Logical Aspects Regarding the Conception of Intelligent Systems

Vasile Mazilescu
vasile.mazilescu@ugal.ro
Dunărea de Jos University of Galați, Romania

The present paper presents some logical aspects of symbolic AI Systems, applied to the Intelligent Systems design. The basic feature of these systems is represented by the processing of the imprecise and dynamic knowledge involved in the synthesis of some decisions. We consider that the paper tackles an up-to-date and little debated aspect especially in the field of decision fuzzy systems. Knowledge-based systems with real-time functioning bear features that the majority of classic systems do not have: reasoning is evolutionary and non-monotonous due to the dynamic nature of the application, and events can change the status of the expert management system. The management architectures based on symbolic techniques acquire characteristics specific to the domain problem and to the type of intelligent system based on a management knowledge structure.

Keywords: Intelligent systems, Logical models, Fuzzy and Temporal logics

1. Introduction

The purpose of Artificial Intelligence (AI) as a field of science is to provide computers a series of possibilities of performing certain tasks that the human is able to do. The approaches with respect to AI are useful in creating systems which support the handling of complex problems, such as understanding natural language, recognising shapes, managing processes, classifying [18,22,27]. The study of reasoning and intelligent behaviour processes refers to understanding, describing and solving problems. A rigorous definition of symbolic AI is based on a series of concrete knowledge about this discipline and, particularly about its logical foundations [2,8,38]. Logics is a discipline that appeared long before AI, and it developed due to reasons and objectives that are very different from those corresponding to AI. Logic is now widely recognized to be one of the foundation disciplines for information technology. It has been said that logic may come to play a role in information technology similar to that played by calculus in physics. Like calculus, logic has found applications in virtually all aspects of the subject, from software engineering and hardware design to programming and AI. While progress in the past has been enhanced by theoretical development, it has to a significant degree proceeded independently of theory. In the future this is unlikely to be the case. Many of the current aspirations are seen to require major theoretical advances, and many lie in areas where logic is one of the basic disciplines. Mathematical Logics appeared at the beginning of the 20[th] century with the purpose to provide a set of answers to certain basic mathematical problems, while its initial goal was to research aspects of computability and demonstration [35,37]. As symbolic, or mathematical, logic has traditionally been part both of mathematics and philosophy, a glimpse at the contributions in mathematical logic at these two events will give us a representative selection of the state of mathematical logic at the beginning of the twentieth century. At the International Congress of Mathematicians Hilbert presented his famous list of problems, some of which became central to mathematical logic, such as the continuum problem, the consistency proof for the system of real numbers, and the decision problem for Diophantine equations (Hilbert's tenth problem). However, despite the attendance of remarkable logicians like Schroder, Peano, and Whitehead in the audience, the only other talk that could be classified as pertaining to mathematical logic were two talks given by Alessandro Padoa on the axiomatizations of the integers and of geometry, respectively.

The basic purpose of logics is to study reasoning in various formal theories, and that is precisely why symbolic AI is based on these formalisms [1,11,12,19,26,29]. Logic is used to analyze the foundations of mathematics. Some parts of logic such as set theory, model theory, recursion theory, constructive mathematics are now considered as mathematics subjects in their own right. In linguistics suitable logics and logical grammars are constructed to analyze our use of language; for example, how we use pronouns, quantifiers, tenses and so on. The use of logic in linguistics is important for natural language processing in computer science.

Section 2 relates the general directions of logical models and the contribution of logics in terms of AI systems. Section 3 presents the basic objectives in the design of an AI System. There is a classification of the systems from the point of view of complexity and the basic techniques used for the synthesis of an inferential subsystem based on fuzzy logic. Section 4 presents the conclusions of some formal logic systems that will support the designing of the symbolic AI systems. From this point of view there is a distinction between real-time reasoning and reasoning over time, the latter being a characteristic specific to Knowledge-based Systems. That is why it was absolutely necessary to explain the significance of time at a designing and implementation level of our intelligent system, highlighting its implications in the concrete case of each concrete application.

