THE RELATIONSHIP BETWEEN FUZZY REASONING AND ITS TEMPORAL CHARACTERISTICS FOR KNOWLEDGE MANAGEMENT

Vasile MAZILESCU
Daniela SARPE

University "Dunarea de Jos" of Galati
vmazilescu@ugal.ro
dsarpe@ugal.ro

The knowledge management systems based on artificial reasoning (KMAR) tries to provide computers the capabilities of performing various intelligent tasks for which their human users resort to their knowledge and collective intelligence. There is a need for incorporating aspects of time and imprecision into knowledge management systems, considering appropriate semantic foundations. The aim of this paper is to present the FRTES, a real-time fuzzy expert system, embedded in a knowledge management system. Our expert system is a special possibilistic expert system, developed in order to focus on fuzzy knowledge.

***Keywords**: Knowledge Management, Artificial Reasoning, predictability.*

1. Introduction

Knowledge management is a discipline focused on systematic and innovative methods, practices, and tools for managing the generation, acquisition, exchange, protection, distribution, and utilization of knowledge, intellectual capital and intangible assets. The knowledge management systems based on artificial reasoning (KMAR) tries to provide computers the capabilities of performing various intelligent tasks for which their human users resort to their knowledge and collective intelligence. At present KMAR is a highly economically important field due to its ability of approaching new sets of problems, different from those dealt with by the classical systems, such as: perception, decision making, planning, diagnosis, natural language comprehension, enterprise knowledge management, learning, semantic technologies, web service interfaces, Semantic Web, etc. The Semantic Web relies on structured sets of metadata and inference rules that allow it to “understand” the relationship between different data resources.

The technologies that form the basis of the Semantic Web by adding these metadata and inference rules are RDF (Resource Description Framework), RDFS (RDF Schema) and OWL-Web Ontology Language (Afandi R., et. al. 2006, Naeve A., et. al. 2005, Sampson D., et. al. 2004).

A key issue for developing cognitive systems is the distinction between architected or human modeled ontology on the one hand, and emergent, largely machine-automated knowledgebase construction, on the other. Therefore, there are in KMAR three large groups of problems that should approach in terms of decision-based applications: human-environment interface, qualitative knowledge modeling and time management. Such applications obviously require dated event operations the life-time of which should be managed by the system, which often works asynchronously with the acquisition and control system. Time restrictions are not excessive in usual applications. Critical time reasoning problems may occur in case of faulty operations and overloading. At present, the reasoning depth developed for such system is still poor.
Conventional expert system shells are too slow for real-time environments, and their inference process is unbounded. We need a reactive, interruptible system that can assimilate data and asynchronous events, and present the operator with a reasoned opinion in a timely manner. Speed alone is not enough. A real-time expert system shell must also represent imprecise, time

This paper was recommended for publication by Prof. Viorel MINZU, PhD

and temporal data, encode temporal knowledge, and manage temporal/fuzzy reasoning (Zadeh, 1983, Zhang and Yang, 1993).

We have developed an object-oriented fuzzy real-time expert system (FRTES) shell to meet the challenges of the dynamic environment. We have investigated the relations between fuzzy reasoning and its temporal characteristics. For this, the impact of ontology and knowledge management has been specified in section 2, and the embedded temporal aspects in FRTES are defined in section 3. To illustrate the theoretical results we provide in section 4 an example of fuzzy reasoning based on knowledge-model for a balancing problem in a specific structural definition. Section 5 presents concluding remarks on future semantic technologies and Semantic Web. Comparisons to relevant research are made throughout the paper.

