior while matching constraints are specified by the user), together with the "good", and/or automatic, connectivity of the motion graph (Ikemoto et al., 2007). Furthermore, the idea of evaluating motion graphs is outlined by Reitsma and Pollard (Reitsma and Pollard, 2004), in order to identify how much motion data should be included in the motion graph, in addition to assessing the character's capabilities while comparing motion datasets for the improvement of the motion graph.

Chiu et al. also use motion graphs as a means for re-using motion capture data by building transitions between motion segments and forming a tightly connected network of movement (Chiu et al., 2007).

Cooper et al. provide some other combinations of motion capture data (Cooper et al., 2007). They produce the character's motion by interpolating the motion capture and they proceed to use active learning for improving the quality and the responsiveness of the motion controller. The

active learning is used for identifying the regions of the control space that do not perform well, and thus improving them. However, the system focuses on a specific model for motion control with a limited number of tasks. Another interesting idea uses motion capture (a motion engine) together with a controller (a control policy) which selects appropriate sequences of clips to achieve some goal quickly and naturally (Treuille et al., 2007). Once again this technique requires large amounts of pre-captured motion data, to produce highly varied controllers.

Another interesting work is by Shapiro et al., whereby once again the combination of motion capture is reported (Shapiro et al., 2003). However, they claim that this is the first time were the combination of dynamic and kinematic control is done in a uniform fashion, thus producing complex motion and interaction for animated characters.

However, more recently research has progressed towards the development of the evolutionary autonomous behaviour through learning and perception. That is, the evolution is to create self animating characters that learn to react appropriately to their perceived environment, thus, reducing much of the workload from the animators. This coincides with the interesting field of artificial intelligence whereby characters are free to react like living creatures. For example, the interaction between a pedestrian and his/her environment is still a focal point of interest in this field.

Researchers are working towards the development of autonomous virtual humans in a proper environment model which can analyze the particularity of each situation/environment that they appear each time, make decisions as well as behave naturally accordingly (Devillers et al., 2002), (Pelechano et al., 2007). Progressing through different levels of abstraction, these models amalgamate proper modelling together with motion however the main issue is the inclusion of perception and behaviour (Shao and Terzopoulos, 2005).

It is well understood that behaviour animation has its origins strongly connected to crowd simulation, (Thalmann and Musse, 2007). Reynolds work on flocks of birds (large groups of entities called boids) and fish (fishes) could be considered as the first example in the field of behavioural animation whereby we could notice that group behaviour is resulting from the interaction between the individual behaviours of its group members (Reynolds, 1987). Although his work was based on three simple rules it led to the emergence of a much more complex flocking behaviour (Reynolds, 1999).

Similar, and even more complex, work by Tu and Terzopoulos created a realistically rich environment inhabited by artificial fishes (Tu and Terzopoulos, 1994). Simulations of large crowd structures and crowd behaviours, in real time, using three levels of autonomy were proposed by the work of Musse and Thalmann (Musse and Thalmann, 2001). Work on simulating crowds of virtual humans while focusing on the individual agents instead of the groups, where the behaviour model was composed of three

levels, was also proposed by Ulicny and Thalmann (Ulicny and Thalmann, 2002). Individual agents behaviour planning is additionally discussed in (Lau and Kuffner, 2005), (Chiu et al., 2007), whereby the automatic generation of realistic motions using a planning algorithm, that performs a global search in a finite-state machine (FSM) of behaviours of organized motion captured clips, is described (the motion capture technique will be discussed later in this section).

Furthermore, HiDAC (High-Density Autonomous Crowds) can be tuned to simulate realistically different types of crowds using psychological, physiological and geometrical rules combined with physical forces (Pelechano et al., 2007). These allow for individual heterogeneous behaviours while accounting for perception, reaction to static and dynamic objects, appropriate individual collision avoidance and collision response. Though, the way of controlling the properties and behaviours of individual agents in the crowd (instead of lower level parameters, i.e. expanding this personality model) remains still a challenge.

As a consequence, researchers in the area of computer graphics and animation of human figures should start thinking of possibilities which would begin from the very top level, and for overcoming specific problems of the certain level, to direct the model towards integrating lower levels. Thus, for example, motion editing techniques would very much be aided by the use of dynamic constraints which in turn would much be aided by kinematic correction models. It is strongly believed that such a combination of techniques would solve and overcome most, if not all, of the problems associated with each model alone. The possibilities are endless.

Conclusion

This paper classifies the techniques that have been used extensively during the last two-to-three decades, for the simulation of human locomotion. The aim has been to survey the different methods, specifying explicitly from where they have started towards where they are directed, that is, to provide a taxonomy of the recent techniques together with their solutions as well as chronological reasons. The relationships (comparing and contrasting) between a number of state-of-the-art methods for animating human-like figures is explored together with a unifying view of several techniques and their applications as well as to give insights to the unexplored regions of the animation world.

It is well understood that things are progressive; however a lot has still to be done before we are in the contented situation whereby we can look at something played by a computer (in particular human figures) and not to be apparent as to whether it is actually real or computer simulated; not to mention the possibility of controlling it in real time (i.e. enhancing interactively in terms of higher-level actions for obtaining adaptive behaviour).

Future Directions

Future work could focus on combining the modelled methods which would solve disadvantages associated with a particular method if used alone. Another possibility would be to extend the animation sequences, together with the animated agents that appear on individual frames, so as to create more interesting, complex scenes and therefore to manage the natural variations overtime for believable crowds. Furthermore, the need for automatically creating large number of believable gaits for a crowd simulation is also a close necessity.

Evaluation of human locomotion is another area of possible research. It could be done qualitatively (e.g. through real and simulated video images) as well as quantitatively (e.g. through simulated and biomechanical data). Although not of particular concern in this paper's contents, the resulting perception of the human motion is affected by the geometric model (Hodgins and Wooten, 1995). There exists an enormous variety of models and rendering styles and it therefore becomes essential to adopt a unifying set of models for presenting results and ease in comparing them, e.g. using statistical data. For this, several forms of evaluation to test the motion, e.g. video comparison, biomechanical data or online Turing test could be used.

REFERENCES

Abe, Y., Liu, C., and Popovic, Z. (2004). Momentum-based parameterization of dynamic character motion. *ACM SIGGRAPH, Eurographics Symposium on Computer Animation*, pages 173–182.

Arikan, O. and Forsyth, D. (2002). Interactive motion generation from examples. *ACM Transactions on Graphics*, 21(3):483–490.