2. General characteristics of logical models

Mathematical Logics are developed in three great directions: i) The Theory of Models. The model is considered the main concept of first order language semantics. The main objective of the theory of models is to describe the models from certain axiomatic theories, in order to highlight their mathematical structures (algebraic, topological); ii) The Demonstration Theory. The notion of demonstration is important here, and one can study complexity-related problems of various demonstrations on a single theorem, according to different deductive systems or languages (regularity, equivalences, similarities); iii) The Computability Theory. This is the case in which one studies the intrinsic notion of computable function, with its various models: the Turing machine, recursive functions, λ- calculus and combinatory logic with a set of properties. Starting from these notions we can study the decidability concept, together with a study of various models used to demonstrate the decidability or non-decidability of mathematical theories. The theory of abstract complexity that generally uses different variants of the Turing machine (as an elementary calculus model to evaluate the necessary calculus resources to solve certain given problems) allows us to establish the complexity classes for decidable problems [9,16,17,24,28,32,37].

We must also highlight the unity of these three great aspects. Calculus and decidability notions are strongly inter-related, and we can thus establish precise correspondences between them. The theory of models can contribute as regards decidability problems and can also provide the semantic justification of the rules of inference and of automated demonstration systems. This unit is an essential characteristic of logics which represents a fundamental theoretic aspect. From the symbolic AI perspective, we can find tendencies specific to the three characteristics that were presented above: semantics and the theory of models, demonstration and reasoning, computability and complexity, but often with different methods compared to theories applied only to mathematical logics. Regarding the contribution of logics in terms of AI Systems, we can mention:

i) On the theoretical level, logic contributes with a series of creation elements and methods such as the syntax/semantics/decision trilogy, coherence (verifying the coherence of the knowledge-base, maintaining the truth systems), decidability (for example, absence logic is not semi-decidable, contrary to first-degree logics, certain temporal logic are decidable, others are not), complexity of the decision methods. Logic is thus a reference model for the foundations of symbolic AI Systems [3,13,21,34].

ii) The formalisation of different types of reasoning. Logic has a normative role even in the absence of some certain knowledge. Automated demonstration, knowledge representation languages, logical programming languages they all represent means that support and integrate reasoning methods. The logic used to describe and create an AI system is precisely the one that applies directly from the point of view of classical mathematical foundations. Mathematical logic is characterised by a coarse classification, which does not distinguish between two representations of the same object, whereas AI systems are rigorously sensitive of different means of representing the object, especially in the presence of some uncertain and imprecise features. Intentional Semantics can be used to capture important aspects of this phenomenon (figure 1).

While practitioners and researchers are carrying on their efforts in terms of creating and building complex systems in the field of AI, it was generally concluded that the uncertainty is not present only in people's knowledge. Allowing a certain degree of uncertainty in describing complex systems may be the most significant means of simplifying them. Different types of uncertainty can be characterised and investigated in a rigorous way in the context of the fuzzy set theory. In this way, the ability to operate in an uncertain or partially-known environment represents one of the basic performances of any real-time AI system. These systems must be designed as multi-agent systems with the possibility of combining various knowledge-based techniques (in order to acquire and process information), with approximate reasoning methods. This will allow the AI system to better emulate the human decision-making process, characterised itself by imprecise knowledge and time variables [5,15,27]. Real-time calculus represents an intense research field, since the functioning correctness of a system in a dynamic and distributed environment depends not only on the logic of creating it, but also on the implied temporal aspects. These kinds of systems also include AI systems, which undergo different complex time restrictions, and comprise various levels of time granularity. Temporal knowledge represents an essential aspect for a great number of AI applications: planning, managing technological and economic processes, handling dynamic situations [4,6,23,30,31]. An intelligent system must bear reasoning capacities to take into account a series of events that can appear within a process: interruptions, limits of processing time, the synchronous or asynchronous aspect regarding the emergence of new information. Considering time must highlight two complementary aspects: handling temporal information and formalising reasoning over time and in real time. Certain approaches are based on numerical models, and others are based on symbolic time representations (figure 1). The reasoning that undergoes real-time restrictions has specific characteristics: real-time functioning often involves a temporal reasoning, but this aspect is not necessarily true backward.