2. The impact of Ontology and Knowledge Management

In the global economy, knowledge has become a key asset in organizations. Knowledge Management (KM) has emerged as a major issue that managers must deal with. To conceive and implement KM is becoming more complicated, because it must assess complex and confusing situations, initiate KM, identify a lot of causal or abductive relationships between components, make appropriate decisions, and guarantee that the recommended action plan will be effective. The last property is essential. In the past few years, the emergence of knowledge management has facilitated the progress for the knowledge demander in searching for knowledge efficiently and effectively [Barthes, J. et. al. 2002, Hendler, J. et. al. 2001].The activity of knowledge management is wide and complex. It can be the management of individual knowledge or the operation of enterprise knowledge. It also includes activities that form the communication of tacit knowledge to the integration of explicit knowledge. In order to achieve the goal of knowledge management, ontology has been considered as an adequate methodology to support a variety of activities of knowledge management, including knowledge retrieval, store, sharing, and dissemination [Pundt, H. et. al. 1999]. In one of the most popular definitions, ontology is the specification of a conceptualization [Waterson, A. et. al. 1999]. For knowledge management system in enterprises, ontology can be regarded as the classification of knowledge. That is to say, ontology defines shared vocabulary for facilitating knowledge communication, storing, searching and sharing in knowledge management systems [O'Leary D.E., 1998]. Defining ontology is a time-consuming and laborious task. In general, the identification and application of ontology is only for some specific domain, such as medicine, industry, or the enterprise. The basic activities of knowledge management are knowledge acquisition, creation, sharing/diffusion, and utilization. There are a variety of technologies have been applied to support these activities, such as e-mail, database and data warehouse, group decision software, intranet and extranet, expert system, intelligent agent, data mining etc. There also exist different knowledge management systems (KMS) that facilitate the activities of knowledge management [Chau K.V., 2002].

In 1990s, the knowledge reuse and sharing already became the major issue in knowledge engineering. To achieve the goal of knowledge reuse, the concept of object orientation has been introduced to knowledge management systems. In the sense a knowledge entity can be treated as a knowledge object (KO). KOs can be numerical data, text streams, validated models, meta-models, movie clips, or animation sequences. Since enterprises are interested in the integration of existed knowledge bases [Waterson A., 1999], how to integrate and share KOs among different KMS is of great necessity and is a crucial challenge. In the literature, metadata has been widely used in the integration of existed knowledge bases [Tiwana A., 2001] whereas the ontology has been considered as a meta-level description of knowledge presentation [Guarino N., 1997]. A three-level architecture for intelligent decision support is possible to be proposed. It contains, from the top to the bottom, application level, description level, and object level. The object level comprises various information and knowledge sources, the so-called KOs. Ontologies are in the description level, which enable users in the application level to intelligently access object-level sources. Users can precisely select and efficiently access knowledge via the description level from the application level. In other words, ontologies are metadata that provide the search engine with the functionality of a semantic match. It is different from traditional search engines that directly search for the contents of data. Without doubt, the most popular markup language of metadata is XML. With the mature of the XML development, different definitions of metadata have been proposed, such as ebXML. From the viewpoint of ontology, XML is not suited to describe the interrelationships of resources in the Internet. Therefore, W3C has proposed the resource description framework (RDF) and RDF schema (RDF/S) [Lassila O.]. Since then, many ontology tools have been developed for implementing metadata of ontology by using RDF and RDF/S , like Ontoprise. Each ontology tool has its characteristics and advantages. KAON, Ontoprise, and Ontopia provide a complete set of ontology tool suites for building, maintaining and utilizing ontologies. In particular, KAON and Ontopia these tool suites can be deployed onto Java J2EE architecture, a distributed component-based architecture, which makes the ontology-based system more flexible and robust. For this reason and the consideration of open source, KAON is chosen as the ontology development platform in this example.

3. Temporal aspects in FRTES

Real-time systems span a broad spectrum of complexity from very simple microcontrollers to highly complex and distributed systems (Stankovic and Ramamritham, 1993). These complex future systems include the space station, integrated vision/robotics/AI systems, collections of human/robots coordinating to achieve common objectives (usually in hazardous environments), and various command and control applications. To further complicate the problem there are many dimensions along which real-time systems can be categorized. The main one includes: the granularity and the strictness of the deadlines, reliability requirements of the system, the characteristics of the environment in which the system operate. The characteristics of the environment, in turn, seem to give rise to how static or dynamic the system has to be. However, one common denominator seems to be that all designers want their real-time system to be *predictable*. It means that it should be possible to show, demonstrate, or prove that requirements are met subject to any assumptions made, for example, concerning failures and workloads. In other words, predictability is always subject to the underlying assumptions being made. In this section we concentrate on predictability with respect to the timing requirements. The use of temporal aspects refers to the design of those tools to solve the following metaequation: *Time = complexity ⊕ real-time ⊕ temporal reasoning*, which is employed in order to integrate time into a process control application (Mazilescu, 1999). This equation is formally found on the inference engine algorithm, able to make full use of the specific knowledge to the process control. The symbolic aggregation metaoperator ⊕ can be instantiated into different classes of specific operators, depending on the goal pursued by the control model. We assume that the process operates like finite nondeterministic system, while the FRTES will operate like a finite deterministic state machine.