Arikan, O., Forsyth, D., and O'Brien, J. (2005). Pushing people around. *Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 59–66.

Badler, N., Hollick, M., and Granieri, J. (1993). Real-Time Control of a Virtual Human Using Minimal Sensors. *Presence*, 2(1):82–86.

Boulic, R. (2003). Re-Appraising Procedural Motion Modeling: Illustration on a Walk Engine. *Eurographics Workshop on Virtual Environments, Zurich*.

Boulic, R., Magnenat-Thalmann, N., and Thalmann, D. (1990). A Global Human Walking Model with Real-Time Kinematic Personification. *The Visual Computer*, 6(6):344–358.

Boulic, R., Ulicny, B., and Thalmann, D. (2004). Versatile Walk Engine. *Journal of Game Development*, 1(1):29–50.

Bruderlin, A. and Calvert, T. (1989). Goal-Directed, Dynamic Animation of Human Walking. *Proceedings of the 16th Annual Conference on Computer Graphics and Interactive Techniques*, pages 233–242.

Bruderlin, A. and Calvert, T. (1996). Knowledge-Driven, Interactive Animation of Human Running. *Proceedings of the Conference on Graphics Interface*, pages 213–221.

Bruderlin, A. and Williams, L. (1995). Motion Signal Processing. *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques*, pages 97–104.

Calvert, T. (1991). Composition of Realistic Animation Sequences for Multiple Human Figures. *Morgan-Kaufmann Series In Computer Graphics And Geometric Modeling*, pages 35–50.

Chai, J. and Hodgins, J. (2007). Constrained-Based Optimisation Using a Statistical Dynamic Model. *International Conference on Computer Graphics and Interactive Techniques*.

Chiu, B., Zordan, V., and Wu, C. (2007). State-annotated motion graphs. *Proceedings of the 2007 ACM symposium on Virtual reality software and technology*, pages 73–76.

Cooper, S., Hertzmann, A., and Popović, Z. (2007). Active Learning for Real-Time Motion Controllers. *International Conference on Computer Graphics and Interactive Techniques*.

Devillers, F., Donikian, S., Lamarche, F., and Taille, J. (2002). A programming environment for behavioural animation. *The Journal of Visualization and Computer Animation*, 13(5):263–274.

Egges, A., Molet, T., and Magnenat-Thalmann, N. (2004). Personalized Real-Time Idle Motion Synthesis. *Computer Graphics and Applications, 12th Pacific Conference*, pages 121–130.

Girard, M. (1987). Interactive Design of 3D Computer-Animated Legged Animal Motion. *IEEE Computer Graphics and Applications*, 7(6):39–51.

Girard, M. (1991). Constrained Optimization of Articulated Animal Movement in Computer Animation. *Making Them Move: Mechanics, Control, and Animation of Articulated Figures*.

Girard, M. and Maciejewski, A. (1985). Computational Modeling for the Computer Animation of Legged Figures. *Proceedings of ACM SIGGRAPH*, pages 263–270.

Gleicher, M. (1997). Motion Editing with Spacetime Constraints. *Proceedings of the 1997 Symposium on Interactive 3D Graphics*, pages 139–148.

Gleicher, M. (1998). Retargetting Motion to New Characters. *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*, pages 33–42.

Gleicher, M. (2001). Comparing Constraint-Based Motion Editing Methods. *Graphical Models*, 6(2):107–134.

Hahn, J. (1988). Realistic Animation of Rigid Bodies. *Proceedings of the 15th Annual Conference on Computer Graphics and Interactive Techniques*, pages 299–308.

Hodgins, J. (1998). Animating Human Motion. *Scientific American*, 278(3):46–51.

Hodgins, J., O'Brien, J., and Tumblin, J. (1998). Perception of Human Motion with Different Geometric Models. *IEEE Transactions on Visualization and Computer Graphics*, 4(4):307–316.

Hodgins, J. and Wooten, W. (1995). Animating Human Athletes. *ACM SIGGRAPH*, 95:71–78.

Ikemoto, L., Arikan, O., and Forsyth, D. (2007). Quick Transitions with Cached Multi-Way Blends. *Proceedings of the 2007 Symposium on Interactive 3D Graphics and Games*, pages 145–151.

Isaacs, P. and Cohen, M. (1987). Controlling Dynamic Simulation with Kinematic Constraints. *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques*, pages 215–224.

Kovar, L. and Gleicher, M. (2003). Flexible Automatic Motion Blending with Registration Curves. *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 214–224.

Kovar, L., Gleicher, M., and Pighin, F. (2002). Motion Graphs. *Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques*, 21(3):473–482.

Lasseter, J. (1987). Principles of Traditional Animation Applied to 3D Computer Animation. *ACM SIGGRAPH Computer Graphics*, 21(4):35–44.

Lau, M. and Kuffner, J. (2005). Behavior Planning for Character Animation. *Proceedings of the 2005 ACM SIGGRAPH / Eurographics Symposium on Computer Animation*, pages 271–280.

Le, D., Boulic, R., and Thalmann, D. (2003). Integrating Age Attributes to Virtual Human Locomotion. *International Archieves of Photogrammy, Remove Sensing and Spatial Information Science*, XXXIV-5.

Lee, J., Chai, J., Reitsma, P., Hodgins, J., and Pollard, N. (2002). Interactive Control of Avatars Animated with Human Motion Data. *Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques*, 21(3):491–500.

Lee, J. and Shin, S. (1999). A Hierarchical Approach to Interactive Motion Editing for Human-Like Figures. *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, pages 39–48.

Lerner, A., Chrysanthou, Y., and Lischinski, D. (2007). Crowds by Example. *Computer Graphics Forum*, 26(3):655–664.

Loizidou, S. (1992). *Dynamic Analysis of Anthropomorphic Manipulators in Computer Animation*. University of North London.

Loizidou, S. and Clapworthy, G. (1993). Legged Locomotion Using HIDDS. *Models and Techniques in Computer Animation*, pages 257–269.

Menache, A. (2000). *Understanding Motion Capture for Computer Animation and Video Games*. Morgan Kaufmann.

Multon, F., France, L., Cani-Gascuel, M., and Debunne, G. (1999). Computer Animation of Human Walking: A Survey. *The Journal of Visualization and Computer Animation*, 10(1):39–54.