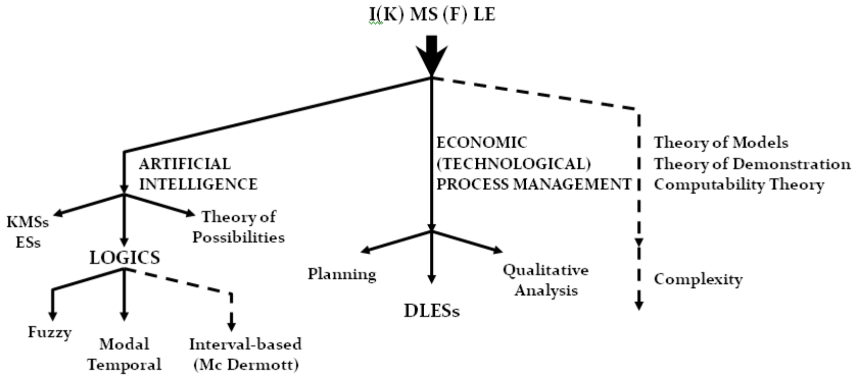

Figure 1. The main important theories and logics embedded in an I(K)MS(F)LE conception

3. The basic objectives in designing of a symbolic AI system

The symbolic AI Systems are guided by their purpose, and looking for the solution to a problem within its space represents their basic characteristic. We can highlight a quasi-continuity relation between the tasks specific to a problem domain characterised by sufficient information or knowledge in order to reach an objective (case 1) and those specific to a domain in which knowledge is insufficient, contradictory, uncertain or imprecise (case 2). In the first case there are explicit algorithms that transform the set of input data into a corresponding output quantity.

There is no search or backtracking notion. Any deviation in terms of execution time is uniquely associated to the dependencies between data, being the case of real-time conventional tasks. In the second case, neither the characteristics of the tasks, nor the interactions with the problem domain are known. There is a need of search heuristics within the space of the domain in order to determine a satisfying result, and there are great deviations regarding the execution timings. Due to these deviations associated to the problem-solving tasks, the traditional designing methods of real-time systems cannot be directly applied in the second case. The increase in the amount of involved knowledge can be applied in order to reduce deviations that occur as a result of the searching process. The searching process occurs on two levels in solving a problem: the level where knowledge is found again and the level of the knowledge specific to the application (the space of the problem). There is a series of methods for the implementation of these two levels [7,10,14,39].

The last three decades have been confronted with a dramatic change regarding the approaches of symbolic AI systems: until 1960 the uncertainty, inaccuracy and imprecision were rather negative attributes. A scientist that could not exactly define a statement was not regarded as a „true" scientist, and uncertainty was considered a disturbing element in the modelling process, being avoided as much as possible. The only accepted theory for the study of uncertainty was the probability theory, which was based on the occurrence frequency of an event and restricted to the situations in which the law of large numbers was valid, whereas uncertainty used to be associated to the stochastic nature. From 1960 on, there was accepted the fact that uncertainty and imprecision are part of the reality that we cannot change and obviously part of the man's current language. Real world is complex: the complexity that emerged due to uncertainty takes the shape of ambiguity. Starting from these observations, the following question was arisen: how can people succeed in rationalising over real systems? The answer is strictly related to the fact that people have the ability to approximately rationalise over the behaviour of complex systems, keeping only a generic significance over the problems. These generality and ambiguity are nevertheless enough in order to understand complex systems.