The closed-loop control expert system can be modeled like a nondeterministic state machine, whose outputs are the process outputs. A major obstacle to the widespread use of (possibilistic) expert systems in real-time domains is the non-predictability of rule execution time. A widely used algorithm for real-time production systems is the Rete algorithm. To achieve a fast reasoning the number of fuzzy set operations must be reduced. For this, we use a fuzzy compiled structure of knowledge, like Rete, because it is required for real-time responses and a fuzzy inference engine (Mazilescu, 1998). The engine represents a method of fast fuzzy logic inference. It must provide guaranteed response times, completing its reasoning within a deterministic amount of time. Systematic analysis methods must be used so that the possibilistic expert system behavior can be studied quantitatively within the developed modeling framework.

The relationships and the analogy between expert and control system architectures are important problems for intelligent control (Passino, et. al. 1994). This is possible because both are problem solving systems with different problem domain (environment) the expert system reasons about and takes actions on. The problem domain must be defined as a collection of problems that the expert system desires to solve. In conventional control, the plant is a dynamical system, described with linear or non-linear differential/ difference equations. An artificial intelligent expert system consists of the planner or the inference engine, the problem domain, the exogenous inputs, and their interconnections. The outputs of the FRTES are the inputs (control actions) to the problem domain. There are unmeasured exogenous inputs to the problem domain (disturbances) that represent specific uncertainty. The measured exogenous input to the FRTES is the goal, and it must represent imprecise, time and temporal data, encode temporal knowledge and manage temporal/fuzzy reasoning. Following a conventional control-theoretic approach, we can introduce a mathematical model for the plant P and the possibilistic expert control system (PECS), which consists of the possibilistic expert system (PES) and the plant (Mazilescu, 1998). The PES must be designed so that it can coordinate the use of the plant outputs and reference (user) inputs, to decide what plant command inputs (or hypothesis/ conclusions) to generate so that the closed-loop specifications are met. Although the PES (viewed as an expert system) are frequently being used to perform complex control functions, most often it is the case that no formal analysis of the dynamics is conducted because mathematical analysis of such systems is often considered to be beyond the scope of conventional control theory.

The FRTES requires adapting the representation of knowledge in order to operate it and to improve the efficacy of its operating using the compilation technique. The accepted data are: variables, atomic and fuzzy constants. The fuzzy constants may appear both in facts and in rules and are always associated with a fuzzy set trough the *constfaz* function. The possibility distribution modeling allows a unified framework for the representation of imprecision and uncertainty (Dubois and Prade, 1991). The λ parameter is used to measure the uncertainty of the fuzzy sets ($0 \leq \lambda \leq 1$). If a fuzzy set is uncertain, the λ parameter must be declared in the *constfaz* function through a list (uncertain λ). We shall admit that a fully uncertain fuzzy set ($\lambda=1$) does not affect the behavior of the system. A fact does not entail variables, that are the terms permitted in facts are only the atomic and the fuzzy constants. In opposition with the facts, a cause is a structured list in which variables may appear. This implies the presence of the variables, of the atomic and fuzzy constants in the structure of the causes. The causes may appear both in the conditional part and in the conclusion of the rules. The operation of the fuzzy expert system proceeds by the following steps:

1. *Acquiring the process output and reference input events at time k;*
2. *Forming the conflict set in the fuzzy match phase from the compiled set of rules in the fuzzy knowledge-base and based on e_{uk}, the current status of the truth of various fuzzy facts, and the current values of variables in the knowledge-base;*
3. *Using conflict resolution strategies (refraction, recency, distinctiveness, priority, and arbitrary) in the select phase to find one rule r' to fire;*
4. *Executing the actions characterized by the consequent of rule r' in the act phase.*

he timing of the event occurrences in FRTES fuzzy expert system is such that the PES is synchronous with the plant. Although every occurrence of an input event of the plant always affects the expert system state, the occurrence of an input event of the expert system does not necessarily immediately affect the plant state. In qualitative analysis of our fuzzy expert system, the focus is on testing if the plant, expert system, and especially the closed-loop PECS satisfy certain properties, as follows: reachability, ciclic properties and stability (Mazilescu, 1999).

We can also analyze the properties of the isolated fuzzy expert system (i.e., *without* the plant). In our case the "plant" is fuzzy compiled knowledge-base, the "fuzzy expert system" is the fuzzy inference engine, the "command inputs" are the changes that the inference engine makes to the knowledge-base, and the "outputs" of the closed-loop system are fuzzy facts or variables in working memory (that the inference engine uses in its decision-making process).

4. Fuzzy Reasoning in Load Balancing Problem

An example is a load balancing problem (LBP) and it is described by a directed graph (C, A) where $C=\{1,2,..., N\}$ represents a set of subsystems $i \in C$, and $A \subset C \times C$ is the set of connections between them ({(1,2), (2,1), (1,3), (3,4), (4,3), (4,2), (3,5), (5,6), (6,5), (6,4)}) (Passino, et al.1994). We require that if $i \in C$ then there exists $(i,j) \in A$ or $(j,i) \in A$ fore some $j \in C$ (i.e., every machine is connected). Also, if $(i,j) \in A$ and if $(i,j) \in A$ $i \neq j$. Each machine has a buffer which hold load, given by x_i, $x_i \geq 0$. Each connection $(i,j) \in A$ allows for subsystem i to pass a portion of its load to subsystem j. It also allows subsystem i to sense the size of the load of subsystem j (for any two subsystem i and j such that $(i,j) \notin A$, i may not pass load directly to j or sense the size of j's load). This problem appears also in the paper (Mazilescu, 1998). Below we consider the discrete case: when the load is in the form of fixed uniform-sized blocks that cannot be subdivided. In this case, the crisp knowledge base contains twelve rules R_i, $i=1,...,12$ in G2, of the following type:

R_i: If (the charge of m1 >= the charge of m2) and (the charge of m1 >= the charge of m3) and (the charge of m1 >= the charge of m4) and (the charge of m1 >= the charge of m5) and the charge of m1 >= the charge of m6) and (xb[1] $\neq$0) and (xb[3] $\neq$ 0) and (the charge of m1 $\neq$ the charge of m2)

Then *in order conclude* that xb[0] = 1 and *inform* the operator and i*nfer* that "[the name of this rule], xb1 [xb[1]], xb2 [xb[2]], xb3 [xb[3]], xb4 [xb[4]], xb5 [xb[5]], xb6 [xb[6]], xb7 [xb[7]], xb8 [xb[8]], xb9 [xb[9]], xb10 [xb[10]]" and *start modify_charge* (xb[0]) and *conclude that* xb[2] = 1 and *conclude that* xb[4] = 1 and *conclude that* xb[6] = 1 and ?*conclude that* xb[8]=1 and *conclude that* xb[9] = 1 and *conclude that* xb[10] = 1 and *conclude that* xb[1] = 0.

In spite of its greater expressiveness, the present crisp model (knowledge base and the simulation results) for the discrete load balancing problem has several limitations: the load cannot be infinitely subdivided, so that not for any initial loads the problem has a good balancing, or acceptable. The expert control system does not have as many ways to perform redistribution, so that only imperfect or inexact load balancing can be achieved. In conclusion, the embedding a metaknowledge was used, like fuzzy knowledge (Klawon and Novak, 1996), represented in our formalism, so that the balancing problem will have good solutions in any initial load cases. This is similar with the continuo load-balancing problem, for which the qualitative analysis can be performed.

The set of command input events is $E_u = \{e_{00}, e_{12}, e_{13}, e_{21}, e_{35}, e_{34}, e_{42}, e_{43}, e_{56}, e_{65}, e_{64}\}$, where e_{00} denotes the event "no part is moved". In this case $E_d = \varnothing$, $E_o = X^{PECS}$. The terms f_e^{PES}, δ^{PES}, g^{PES} are defined in (Mazilescu, 1999). The sequence of events executed was ($e_{ij,k}$ means transfer parts from i to j at time k or HC means human control): e_{34}, e_{13}, e_{21}, e_{13}, e_{21}, e_{13}, e_{34}, e_{13}, e_{21}, e_{13}, e_{21}, e_{34}, e_{13}, e_{21}, e_{34}, e_{13}, e_{35}, HC, e_{13}, e_{21}, e_{43}, e_{35}, e_{21}, e_{42}, e_{43}, e_{56}, HC, e_{13}, e_{21}, e_{43}, e_{35}, e_{13}, e_{21}, e_{13}, e_{35}, e_{56}, e_{13}, e_{21}, HC, e_{43}, e_{13}, e_{35}, e_{13}, e_{21}, e_{56}, e_{35}, e_{13}, e_{21}, e_{35}, e_{13}, e_{56}, e_{35}, HC, e_{43}, HC, e_{43}, e_{35}, HC, e_{43}, e_{21}, HC, HC, HC, e_{13}, e_{00}.

It is obvious that the open-loop plant has cyclic properties that may prevent the open-loop from achieving the desired control objective. When closed-loop fuzzy expert control is used, as in our example, the invariant set exists, by simple analysis of the system dynamics. Using a search algorithm, we show that there exists at least one path from any given initial part distribution in the LBP. The reachability result (the BLP described above is reachable for all initial states, because there exists a sequence of events to occur that produces a state trajectory, so that the end state of the plant is in the invariant set). In our fuzzy expert system, any rule whose "partially matches" the current data can "fire" (i.e., contribute to specifying the control input). In the FRTES we consider here, there may be more than one rule whose antecedent "exactly matches" the current data, but our inference engine allows only one rule to fire at a time.

We have shown that conventional knowledge-based debugging tools can ignore important dynamic behavior that can result from connecting the full fuzzy expert system (i.e., with an inference engine) to user inputs and a dynamical process. We have illustrated the results by modeling and analyzing expert systems that solve a LBP as a simple control problem. The results of this paper shows that fuzzy expert control system are a class of (heuristically constructed) nonlinear control systems that can be studied with the analytical tools available from conventional control theory. Current research in real-time KMAR is driven by a need to make knowledge-based systems function in real-time, to be predictable, and a need to integrate approaches to handle non-linearities. Response time analysis is in general undecidable, and is PSPACE-hard in the case where all the variables have finite domain (Liberatore, 1997).

5. Concluding remarks

The work reported in this paper serves to promote the development of a firm mathematical foundation on which to perform careful analysis for the verification and validation of the dynamics of expert control systems that operate in critical environments. There are important another future directions for this work, investigating the dynamics of reasoning systems that utilize learning and planning in various complex applications, studying computational complexity issues relative to conflict resolution strategies and metaknowledge representation, and modeling realistic applications that involve intelligent models, like knowledge-based systems, Semantic Web agents, etc. Acquiring models means learning by observation, exploration and experiment, teaching and coaching, or reading. Using models, means reasoning including mental simulation and testing, hypotheticals, plausible inference, logical thinking, and value-based trade-off.. The Semantic Web has recently emerged as a new and highly promising context for knowledge and data engineering. Within an atmosphere of high expectations, many myths as well as many visions have exhibited a number of different approaches for the exploitation of the Semantic Web in both academia and industry. However, a struggling business reality requires a concrete strategy as well as the development of specific competencies from the knowledge and data engineering community in order to prove the value of the Semantic Web to society. The Semantic Web agent does not include artificial intelligence – rather, it relies on structured sets of information and inference rules that allow it to "understand" the relationship between different data resources. The computer doesn't really understand information the way a human can, but it has enough information to make logical connections and decisions. The true impact of the Semantic Web will not be known for quite some time, but its potential is staggering. Some Semantic Web proponents have asserted that it will lead to the evolution of human knowledge itself by allowing people - for the first time - to quickly filter and synergize the massive amounts of data that exist in the world in a relevant, productive.