Musse, S. and Thalmann, D. (2001). Hierarchical Model for Real Time Simulation of Virtual Human Crowds. *IEEE Transactions on Visualization and Computer Graphics*, 7(2):152–164.

Muybridge, E. (1955). *The Human Figure in Motion*. Dover Publications.

Pelechano, N., Allbeck, J., and Badler, N. (2007). Controlling Individual Agents in High-Density Crowd Simulation. *Proceedings of the 2007 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 99–108.

Perlin, K. (1995). Real time responsive animation with personality. *IEEE Transactions on Visualization and Computer Graphics*, 1(1):5–15.

Popović, Z. and Witkin, A. (1999). Physically based motion transformation. *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 11–20.

Reitsma, P. and Pollard, N. (2004). Evaluating Motion Graphs for Character Navigation. *Proceedings of the 2004 ACM SIGGRAPH / Eurographics Symposium on Computer Animation*, pages 89–98.

Reynolds, C. (1987). Flocks, Herds and Schools: A Distributed Behavioral Model. *ACM SIGGRAPH Computer Graphics*, 21(4):25–34.

Reynolds, C. (1999). Steering Behaviors for Autonomous Characters. *Game Developers Conference*, 1999:763–782.

Rose, C., Guenter, B., Bodenheimer, B., and Cohen, M. (1996). Efficient generation of motion transitions using spacetime constraints. *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 147–154.

Safonova, A. and Hodgins, J. (2005). Analyzing the physical correctness of interpolated human motion. *Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Com-*

puter animation, pages 171–180.

Safonova, A. and Hodgins, J. (2007). Construction and Optimal Search of Interpolated Motion Graphs. *International Conference on Computer Graphics and Interactive Techniques*.

Shao, W. and Terzopoulos, D. (2005). Autonomous Pedestrians. *Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 19–28.

Shapiro, A., Pighin, F., and Faloutsos, P. (2003). Hybrid control for interactive character animation. *Computer Graphics and Applications, 2003. Proceedings. 11th Pacific Conference on*, pages 455–461.

Sharon, D. and van de Panne, M. (2005). Synthesis of Controllers for Stylized Planar Bipedal Walking. *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2387–2392.

Sok, K., Kim, M., and Lee, J. (2007). Simulating Biped Behaviors from Human Motion Data. *International Conference on Computer Graphics and Interactive Techniques*.

Sun, H. and Metaxas, D. (2001). Automating Gait Animation. *Proceedings of ACM SIGGRAPH 2001*, pages 261–270.

Terzopoulos, D. (1999). Artificial Life for Computer Graphics. *Communications of the ACM*, 42(8):32–42.

Thalmann, D. and Musse, S. (2007). *Crowd Simulation*. Springer, 1st edition.

Torresani, L., Hackney, P., and Bregler, C. (2007). Learning Motion Style Synthesis from Perceptual Observations. *Proc. NIPS*, 19.

Treuille, A., Lee, Y., and Popović, Z. (2007). Near-Optimal Character Animation with Continuous Control. *International Conference on Computer Graphics and Interactive Techniques*.

Tu, X. and Terzopoulos, D. (1994). Artificial Fishes: Physics, Locomotion, Perception, Behavior. *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques*, pages 43–50.

Ulicny, B. and Thalmann, D. (2002). Towards Interactive Real-Time Crowd Behavior Simulation. *Computer Graphics Forum*, 21(4):767–775.

Vasilonikolidakis, N. and Clapworthy, G. (1991). Design of Realistic Gaits for the Purpose of Animation. *Computer Animation*, 91:101–114.

Wilhelms, J. (1987). Using dynamic analysis for realistic animation of articulated bodies. *IEEE Computer Graphics and Applications*, 7(6):12–27.

Witkin, A. and Popovic, Z. (1995). Motion Warping. *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques*, pages 105–108.

Yin, K., Loken, K., and van de Panne, M. (2007). SIMBICON: Simple Biped Locomotion Control. *International Conference on Computer Graphics and Interactive Techniques*.

Zeltzer, D. (1983). Knowledge-Based Animation. ACM SIGGRAPH SIG ART. *Workshop on Motion*, pages 187–192.

Zordan, V. and Hodgins, J. (1999). Tracking and modifying upper-body human motion data with dynamic simulation. *Computer Animation and Simulation99*, 2:280–285.

Zordan, V. and Hodgins, J.K. (2002). Motion Capture-Driven Simulations that Hit and React. *In Proceedings of the 2002 ACM SIGGRAPH / Eurographics Symposium on Computer Animation*, pages 89–96. ACM Press.

Zordan, V., Majkowska, A., Chiu, B., and Fast, M. (2005). Dynamic response for motion capture animation. *ACM Transactions on Graphics*, 24(3):697–701.

Student Paper
in
Modelling and Simulation

THE ROAD MAP TO APPLY EVOLUTIONARY INTELLIGENCE TO ASPHALT PAVEMENT MODELLING

Reus Salini (1)
José Neves (2)
António Abelha (3)

Universidade do Minho
Campus de Gualtar
4710 - 057 – Braga – Portugal
E-mail (1): R**.**Salini@gmx.net
E-mail (2): jneves@di.uminho.pt
E-mail (3): abelha@di.uminho.pt

KEYWORDS

Evolutionary Intelligence, Asphalt Pavement Modelling.

ABSTRACT

Modelling the service life for asphalt pavements is widely made with base on empirical methods, developed half-century ago, with poor results. The new reality of the XXI century, with high construction costs, environmental restrictions and growing volumes of vehicles in the highways, enforce the shift to a new level of quality and accuracy to predict the service life of the pavements. This paper presents some insights either theoretical or experimental into a making from the ground, of an approach to predict and model the asphalt pavement behaviour using soft computing tools and, at the same time, create a way to accumulate the knowledge in this engineering field. The knowledge about the asphalt pavement life cycle is organized in a hierarchical way in order to be reused in a formal way, leading to an evolutionary process of adaptation and construction (Neves et al. 2007).

INTRODUCTION

Created in the old Roman Empire, the paved roads are at the center of a big revolution in the human history. The pavement allows any person, day or night, with raining or snow, to move between two locations in an effective way, helping to grow all human activities, as the commerce, the industry, the health sciences, just to name a few.

Thousands years after the Via Áppia was paved by the Romans, the most used methods for pavement modelling and design remain at an unbelievable empirical level. The studies based on fatigue tests made in the last half-century, after the introduction of the asphalt as material for highways, do not allow any true progress to understanding and modelling the asphalt pavement behaviour.

This empirism results in an early and high rate of failure in some pavements and, in anothers, over-design; both cases have a poor cost-effectiveness for the used money. The empiric approaches also do not provide any way to scientifically accumulate the knowledge, making hard for the pavement engineering use the experience of one highway in to another as the experience remain empirically accumulated in the persons, and can be easily lost due to forgetting, retirement or death.

The new challenges of the XXI century, with the growing quantity of the vehicles in the streets, growing capacity of load of the trucks, high costs of the materials used in pavements, high cost of the labour and the crescent environment restrictions to mining and the use of materials like stones, enforce the achievement of a new level in prediction of the service life and modelling the asphalt pavement behaviour (Flintsch 2003).

Due the ultra-complex behaviour of the asphalt pavement, quality modelling cannot be achieved with empirical or *semi-religious* approaches. A strong tool, able to handling all the necessary variables and its near infinite combinations need to be used. The soft computing approach to modelling may be in line to provide the necessary power to this task (Yang et al. 2003; Bosurgi and Trifirò 2005).

THE SOURCE DATA

In order to be able to use and reuse the knowledge available is necessary, first, to generate and organize the information about the performance of the pavement.

The initial step, and the first challenge, is in the instrumentation of a section of highway open to live-traffic, with sensors to register in a database and/or knowledge base all variable data believed as important and connected to the durability of the asphalt pavement. As its service life do not decrease in a linear or constant rate, the database need to register the information for one full cycle of the service life of the asphalt pavement, or in another words, to get data all along several years.

The desirable variables to construct the database can be grouped in three areas: (a) environmental, (b) vehicles and (c) structural response of the pavement.

The asphalt pavement is subject of a well known influence of the environment, where the desirable variables are: air temperature, rainfall, wind speed, groundwater level, solar radiation, UV radiation and the pavement temperature.

The pavement is a structure subject to loads by vehicles, and the desirable variables in this arena are: vehicle counting, speed, classification and configuration by axle, and weight by axle.

The asphalt pavement has, in general terms, an asphalt concrete layer and a granular layer. To know how the pavement structure is performing under loads and with the influence of the environment, the desirable data are: asphalt layer vertical strains, asphalt layer pressures, asphalt layer horizontal strains, moisture in the granular layer, granular layer pressures and strains.

The Figure 1 shows the desirable variables in a chart format. Every time a vehicle crosses the instrumented section all sensors are read and the generated data is recorded in the database. The recording is only stopped when the pavement meets the end of the service life.

The criteria to the end of the service life can be anyone, technical, financial, functional or of any other nature. Also a multiple criteria can be used with this method and, in this case, the data need be recorder until the last criteria is meeting. For each criteria the database will be "cuted" in a different point and a new matrix of data is created.

THE CONSTRUCTION OF THE KNOWLEDGE

With the database already built, the next step is to compile the information available to another language, a math language (Figure 2). This is a fundamental step in order to make easy handling of the data to fix the importance of each variable and its interaction to the pavement durability.

The data need be organized in a form of a matrix, with one line per vehicle.

Now, one has to calculate the relative relevance of each variable and the relative importance of each vehicle for the pavement service life in a consistent way, which is the second challenge in this method. This can be solved using one or more topologies based on Evolutionary Intelligence like evolutive-genetic algorithms, logic-mathematic functions or particle-swarm optimization and, if the translation of the field data to the math is made in a consistent way, a matrix of

equations will be generated, which may be solved with traditional approaches. A combination of Evolutionary Intelligence and matrix of equations can also be used to find the best possible classification for the lines (Mendes et al. 2004; Rocha et al. 2005; Rocha et al. 2006).

Once this task have been solved, every line in the matrix of data receives a "pavement damage" index representing the relative importance of the line in relation to the other ones (Figure 3).

It must be pointed out that the importance of the individual variables in the line is useful in order to testify the consistence of the solution of the matrix, but it is not critical information for this method.

The Figure 4 shows the diagram with the tasks to transform the source "brute" data to an information fine to train a neural network system.

The knowledge can be easily accumulated just adding new sets of field data to the initial database. The accumulation allows continuous improvement in the accuracy and usability.

HANDLING THE ERRORS

Regardless the used tools – matrix of equations and/or Evolutionary Intelligence based tools and/or computational topologies – some level of error will remain in the final pavement damage index. The third challenge is to handle this error in an efficient way.

The data obtained in the previous tasks that lead to the pavement damage index, is to be understood as the result of the "true" damage multiplied by an error (Figure 5).

At this stage of the development of the method it is not possible to know if there exist some way to calculate and isolate this error or not. In any case, an excellent solution to handling it was found.

The error can have any size if it is well distributed inside the matrix, or in another words, if the error in the pavement damage index in the line 1 is similar to the error in the line 2, and so on. As the result, the sum of the pavement damage indexes for the full matrix will have the same similar error.

Instead to use the pavement damage index as an absolute value, in this approach it is used in a comparative way, with the individual pavement damage index divided by the sum of the damages; as the error exists in both, the error is divided by itself and eliminated (Figure 6).

Handling the error in this way it helps the calculus of the pavement damage indexes, once it allows the use of

some tips, as mix artificial data with true field data, in order to potentiate an easy convergence and a better solution for the matrix. The only rule to be accommodated is: the error need to be well distributed inside the matrix.

TRAINING THE NEURAL NETWORK

The matrix of knowledge made in a consistent way and ready to be used is named "matrix of performance", a source of "intelligence" to train a neural network system.

The only challenge in this step is the computer power to train a so big matrix of data, with dozen of millions of data items.

On the other hand the neural network has to be trained until one is be able to do predictions with a realistic accuracy. This can be easily checked just proposing lines of the matrix of performance as problems to be solved. The neural network will be well trained if found a pavement damage index similar to the one in the matrix of performance for the same lines.

The Figure 7 shows a chart with the matrix of performance and a pavement design proposed as a problem to the trained neural network evaluates the pavement damage.

VALIDITY AND USE OF THE MODEL

The matrix of performance can be used to evaluate the damage to pavements constructed with the same materials used in the instrumented test section used as a data source to the initial database.