A classification of complex systems from the point of view of complexity and implicitly, of theoretical and practical means in their useful modelling, can be summarised as it follows: 1) for low-complexity systems, the wordings based on exact mathematical expressions are sufficient; 2) for more complex systems but for which there are enough significant data available, the neural models provide a powerful environment in order to reduce uncertainty, based on the learning of the configurations from the available data; 3) for highly-complex systems, for which there is a limited number of data, and the available information is ambiguous or imprecise, the reasoning based on the fuzzy logic is a modality to understand the behaviour of the system, through operations on certain symbolic structures associated to the observed inputs, the output situations, the intermediary variables that support the linking of certain inferential processes [25].

The present article is situated at point three of the above classification and is part of the approaches of symbolic AI systems. Its main purpose is to present the major aspects that characterise the creation of the inferential system based on the fuzzy logic for an expert system, with application in the management of discrete processes. The intelligent system elaborated by the author (SECOMBCF) with this purpose is an intelligent system with discrete logic events, and it includes compiled fuzzy knowledge. The basic time meta-equation proposed within this approach covers the term of complexity and (potential and reasoning over time) real time. The development of real-time systems must be subordinated to the causal specifications of these systems, since if we do not respect time restrictions we can make them become non-functional [31,33].

Real-time calculus is an open field of research not only in computers science but also in the conception of current technologies. It depends not only on the logic correctness of results, but also on the necessary time for their occurrence. Real-time systems play a major role in the present society and cover a large spectrum of applications that include experimental laboratories management, technological processes management, nuclear powers, air-force and robotics, military systems. Real-time systems are complicated and expensive, and time restrictions that these systems must accomplish are verified with ad-hoc techniques or with the help of some difficult simulation techniques. Integrating new components into the structure of a real-time system is really challenging and leads to a substantial growth of the costs of this system. The present generations of real-time systems are distributed systems that include intelligent and adaptive modules in order to satisfy some high dynamic performances. We can mention two characteristics for the current generation of real-time systems: their necessity for some AI abilities and the rapid evolution of a corresponding hardware. There is no science for the design of real-time systems. This means that a scientific approach of them is still not possible. Real-time calculus is the equivalent of a rapid calculus. The purpose of a rapid calculus is to minimise the average response time for a given set of tasks, and satisfying individual time exigencies of each task. The most important feature of real-time systems is their predictability, i.e. the behaviour of the system must be deterministic and must answer to the conception specifications. Rapid calculation represents a support for reaching the specifications, but it cannot guarantee on its own the predictability of the system. Other factors that contribute to create predictability are represented by a fast hardware and algorithms with good complexity. Many times the implementation language of a real-time application may not be sufficiently expressive to describe certain real-time descriptions [20,36,39].

An important aspect of the research in the field of real-time systems is constituted by the investigation of the effective strategies of allocating resources, an aspect that can be regarded as a performance element in the real-time systems engineering. An adequate designing of these systems implies finding some solutions also for other types of interesting problems regarding the specification and the verification of time behaviour of the system, as well as elaborating certain programming languages whose semantics to relatively correspond to time exigencies. An important problem in the study of real-time systems is the investigation of the role that the synchronisation mechanisms play. The problems related to the creation of real-time systems have been solved in fields of research, such as computers science and operational research. An important approach used on the large scale design of real-time systems is the hierarchical dissolution of systems in modules. Even though this methodology allows us to analyse the correctness of calculations on each level of abstraction, it does not support an analysis relatively to time and reliability characteristics of the overall system, which represents two vital aspects of real-time systems. It is necessary to develop certain designing technologies for these systems in order to include characteristics such as correctness, response-time, and reliability on each level of abstraction and to combine the results of each level within the integrated performances of the overall system. The main problem in stating and verifying a real-time system is the incorporation method of a time metrics. The usual approaches for stating the behaviour of a system is based on highlighting the events and the actions specific to the system as well as the order in which they occur. Including a real-time metrics creates problems in the semantics of competitor models and can complicate the verification problem of the system. Verifying these systems means satisfying the restrictions due to the environment and to implementation, and it is necessary to make a quantitative analysis prior to a qualitative one.

Management implies a strong relation between the process and the management system, which must react to the events that occur. In the given context, the management system bears certain AI features, if in the presence of some information of minimum orientation from the human governor; it can carry out complex actions as a response to the events from the outward environment. Intelligence can include the ability to accept certain abstract specifications of tasks in a general form of purposes/restrictions and the production of some reasonable actions that agree with the specifications. In any real-time AI system there is a fundamental compromise between the action and the reasoning. Time is a precious resource that uses up if the system must create reasoning over the actions before their execution. This time consumption can limit the number of the desired alternative actions and in this way the reasoning task becomes much more difficult to realise. In certain cases, the lack of success of an action can be the most unfavourable solution, while in other cases any of the actions can be better that total lack of it. The necessary time for reasoning can warn us over certain delays or disasters. The efficiency of a real-time AI system depends on its ability to allocate reasoning efforts in concordance with the process situations. The process of allocation is most of the time a difficult process due to countless overloads of information. The visible state of the process is huge and can contain incomplete, contradictory or uncertain information, which entails the use of some modules in the structure of the management system as some problem-solving ones. Furthermore, this information is often changing. It is actually impossible for a real-time system based on symbolic AI techniques to completely process all the information at a given moment and to choose a convenient type of reasoning, and to also respect all real-time restrictions.

That is why these systems must focus on some important sub-problems and to give time in a corresponding way to the available resources. The present research in the field of real-time AI is guided by the design and the creation of knowledge-based systems that can be integrated in management applications. The fundamental problem in the case of AI management systems is the fact that we do not know the most unfavourable execution time. This leads either to hard-to-plan systems or to a low usage. Moreover, if the execution time misbehaviours of the reasoning tasks are not restricted, these cannot be integrated in real-time conventional systems, since these misbehaviours can alter the predictability properties of the initial system. The misbehaviour of the execution time for the problem-solving tasks develops itself on two levels: the methodological one and the one related to the architecture of the system. In order to ensure the predictability of a real-time AI system it is necessary to approach it on its both levels.

Knowledge-based Systems or I(K)MS(F)LE are characterised by the combination of some multiple models and some reasoning techniques (figure 1). Furthermore, these systems have certain common features. For applications there are necessary different models and problem-solving methods, their choice having a great impact on the efficiency and analysis of the knowledge-based system. Another common characteristic is the separation of the knowledge used in the way they are implemented, i.e. there is a distinction between the „knowledge level" and the „symbols level". These systems often combine multiple representations of knowledge, problem-solving strategies, as well as learning methods within the same system. The distinction between knowledge and representation is important, not only in the design but also in the analysis of the knowledge-based systems. When it comes to designing, the distinction corresponds to the two stages in the development process of such a system: the acquisition of a quantity of knowledge sufficient to solve the advanced problem and the creation of a system of programmes able to process this knowledge. Problem-solving often involves different types of knowledge, relative to the tasks of the problem (that define the problem), methods (referring to the way in which the problem is solved) and models (relatively to different possible behaviours of the system). This last type of knowledge is often referred to as domain knowledge, while the first two are called control knowledge. Knowledge-based systems invariantly include all these types of knowledge, even though they are not explicitly represented. Knowledge can be also be divided on multiple abstraction levels, each level offering details relatively to the problem that is to be solved.

If computers achieve consciousness and some type of advanced evolutionary level on their own, humans could end up using the computer as a medium for human perpetuation and development. If they develop a nervous system that is functionally similar to the human brain, humans may be able to input (download) into the computer their memories, thoughts, feelings, and sense of personal identity. During the last decade, there was a transfer from data-based information systems to information systems based on processes. Business Process Management includes methods, techniques and tools to support design, use, management and analysis of operational business processes involving people, organizations, applications, documents and other sources of information. Business Process Management can be considered an extension of classical systems based on workflow management. A business process is a set of activities that follows a logical flow, resulting an output. A business process is to specify the tasks to be fulfilled to achieve a business objective. Inputs and outputs may be facts and / or information, and the transformation can be performed by human actors, machines, or both. Although an organization's business processes can be tracked separately, by integrating them is obtain value-added which, long term, leads to positive results, a good control over the resources involved and over the environment in which it operates. Petri networks are tools of business process modeling.

One of the advantages of using Petri networks for modeling workflows is access to many analysis techniques based on Petri networks. Accuracy, applicability and efficiency of business processes supported by systems based on workflow management are vital to any organization. It is important to review the definition of a process through network flows before applying it in practice. In theory, there are three types of analysis: (1) validation, testing whether a workflow behaves as expected; (2) verification, determining the accuracy of the network flow; (3) performance analysis, evaluating the ability to meet the requirements, taking into account the time, levels of work and resources. Validation can be done through interactive simulation: are analyzed a number of cases to see if the system behaves properly. For verification and performance analysis, are used advanced analysis techniques specific to Petri networks, such as invariants, reachability trees, coverage graphs analysis, etc. The multitude of available analysis techniques shows that Petri networks can be seen as an independent environment (solution) between the design of a process' definition through workflows and the analysis of the resulted workflow.

Another important direction in the synthesis of intelligent KMSs in economy is the development of heterogeneous computing environments, in which one can develop adaptive algorithms for planning and resource allocation using genetic algorithms and evolutionary computation technology [18]. Heterogeneous computing environment is a suite of processing units, high-speed interconnections, interfaces, operating systems, communication protocols, and programming environments. It provides a variety of architectural options and skills that can be adapted to perform an application in accordance with different execution requirements. Different parts of an application task often require different types of calculation. It is generally impossible for a single architecture along with the compiler, operating system and associated programming tools to equally meet the computational requirements in such an application. In the heterogeneous computing environment considered, an application task can be decomposed into subtasks, each subtask being computationally homogeneous (appropriate, indicated for a single processing unit) and different subtasks may have different architectural requirements. These subtasks may have data dependencies between them. If the application task is decomposed into subtasks, the following decisions must be taken: subtasks' distribution on different processing units and scheduling, ordering subtasks' execution for each processing unit, as well as planning data transfer between processing units. In this context, the aim of heterogeneous compilation is to obtain a minimum duration of execution, i.e. the minimum total execution time of the application task in the suite of processing units.

It is well known that such an allocation and planning problem is generally NP-complete. It was proposed a large number of approaches to different aspects of this problem. This chapter proposes an approach based on genetic algorithms to solve the problem of concurrent allocation and planning in heterogeneous systems from the computational point of view. It decides the subtasks' allocation to processing units, orders the execution of the subtasks allocated to a processing unit and coordinate the data transfer between subtasks. The characteristics of this approach include: separation of allocation and planning representation, the independence of chromosomes' structure from communication subsystem's details and an assessment of the coverage degree and overlap between all communications and calculations that meet the precedence constraints of subtasks.

4. Conclusions

The importance of the paper resides in the demonstration of the possibility to use symbolic intelligent systems in management and process planning problems (technological, economic) using imprecise knowledge. On the other hand, it has been constantly tried to use some formalisms and concepts specific to either traditional or modern AI, with a series of models and techniques specific to Automatics, especially for the process management application. From this point of view, it was constantly necessary to adapt some known models (for example, the possibility theory, discrete events systems) in order to synthesise a management structure based on fuzzy knowledge. We also tried to create a conceptual opening towards a real management structure of multi-agent type, which is a solution to respond to a series of exigencies regarding the complexity of the management act. This type of systems, especially those based on a memory sharing communication between the agents, has well adapted features for real-time applications, such as:
i) Integrating heterogeneous agents; ii) The interaction between the acquisition, reasoning and action activities over the outward environment; iii) The fusion of data that come from nature translators and with different functioning; iv) The suppleness and efficiency in the integration of new data necessary for the reasoning process by simple writing in the common memory; v) The access facility of an agent to a piece of information which he needs; vi) The efficacy of control structures of reasoning; vii) The focus of the agent's functioning over a precise action due to some events, that takes into account the major amendments that interfered in the problem-solving process. Over the last years the interest regarding formal logic systems with applications in symbolic AI systems has considerably grown. It is the case of fuzzy and temporal logics that allow a more deepened understanding of the mathematical means that support the design of AI systems used in the decision-making process, in situations in which the information about the undergone process are partially known, incomplete and time variable. Real-time applications based on AI techniques need the cooperation of some reasoning elaborated processes. The essential point in the integration of cognitive/ reactive aspects is represented by the modelling of the relations between the evolutions of a process with certain inferential methods, which allows the closed-loop system to bear a series of entailed performances. The closed-loop system starts from an initial state and allows the planning process of a number of states, in order to reach the desired final state. At the end of its functioning, the expert management system provides the human governor with the possible actions (in specific conditions of the problem). Temporal aspects represent an important dimension of an expert management system. From this point of view there is a distinction between real-time reasoning and reasoning over time, the latter being a characteristic specific to knowledge-based systems. That is why it was absolutely necessary to explain the significance of time at a designing and implementation level of our intelligent system (agent), highlighting its implications in the concrete case of each concrete application.

References

1. *S. Asaoka, H. Maeda, S. Murakami, (1993), Dynamical Fuzzy Reasoning and Its Application to System Modeling, The Fith IFSA World Congress, p. 104-107*
2. *J.F. Baldwin, (1886), Inference for information systems containing probabilistic and fuzzy uncertainities, Fuzzy Logic for the Management of Uncertainty, Edited by Lotfi Zadeh and Janusz Kacprzyk, Wiley Professional Computing*
3. *L. Bolc, J. Cytowski, (1992), Search methods for AI, Academic Press Inc., ISBN 0-12-111240-03*
4. *B. Bouchon-Meunier, (1992), Fuzzy logic and knowledge representation using linguistic modifiers, Fuzzy Logic for the Management of Uncertainty, Edit. Lotfi Zadeh and Janusz Kacprzyk, Wiley Professional Computing, p. 399-414*
5. *T. Bylander, (1994), The computational complexity of propositional STRIPS planning, AI, nº 69, p. 165-204*
6. *M. Cadoli, F.M. Donini, P.Liberatore, M. Schaerf, (1996), Feasability and unfeasability of off-line processing, Proceedings of the 4th Israeli Symposium on Theory of Computing and Systems, p. 100-109*
7. *J. Cano, M. Delgado, S. Moral, (1993), An Axiomatic Framework for Propagating Uncertainty in Directed Acyclic Networks, International Journal of Approximate Reasoning, Vol. 8, nº 4, p. 253-280*
8. *C.L. Chang, C. Lee, (1973), Symbolic Logic and Mechanical Theorem Proving, Academic Press, ISBN 0-12-170350-9*
9. *V. Cutello, J. Montero, 1993), A Characterization of Rational Amalgamation Operations, International Journal of Approximate Reasoning, Vol. 8, nº 4, p. 325-344*
10. *H.P.Chen, T.M. Parng, (1996), A new approach of multi-stage fuzzy logic inference, Fuzzy Sets and Systems, Vol. 78, nº 1, feb. 26, p. 51-72*
11. *N. Cat Ho, H. Rasiowa, (1987), Semi-Post algebras, Studia Logica, 46, p. 147-158*
12. *N. Cat Ho, H. Rasiowa, (1987), Plain semi-Post algebras as a poset-based generalization of Post algebras and their representability, Studia Logica, 48, p. 509-530*
13. *D. Dubois, J. Lang, H. Prade, (1991), Fuzzy sets in aproximate reasoning - Part 2: Logical approaches, Fuzzy Sets and Systems, nº 40, p. 203-244*

14. D. Dubois, J. Lang, H. Prade, (1992), Inconsistency in possibilistic knowledge bases: To live with it or not live with it, Fuzzy Logic for the Management of Uncertainty, Edited by Lotfi Zadeh and Janusz Kacprzyk, Wiley Professional Computing, p. 335-351
15. D. Dubois, H. Prade, (1989), A typologyof fuzzy "if...then..." rules. Proc. 3rd International Fuzzy Systems Associations (IFSA) Congress ,Seattle, WA 6-11 August, p. 782-785
16. J.A. Drakopoulus, (1995), Probabilities, possibilities, and fuzzy sets, Fuzzy Sets and Systems, Vol. 75, nº 1, 13, p. 1-17
17. B.R. Gaines, (1977), Foundations of fuzzy reasoning, in Fuzzy Automata and Decision Processes, Amsterdam: North-Holland, p. 19-75
18. S. Gottwald, (1993), Fuzzy Sets and Fuzzy Logic/Foundations of Application - from a Mathematical Point of View, Braunschweig; Wiesbaden: Vieweg, ISBN 3-528-05311-9
19. J. P. Haton, et. col., (1991), Le raisonnement en intelligence artificielle, InterEditions, Paris
20. T. Hrycej, (1993), A temporal extension of PROLOG, The Journal of Logic Programming, Vol. 15, nº 12, p. 113-142
21. E.E. Kerre, (1992), A comparative study of the behavior of some popular fuzzy implication operators on the generalized modus ponens, Fuzzy Logic for the Management of Uncertainty, Edited by Lotfi Zadeh and Janusz Kacprzyk, Wiley Professional Computing, p. 281-296
22. R. Kruse, J. Gebhardt, F. Klawonn, (1994), Foundations of Fuzzy Systems, John Wiley&Sons, ISBN 0-471-94243-X
23. S. F. Klawonn, V. Novák, (1996), The relation between inference and interpolation in the framework of fuzzy systems, Fuzzy Sets and Systems 81, p. 331-354
24. J.F. Knight, K.M. Passino, (1987), Decidability for temporal logic in control theory, Proceedings of the Twenty-Fifth Allerton Conference on Communication, Control, and Computing, University of Illinois at Urbara-Champaign
25. R. Kruse, E. Schwecke, J. Heinsohn, (1991), Uncertainty and Vagueness in Knowledge Based Systems, Springer-Verlag, ISBN 0-387-54165-9
26. R. Lassaigne, M. Rougemont, (1996), Logique et complexité, Hermes, Paris
27. G. F. Luger, W. A. Stubblefield, (1993), AI, Structures and Strategies for Complex Problem Solving, Second Edition, The Benjamin/Cummings Publishing Company, Inc., ISBN 0-8053-4780-1
28. J. McDermott, (1988), Preliminary Steps Towards a Taxonomy of Problem-Solving Methods, in Automating Knowledge Acquisition for Expert Systems, S. Marcus, ed., Kluwer, Boston, p. 225-255
29. F. M. McNeill, E. Thro, (1994), Fuzzy Logic / A Practical Approach, Academic Press, ISBN 0-12-485965-8, Foreward by Ronald R. Yager
30. J.P. Müller, (1996), The design of intelligent agents: a layered approach, Lecture Notes in AI (Subseries of Lecture Notes in Computer Science), nº 1177, ISBN 3-540-62003-6
31. B. Nebel, C. Bäckström, (1994), On the computational complexity of temporal projection, planning and plan validation, AI 66, p. 125-160
32. V. Novák, (1992), Fuzzy logic as a basis of approximate reasoning, Fuzzy logic for the management of uncertainty, Edited by Lotfi A. Zadeh, Janusz Kacprzyk, John Wiley&Sons Ltd., p.247-264
33. Da-qun Qian, (1992), Representation and use of imprecise temporal knowledge in dynamic systems, Fuzzy Sets and Systems, Vol. 50, nº 1, Aug. 25, p. 59-77
34. J. Paul, A. Acharya, B. Black, J. K. Strosnider, (1993), