SEMANTIC FEATURES AND FACTUAL AGENTS: A MODEL TO REPRESENT CHANGEABLE SITUATIONS

Thierry Galinho & Frédéric Serin
LIH, Computer Science Laboratory of Le Havre
University of Le Havre,
F76058 Le Havre Cedex, France
E-mail: Thierry.Galinho@univ-lehavre.fr, Frederic.Serin@univ-lehavre.fr

KEYWORDS

Factual Agent, Semantic Feature, Decision Support System, Ontology Representation.

ABSTRACT

In this paper we present a part of a Preventive Monitoring Information System. First, we describe the global architecture of the multiagent decision support system. Then, we focus on one organization of agents called *factual agents* whose aim is to represent semantic features. We also expose our ontology with its modeling. Our goal is to store and to use general and specific knowledge and information, in order to compare factual agents.

INTRODUCTION

We propose to help managers evaluate all the aspects of the current situation using data coming from different information sources like databases, managers, sensors…The goal is to build the most accurate image as possible of the significant elements of the current situation. In other words, the system must help the actors to analyse the description of the current situation in order to assist management of operational decision-making. From this viewpoint, the information system is able to inform them about the current situation and past facts related to it. This system can also be used in a situation of crisis because it provides a synthetic view of the situation and it evaluates this situation in order to anticipate its potential consequences. The goal is to improve the decision-making process of the actors by providing them, as soon as possible, with the information about what occurs and what could occur.

In this paper we present a part of the information system which we call Preventive Monitoring Information System - PMIS (Boukachour et al., 2002). We will focus on knowledge representation and the organisation of factual agents which deals with it (Boukachour et al., 2003). We shortly present the functionalities of the PMIS, its architecture and its different components focusing on semantic features, factual agents and related ontology.

This paper breaks up into four parts: a presentation of the architecture of our multi-agent system with a detailed description of our model, a definition of the semantic features, a description of the agentification and particularly of the factual agents and, finally, a conclusion.

The structure of the factual agents uses known models and has a generic management structure for behaviour designed using an automaton. For these agents, we describe the mechanism of reinforcement and weakening of the agents which is managed at this time by the semantic features arrivals or by the interactions between agents.

MODEL & FRAMEWORK

The PMIS is based on a model using agent organisations. The agent paradigm allows to take into account the dynamic aspect produced by the evolution of the situation. The organisation of factual agents deals with the dynamic information representation process; this description includes the presentation of weighted graphs of the knowledge representation. An ontology is used to give measures to compare factual agents.

Our multiagent system (MAS) is made up various families of agents charged to recognize, to interpret, and to detect anomalies by correspondence with known or identified situations.

Global System Architecture

The system provides information about the current state of the situation. It receives facts and information coming from different sources and must: validate information, place information in relation to the situation context, evaluate the possible evolutions of the situation and give their potential consequences, dynamically increase the relevance of the situation description.

Our PMIS is made of three interfaces and a kernel based on different agent organisations (Boukachour et al., 2002) (Boukachour et al., 2003): a Human-Computer Interface for the decision-making actors, a query interface

Proceedings 18th European Simulation Multiconference
Graham Horton (c) SCS Europe, 2004
ISBN 3-936150-35-4 (book) / ISBN 3-936150-36-2 (CD)

which has access to different Information Systems. Indeed, information about the situation can be retrieved using various Distributed Information Systems. A conjectural interface which has access to a base of scenarios. The system kernel makes the connection between these three interfaces allowing them to communicate. This kernel contains the organisation of factual agents.

The situation is represented by an organisation of factual agents which is dynamic. This property allows to benefit from emergence mechanisms that lead to a dynamic configuration of the organisation representing a summary of the current state of the situation.

Graph Framework

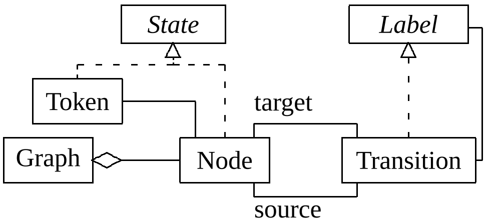

Figure 1: Graph Framework

We have created a framework called *Graph* that is made up of six classes (see figure 1). This framework is used to describe both ontologies for semantic features comparisons and automata for agent behaviours. We carry out a *labelled graph* to represent it. This graph is both itself a framework, and a part of the frameworks defining aspectual agents and ontology representation (see below).