The Figure 8 shows how to use the trained neural network to do predictions about the pavement service life. The problem need be proposed in the exact same format used to construct the matrix of performance and with the same information, including data about the environment, vehicles and pavement structural response. The data about the environment can be easily found with meteorological institutes and the vehicles can be counted, for existing highways, or evaluated with traditional procedures, for new highways. For new pavements the structural response can be calculated with finite elements procedures; for existing highways can be calculated with a combination of non-destructive tests (FWD, Benkelman Beam, etc.) and finite elements.

The Figure 9 shows a comparison between the approach here proposed and the equation of fatigue, the current state-of-the-art.

CONCLUSIONS

It is shown in Figure 9 that the (state-of-the-art) technology used in asphalt pavement modelling and design is based on the "equation of fatigue", an empirical-archaic approach that does not meet the requirements of the XXI century needs.

The approach proposed herein is based on soft computing tools leading, to:

- The establishment, for the first time in the pavement engineering, of a viable way to use strictly scientific procedures to model and predict the asphalt pavement behaviour and service-life, taking into consideration all problem variables;

- A change of paradigm with the potential to provide asphalt pavement modelling with unprecedented level of accuracy and reliability, with no subjective or empirical influence;

- The creation of an efficient and suitable form to knowledge gathering, allowing its use and reuse to optimize the accuracy and expand the usability of the model in a continuous way.

REFERENCES

Bosurgi, G. and Trifirò, F. *A Model Based on Artificial Neural Networks and Genetic Algorithms for Pavement Maintenance Management*. International Journal of Pavement, Vol. 6, No. 3, pp. 201–209, 2005.

Flintsch, G.W. *Soft Computing Applications in Pavement and Infrastructure Management: State-of-the-Art*. Transportation Research Board Annual Meeting, 2003.

Mendes, R.; Kennedy, J. and Neves, J. *The Fully Informed Particle Swarm: Simpler, Maybe Better*, Journal of the IEEE Transactions of Evolutionary Computation, 2004.

Neves, J.; Machado, J.; Analide, C.; Abelha, A. and Brito, L. *The Halt Condition in Genetic Programming*. In Lecture Notes in Artificial Intelligence 4874 – Progress in Artificial Intelligence – Springer, 2007.

Rocha, M.; Cortez, P., and Neves, J. *Simultaneous Evolution of Neural Network Topologies and Weights for Classification and Regression.* In Lecture Notes in Computer Science 3512, Proceedings of the IWANN Conference, Barcelona, Springer, 2005.

Rocha, M.; Cortez, P. and Neves, J. *Evolution of Neural Networks for Classification and Regression.* Neurocomputing, Elsevier, 2006.

Yang, J.; Lu, J. J.; Gunaratne, M. and Xiang, Q. *Overall Pavement Condition Forecasting Using Neural Networks: Application to Florida Highway Network*. 82nd Annual Meeting of the Transportation Research Board, Washington, D.C., 2003.

AUTHOR BIOGRAPHIES

REUS SALINI, Gaúcho born at the Rio-Grandense Republic. Has received the award for first place in the Civil Engineering course at Santa Catarina State University (Brazil); obtained his Master of Science course in Highway Engineering at the Federal University of Santa Catarina (Brazil). Works as pavement and highway designer engineer since 1995 with strong focus on innovative solutions. His e-mail is R.Salini@gmx.net and Skype username is "sukhoi47".

JOSÉ NEVES was born in Vila do Conde, Portugal, and went to the University of Coimbra, Portugal, where he studied chemical technology and obtained his degree in 1976. He worked for a couple of years at the University of Minho, Braga, Portugal, and then went to Heriot-Watt University, in Scotland, where he got a MSc degree in Software Development and Analysis, and a PhD degree in Logic Programming and Data Bases. He is now full Professor at the University of Minho, Braga, Portugal, where he leads the Artificial Intelligence Group. His main interests are in the area of Knowledge Representation and Reasoning, Evolutionary Intelligence and Logic Programming.

ANTÓNIO ABELHA was born in Vila do Conde, Portugal, and went to the University of Minho, Portugal, in 1982, where he studied Informatics. He worked from 1993 at the University of Minho, Braga, Portugal, where he got a MSc degree in Computer Science, and a PhD degree in Artificial Intelligence. He is now Auxiliary Professor at the University of Minho, Braga, Portugal, in the Artificial Intelligence Group. His main interests are in the area of Knowledge Representation and Reasoning, Logic Programming and Data Bases, and Simulation.
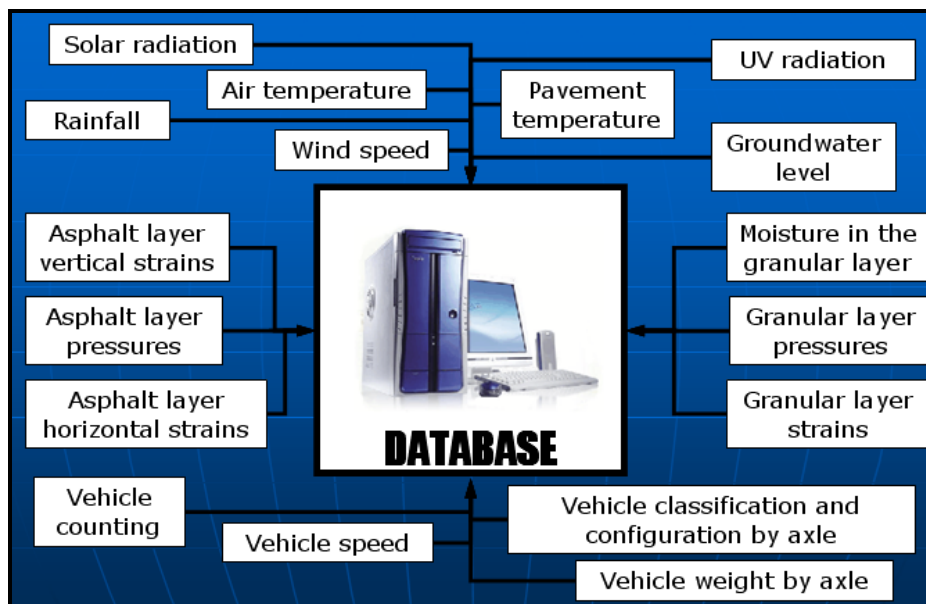
Figure 1: Desirable Variables to a Database Suitable for
Asphalt Pavement Modelling Using Artificial Intelligence

Figure 2: Field Data Organized in Form of a Matrix of Numbers
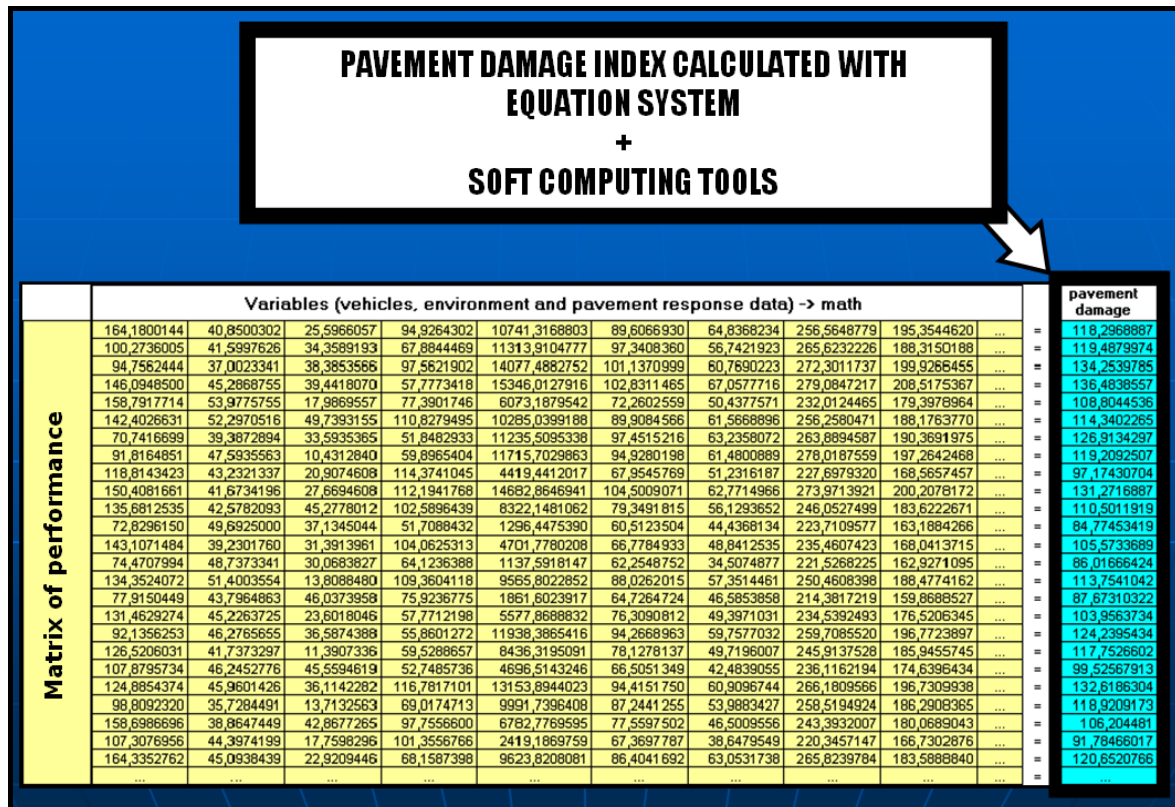
Figure 3: The Pavement Damage Index Calculated
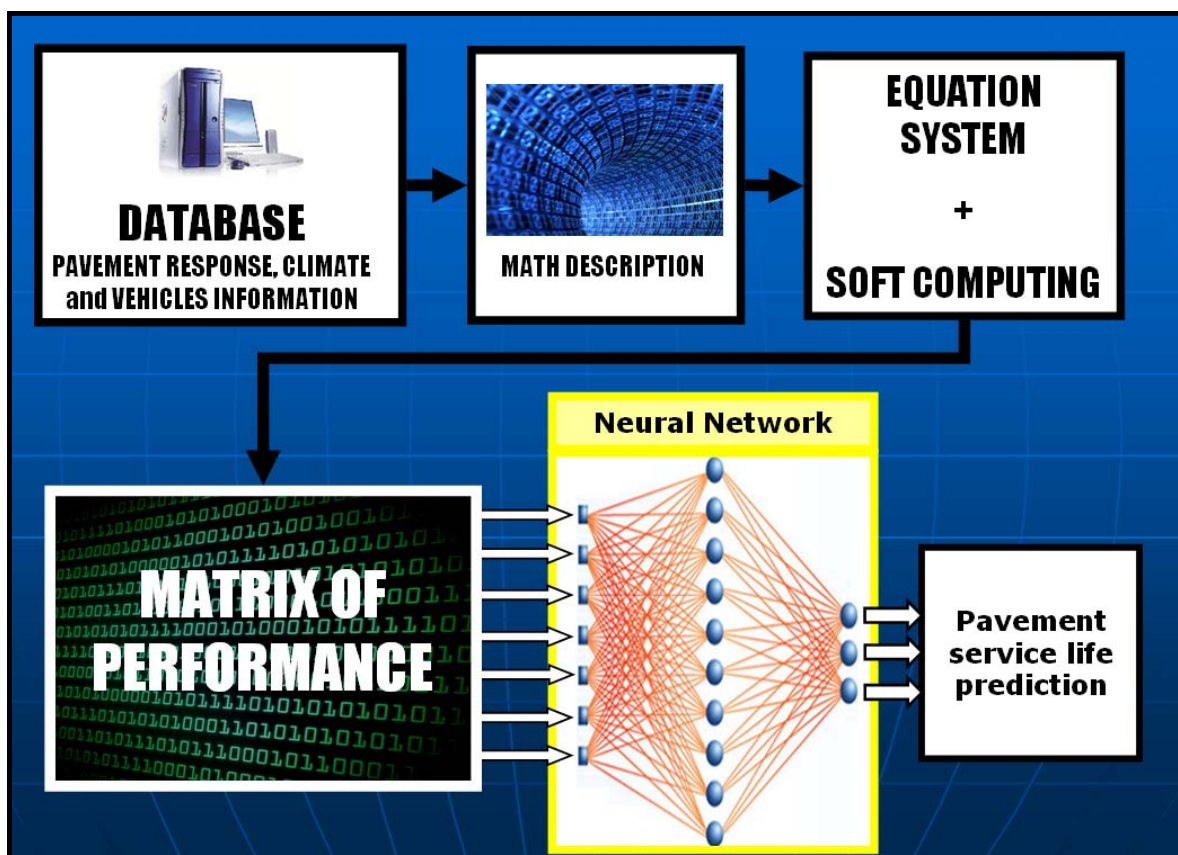
Figure 4: The Database With the History of the Pavement is "Handled"
to Become a Matrix of Performance to Train the Neural Network System

Figure 5: The Found Pavement Damage Index has an Unknown Error

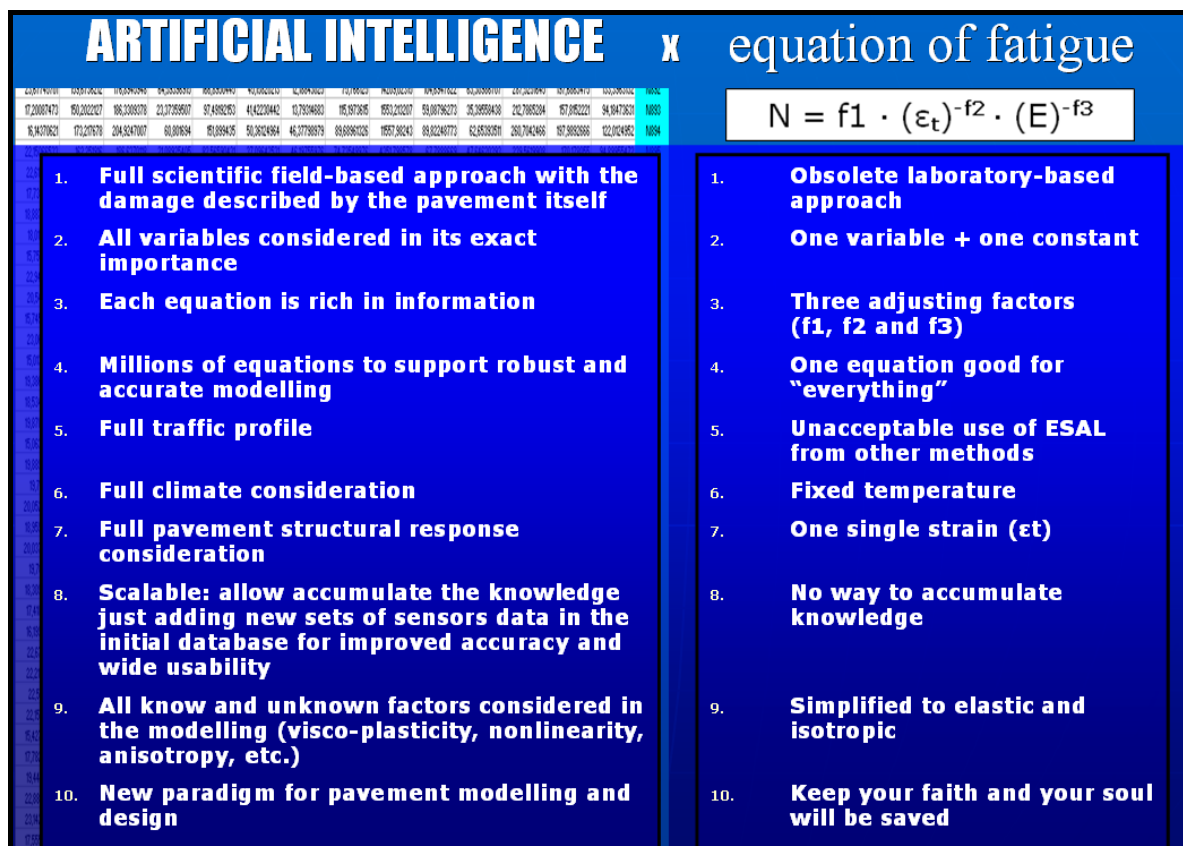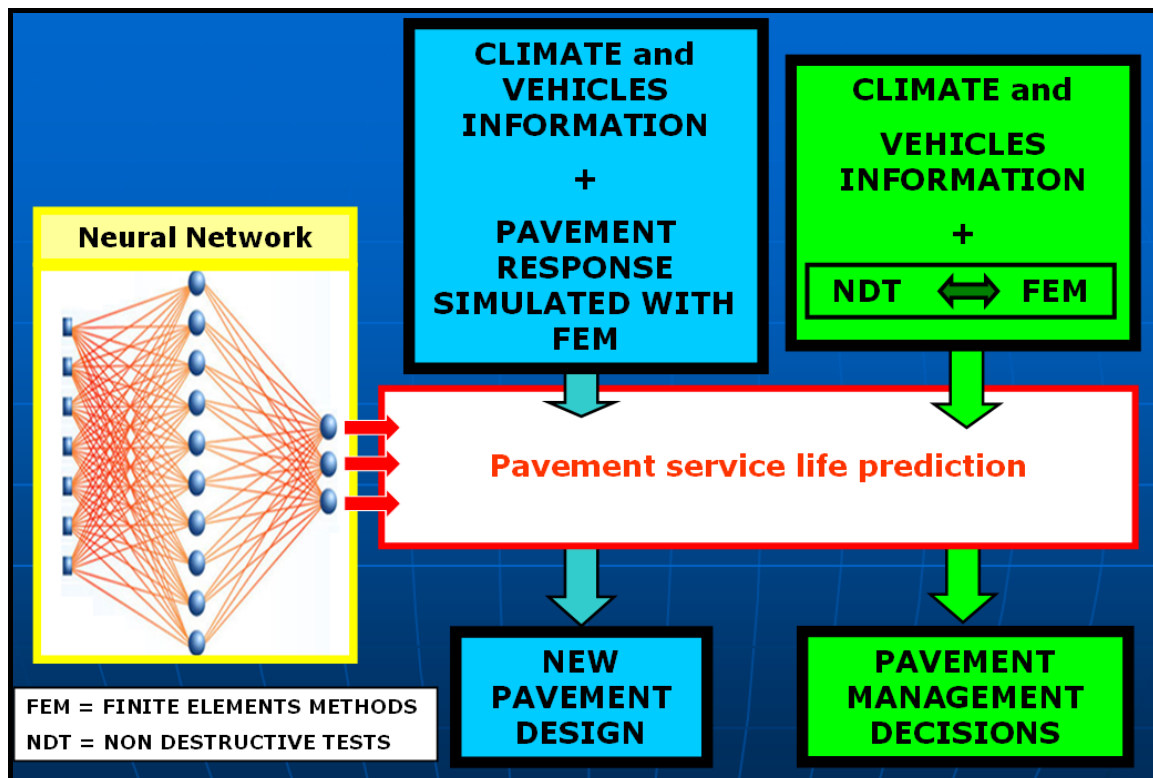Figure 6: The Error is Divided by Itself and Eliminated
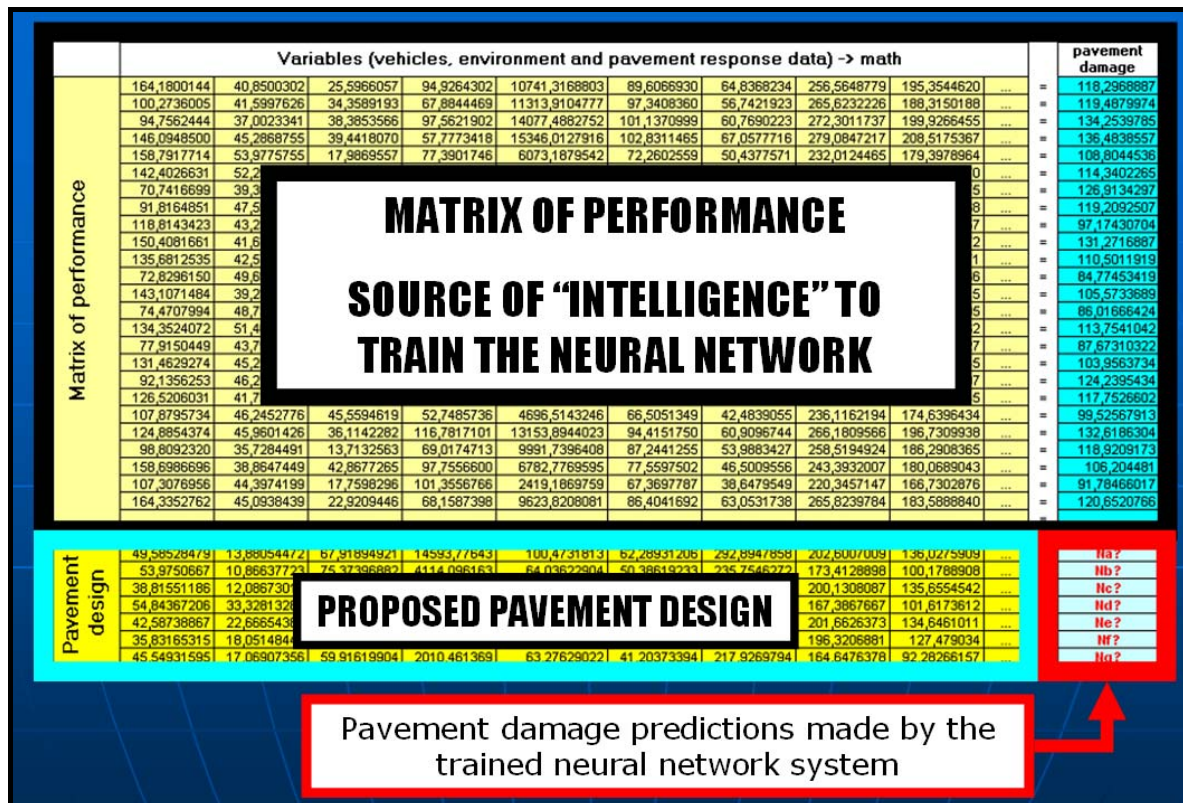
Figure 7: The Neural Network Will be Able to Do Predictions for Problems Proposed in the Exact Same Way Was Described the Initial Database Used to Generated the Matrix of Performance

Figure 8: The Trained Neural Network System Will do Predictions About the Service Life for Both, Design New Pavements and Take Decisions in a Pavement Management System

Figure 9: Comparison Between the Modelling Using Neural Network Trained With the Matrix of Performance Against the Current "Technology" of The Equation of Fatigue

AUTHOR INDEX

238 Karpenko, Simon M.	131 Nahodil, Pavel
90 Kindler, Eugene	441 Neumann, Gaby
387 Klimov, Ruslan	573 Neves, José
157 Kmet, Tibor	234, 238 Nikolaev, Dmitry P.
410 Ko, Min.S	238 Nikolaev, Ilia P.
299 Koch, Roman	238 Nikolayev, Petr P.
131 Kohout, Karel	118 Novais, A. Q.
139, 403 Kononov, Dmitry A.	519 Oplatková, Zuzana
471 Koutsoukos,Xenofon	299 Osterloh, Christoph
375 Kožusznik, Jan	37, 345 Otamendi, Javier
90 Krivy, Ivan	71 Papantoniou, Agisilaos N.
414 Kuehn, Wolfgang	67 Papastamatiou, Nikolaos P.
139, 403 Kul'ba, Vladimir V.	71 Papatzelou, Katerina
370 Kulka, Anna	164 Papavassilopoulos, G. P.
423 Kunz, Christian	410 Park, Sang C.
363 Labrique, Francis	37 Pastor, José Manuel
247 Lang, Teresa	399 Patlins, Pavels
31 Lee, Duckwoong	270 Piera, Miquel Àngel
261 Legato, Pasquale	299 Pionteck, Thilo
53 Legato, Pasquale	277 Poggi, Simonluca
285 Léonard, Joël	104 Pontier, Matthijs
207 Levchenkov, Anatoly	37 Poyatos, Miguel
322 Liehr, Andreas W.	285 Prévost, Stéphanie
562 Loizides, Andreas	519 Prokopová, Zdenka
118 Lopes, Fernando	548 Psonis, Panagiotis
97 Lotzmann, Ulf	227 Qeli, Ermir
67 Loumos, Vassili	478 Rahmani, Ahmed
548 Loupos, Konstantinos	541 Raja, Vinesh
285 Lucas, Laurent	188 Rekdalsbakken, Webjørn
370 Lyczek, Michal	60 Řezáč, Martin
512 Macků, Lubomír	313 Romero, Gregorio
299 Maehle, Erik	471 Roychoudhury, Indranil
25 Majeed, Basim	145 Rucins, Adolfs
457 Marquis-Favre, Wilfrid	345 Sainz, Jorge
313 Martínez, M. Luisa	573 Salini, Reus
375 Martinovič, Jan	525 Samek, David
387 Merkuryev, Yuri	234 Schaefer, Gerald
403 Merkuryeva, Galina	218 Senkerik, Roman
97 Möhring, Michael	363 Sente, Paul
357 Montes, José Luis	261 Simino, Riccardo
84 Mosler, Hans-Joachim	48 Sokolov, Boris V.
254 Mwansa, Laban	164 Somarakis, C. E.