References

- *R. Afandi, J. Zhang, M. Hafiz, and C. A. Gunter. (2006): Adaptive Messaging Policy. In 4th IEEE European Conference on Web Services (ECOWS'06). IEEE, December.*
- *Barthès, J.-P. A., and Tacla, C. A Agent-supported Portals and Knowledge Management in Complex R&D Projects, Computers in Industry, Vol. 48 (1), pp. 3-16, 2002*
- *Chau, K.W., Chuntian, C., and Li, C.W. Knowledge Management System on the Flow and Water Quality Modeling, Expert System with Applications, Vol. 22 (4), pp. 321-330, 2002*
- *Dubois, D., Prade, H. (1991). Fuzzy Sets in Approximate Reasoning. Part 1: Inference with Possibility Distributions. Fuzzy Sets and Systems, 40: 143-202. Part 2: Logical Approaches, Fuzzy Sets and Systems, 40: 203-244*
- *Guarino, N. Understanding, Building, and Sing Ontologies: A Commentary to Using Explicit Ontologies in KBS, Development, by van Heijst, Schreiber, and Wielinga International Journal of Human and Computer Studies 46, pp. 293-310, 1997*
- *Hendler, J. Agent and the Semantic Web, IEEE Intelligent Systems, March/April, pp. 30-37, 2001*
- *Klawonn F., Noväk V. (1996). The relation between inference and interpolation in the framework of fuzzy systems. Fuzzy Sets and Systems, 81, p. 331-354*
- *Lassila, O., and Swick, R. Resource Description Framework (RDF) Model and Syntax Specification, W3C Recommendation,World Wide Web Consortium; http://www.w3.org/TR/REC-rdf-syntax.*
- *O'Leary, D.E. Enterprise Knowledge Management, Computer, Vol. 31 (3), pp. 54-61, 1998*
- *Liberatore, P. (1997). On the Compilability of Diagnosis, Planning, Reasoning about Actions, Belief Revision, etc. In Proceedings of the Sixth International Conference on Database Theory (ICDT-97), p.144-155*
- *Mazilescu, V. (1998). Fuzzy real-time expert control system. Characteristics and semantic of fuzzy compiled models. Applications of Artificial Intelligence in Industrial Automation (Tempus Intensive Course), Ed. Academica Publishing, may 22-27, ISBN 973-97816-8-3, p. 26-1/26-15*
- *Mazilescu, V. (1999). SECOMBCF: Sistem expert fuzzy de conducere a proceselor tehnologice. Revista Română de Informatică şi Automatică, Vol. 9, nr. 3, p. 39-52*
- *Naeve A. (2005). The Human Semantic Web: Shifting from Knowledge Push to Knowledge Pull. International Journal of Semantic Web and Information Systems, vol. 1, no. 3, pp. 1-30*
- *Passino, K.M., Michel, A.N., Antsaklis P.J. (1994). Lyapunov stability of a class of discrete event systems. IEEE Transaction on Automatic Control, Vol. 39, p. 269- 279*
- *Sampson D., Lytras M., Wagner G., Diaz P. (2004). Ontologies and the Semantic Web for E-Learning. Educational Technology & Soc. J. Vol. 7, no. 4, pp. 26-28*
- *Tiwana, A., Ramesh, B. Integrating Knowledge on the Web, IEEE Internet Computing, Vol. 5, No. 3, pp. 32-39, 2001*
- *Zadeh, L.A. (1983). The role of fuzzy logic in the management of uncertainty in expert systems. Fuzzy Sets and Systems, n°. 11, p. 199-227*
- *Zhang, S.C., and Yang, X.W. (1993). A N1NF Temporal Relational Algebra. The Fifth IFSA World Congress, p. 521-524*