This framework contains the classes required for the construction of labelled graphs. A graph aggregates nodes; nodes are linked by transitions. A label inputs informations into the graph coming outside from sensors or agents for example. The state notion has been created to be shared both by nodes and tokens. Token is usefull to simulate the travel through a graph for example a vehicle in a road map, a token in a Petri net, several objects or agents sharing the same automaton, etc. A label defines a generic type of object for communication from a node-state to another one.

A transition has the label's signatures as behaviour, and it is able to transmit the received messages to a linked object of label type. The differentiation between transition and label is usefull when informations are present in two or more transitions. A classical example is an object *way* shared by two transitions representing two possible directions.

In an ontology graph, the labels are qualifiers of the links between nodes. We have for example: links of semantic relation: "is a specialization of", "is a generalization of", "is similar to"; links of causality or dependence; links for composition or aggregation; links of action which can carry out an action.

In an agent, one automaton describes the different states that determine its behaviours. We use the token notion because the same automaton can be shared by several agents or, for the same agents, to describe different roles.

Creation of an Ontology

Agents have to communicate in a way that makes sense for them, so they must share the same language and vocabulary. This is evident according to FIPA communicative acts. However, we define our own vocabulary and semantics for the content of the semantic features. This means defining an ontology. In other words, an ontology is an explicit, partial specification of a conceptualization that is expressible as a meta-level viewpoint in a set of possible domain theories for the purpose of modular design, redesign and reuse of knowledge-intensive system components (Guarino, 1996).

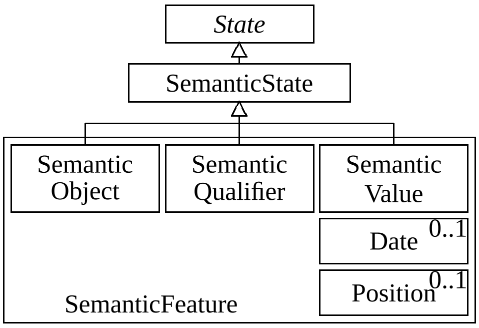

Figure 2: Categories of semantic elements

We have created different ontologies to represent the relationships between terms coming from different environments. These terms are included in the semantic features. The relationships existing among the terms are modelized with labeled transitions according to our graph framework.

SEMANTIC FEATURES

Presentation

Our system uses a model based on both a system of reception and an analysis of elementary information. These elementary informations are presented in the shape of semantic features. Each semantic feature (SF) is regarded as a fact, it is sent to the MAS which assimilates it by creating its corresponding factual agent. These agents treat these SFs (messages) and can decide whether to take into account the contained information (Wooldridge et al., 2000). The comparisons between semantic features will be done using factual agents which contain the SF. The organisation of the agents in groupings with the use of algorithms of clustering (Coma et al., 2003) will allow the comparison between the clusters of agents and scenarios.

Information translates both particular and partial aspects of an observed situation. It produces a set of SFs. Each SF represents an elementary information. The modeling of the SFs allows to obtain a homogeneous structure in the studied information system. This homogeneïty is of primary importance because it allows to establish comparisons between SFs. Actually, with these comparisons, the system is able to evaluate a current situation by comparing them with referred situations (called scenarios). These situations of reference result from passed experiments, studied situations, deductions, analysis or extrapolations. We need to define the set of the observations sent to our system, that is the goal of the followed section.

Let us define a SF (semantic feature) by several properties. Each observation is an information with an index of confidence (not yet used) and a *semantic value*. A SF is an elementary information coming from the environment and the semantic value produces some semantic features.

The SF θ is an elementary information and it is part of the observation ω. It consists of five values. It integrates the event date and the location of the information. The over three informations of the SF constitute a triplet made up of an object, a qualifier of this object and a value attached to this qualifier (see figure 2). This triplet is defined by:

1. first component of a semantic feature is an object related to a certain type of object $c \in \mathcal{C}$,
2. its qualifier $q \in \mathcal{Q}$,
3. a set of possible values $\mathcal{V}_{c,q}$ which are related also to a type of object c.

Our model takes inspiration from a real case study. At the beginning of this study, no type of object was defined *a priori* . This study allows us to test our model and to define these types of objects.

The index of confidence and the origin of the information must be treated upstream (ontological treatment and determination of confidence) of the creation of the semantic feature. They will not be treated here as an indication of confidence of the observation (or rough relevance of information). The two others parameters constituting the SF are the date of the observation, and the location of the fact described by the information.

The different types of objects issuing from the study can take five identified values: *phenomenon*, *action*, *object*, *person* and *mean*. Phenomena and actions have dynamic properties, for these objects, it is necessary to associate complete temporal data: time of beginning, duration, time of observation… The other types of objects are regarded as descriptions of a persistent situation (at least until it is invalidated by new information). The objects, the means and the persons are called entities or persistent objects. To summary our study, we distinguish two types of objects: dynamic objects and persistent objects. Phenomena and actions are activities respectively observed (or indirectly noted) and started (or ordered). An action is an activity with a known origin and a determined immediate goal. Phenomenon is an activity which is not an action, it has an unknown origin or it is the result of an action or another phenomenon.

We define various qualifiers and their associated values. For example:

- Qualifiers shared by the actions and the phenomena: "is-a", "state-change", "beginning-hour", "space-localization", "scale".
- Some qualifiers are specific to an action: "activity", "localization", "target-object".

The value v_{q_c} associated to the qualifier q_c can be, in some cases, identified by its type.

Persistent objects (objects, persons and means) are entities that seem persistent, objects are real entities like, for example, "vats", "valves", "vehicles". Persons are particular objects, they have an obviously significant value in the problems of management of risks and crisis where they are a goal of preventing except they are victims. They also have the characteristic to be able to have undefined behaviors. Means are sets, they join together objects and/or persons. This gathering of various entities allows specific actions. A mean may be also the way to qualify the property of particular action.

It appears, with the sight of the experiment, that each qualifier can be typed. The typing of q_c allows to define the set of the possible values v_{q_c}. This typing is significant in order to be able to establish clearly and formally rules of comparisons. The qualifiers have a representation close to the attributes' representation in a class. For example, the objects have a space localization and an identity.

The values are quantitative or qualitative, the qualifiers depend on the types of objects which they are linked up. Managing comparisons between the quantitative qualifiers is more easy than establishing a relationship between the qualitative values. However, the ontologies permit to define some proximities between qualitative values.

Proximity

The proximity of the semantic features is useful to be able to have a distance between two SFs. We aim to lay out in a formal way of one or several functions which compute the distance between two SFs or two families of SFs. Establishing distances between SF allows to reinforce or to weaken the factual agents carrying the SFs. We limit the properties of the distances to define our proximities and we adopt dissimilarities (a dissimilarity does not respect the property of the triangular inequality).

We distinguish three types of proximities: time proximity (P_t), spatial proximity (P_e), and semantic proximity (P_s). We introduce time proximity to take into account that more two events are distant, more the proximity is small. For the spatial proximity, the same reasoning is applied. We can speak about time and spatial distances. The global proximity between two SFs multiplies together these three proximities.

$$P_{SF} = P_t \times P_e \times P_s$$

Let us use Δt be the difference of time.

$$P_t = \frac{4e^{-\Delta t}}{(1+e^{-\Delta t})^2}$$

Δe is the difference of space.

$$P_e = \frac{4e^{-\Delta e}}{(1+e^{-\Delta e})^2}$$

These two proximities (for time and space) are a sigmoid function, it takes into account the negative values. It is written to remain on the interval $[0, 1]$. It brings the five following advantages: its continuity, its derivability, the knowledge of its primitive, its definition on $\Re$ entire (including negative values) and its symmetry in zero.

The definition of a semantic proximity is related to the definition of an ontology. Proximity between two semantic features θ_1 and θ_2 provides a value on $[-1, 1]$. For example, $P_s(\theta_1, \theta_2) = 0,8$ signifies that the two SFs are relatively close semantically speaking. Such measurement of proximity must relate to an ontology. This ontology graphically appears as elements in relations the ones with the others. These elements can thus be represented like nodes of a graph linked by labelled transitions. We carry out a representation using the labelled graph (see figure 1). The labels are qualifications of the links. Some of these labels can be very close such as "causes" and "can create an event". It is necessary to clearly define each field with regard to each label.

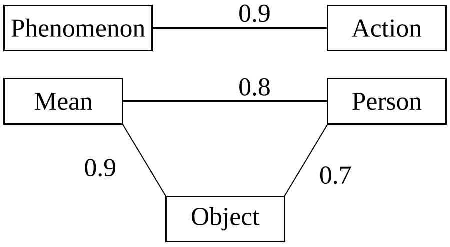

Figure 3: Graph for proximities between objects

We have created three ontological graphs. We have the graph of proximity between the five types of objects. We build this graph by taking the following values of proximity (see figure 3). The two others define proximities between qualifiers, and proximities between values.

STRUCTURE OF FA

Information is coming from different sources and we do not know if a specific datum will be important or not. So, we inject the data in the MAS to let emergence detects some subsets of all the information (Boukachour et al., 2003).

We design factual agents for managing a semantic feature. The agent must be able:

- To represent a semantic feature.
- To compare with other SFs, i.e. to compare with other FAs in respect with the ontology.
- To try to achieve its goal (modelled by a multi-state automaton).
- To measure its own evolution and to compute its strength.

Behaviour Description

We will now describe the states and the transitions of the automaton which models the behaviour of the FA. Factual agents are both reactive and cognitive.

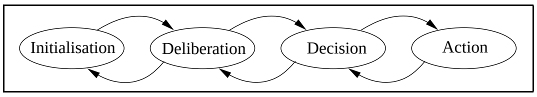

Figure 4: The inside automaton of a factual agent

As shown in figure 4, the four states of the automaton are ((Sfez, 1992)(Cardon, 1997)):

1. initialisation which is the state where an agent is created. The created agent receives messages (i.e. SF) from other agents, it updates its internal variables when it detects than some SF are close - in respect to the ontology - to its own SF. Agents stay in that state until internal variables increase enough to go over the threshold value and then change to the next state.
2. deliberation. In that state, the agent asks all the others to receive information it can use to determine if the sender is close, neutral or opposite to its SF. In that state, the agent will be more active than in its initialisation state.
3. decision is the state when the global activity of this agent rises. As in the previous states, it keeps listening to SF. But, it starts to be active by attacking one of its enemies (with opposite SF) he knows. The strategy here is to attack one of the weakest enemies.
4. action is the strongest state of the four. The goal of the agent is to try to reach that state. In that state, the other organisations of agents, as for example the clustering part, need to consider the semantic feature embedded in the agent to be of some interest to represent the current situation. If the support for that feature decreases, the agent goes back to the previous state. Here the strategy for aggression is to attack some of the strongest enemies chosen in the accointance network.

Going from one state to another is based on a comparison between some threshold values and the internal environment of the agent. The structure of each FA assimilates information which comes to it both in the form of SFs and in the form of interactions between FAs. These last interactions are of three types: aggression, defense and collaboration. The behaviour of each FA is given according to a state contained in an automaton.

As we explain in previous section, each semantic feature (SF) creates a factual agent (FA) in the system. The behaviour of a FA depends of its state that is a part of its knowledge. The state is a node of the automaton of the FA.

The knowledge of a FA consists of four families of fields : semantic feature, strength (force, energy and power of this agent), state (that determines the behaviour) and automaton (instance of a graph, proactiveness of this agent). The current state is a token moving in the automaton. So, implicitely, the automaton determines the behaviour of the agent.

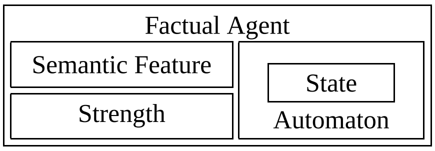

Figure 5: Fields Categories in a Factual Agent

An Agent is an atomic unit of automomous behaviour. The core concept of Agent is independent of how the autonomous behaviour is embodied. An agent has purpose, means of receiving information from its environment, means of performing actions, mental state (knowledge and beliefs) expressed as relationships between the agents, a specification of how it responds to what it perceives in order to carry out its purpose.

Behaviour includes reasoning and decision-making processes that affect mental state, and mental state includes behaviour-related elements allowing it to perform complex, goal-directed (Caire et al., 2001). The internal behaviour of an agent can be described as a repeating three-stage cycle:

- Perception: an agent perceives its environment through Information.
- Decision: an agent decides what tasks and actions to perform on the basis of his objectives and knowledge and beliefs.
- Reaction: the agent performs the actions that it decided upon.

Scale & Thresholds

Scale represents the importance of an activity (impact on its environment) or the size of a persistent object (occupation of space or cardinality). All distances are related to a scale. For example, if the studied problem relates to atoms, a distance of 1 m is of a gigantic size; while it is infinitesimal for an astronomical study.

For the proximity computation, the used reference value is 0 that is to say neutral action of the arriving SF. If the value is different to 0, it intervenes in the computation of the strength of the FA. Some different positive values with their signification are represented in the figure 6. The negative values mirrors the positive ones (replacing *close* by *different*).

Restricting the values in a given range allows to apply the strategy to any problem independently of the real values. The different indicated marks in the figure just help the user to fix the coefficients for the semantic distances.

The thresholds are the values used in the transitions of the FA's automaton (see above in the behaviour description).

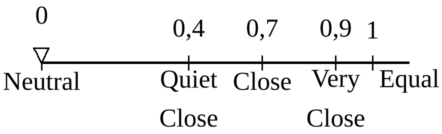

Figure 6: Scale for proximities

Strength of a FA

The reinforcement or the weakening of a factual agent is computed by the use of the interest measurement (also called P_{SF}. We use the range $[-1, 1]$ to indicate the interest of a SF for a FA viewpoint. This interest value modifies the force of the factual agent.

A factual agent (FA) has a weight m_{FA} and a speed of $\vec{v_{FA}}$. It receives a semantic feature (SF) with the force $\vec{F_{SF}}$. The power given by the SF to the FA is $P = \vec{F_{SF}}.\vec{v_{FA}}$. The initial energy of the factual agent is $E_{FA} = \frac{1}{2}mv_{FA}^2$. This energy is increased with $\Delta E = T = \int P.dt$ called the work of the force.

The force $\vec{F_{SF}}$ applied to the FA is function of the proximity between the SF and the encapsulated semantic feature of the FA. The new speed of the FA becomes $v_{FA} = \sqrt{\frac{2(E_{FA}+\Delta E)}{m}}$

The speed, the force, the power, and the energy characterize the strength of the FA.

CONCLUSION

The introduced framework built on factual agents has been developped with an empirical validation from a real world case study. The case study we use is defined with data coming from Total's minutes book derived from a crisis training. The informations of this French petroleum company permit us to create a specific ontology, to determine objects in the ontology graph and to define the different elements of a semantic feature.

This work is part of a long term project. We are developing a prototype to test the collective behaviour of our factual agents and to correct values embedded in the ontology. One of the next steps is to connect factual agents to other organisations such as the clustering part which is developed by other members of the team (Coma et al., 2003). Clustering agents characterise the factual agent organisation in order to provide a synthetic view of the current state of the situation. They identify groups. Indeed, if a group containing factual agents in great development during the same period is identified, these agents probably contain important semantic features according to the current situation.

Among the further works we have to do, we can list: to develop automatic acquisition of new semantic features not correctly described by the ontology, to learn new scenarios to fill the base of scenarios (actually manually done). But we also have to wonder about some decision we took: is a graph the most suitable representation for the ontology, do we have to add or change the internal variables of the factual agent?

At this time we are developing a graphic system to track the behaviour of the factual agents and to improve the representation of the current situation by the organisation of the factual agents.

THIERRY GALINHO received his Ph.D. degree in Job-Shop Scheduling in 1994 from University of Rouen. Dr. Galinho is currently Associate Professor at University of Le Havre. His current application domain now concerns Agent-Based Software Engineering and Logistics.

FREDERIC SERIN received his Ph.D. degree in Object-Oriented Simulation in 1996 from University of Rouen. Dr. Serin is currently Associate Professor at University of Le Havre. His current application domain now concerns Agent-Based Software Engineering.

References

Boukachour, H., Galinho, T., Person, P., and Serin, F. (July 2003). Towards an architecture for the representation of dynamic situations. In *Las Vegas IC-AI 2003: 659-664*.

Boukachour, H., Simon, G., Coletta, M., Galinho, T., Person, P., and Serin, F. (July 2002). Preventive monitoring information system : a model using agent organizations. In *SCI 2002, Orlando, USA*.

Caire, G., Coulier, W., Garijo, F. J., Gomez, J., Pavón, J., Leal, F., Chainho, P., Kearney, P. E., Stark, J., Evans, R., and Massonet, P. (2001). Agent oriented analysis using message/uml. In *AOSE2001, Montreal, Canada*, pages 119–135.

Cardon, A. (1997). A multi-agent model for co-operative communications in crisis management system : the act of communication. In *7th European-Japanese Conference on Information Modelling and Knowledge Bases*, pages 11–123.

Coma, R., Simon, G., and Coletta, M. (2003). A multi-agent architecture for agents clustering. In *Agent-Based Simulation'2003, Montpellier, France*.

Guarino, N. (1996). Understanding, building, and using ontologies. In *Tenth Knowledge Acquisition for Knowledge-Based Systems Workshop, Banff, Canada*.

Sfez, L. (1992). *Critique de la décision*. Presse de la Fondation Nationale des Sciences Politiques.

Wooldridge, M., Jennings, N., and Kinny, D. (2000). The gaia methodology for agent-oriented analysis and design. In *Journal of Autonomous Agents and Multi-Agent System*, volume 3, n°3, pages 285–312.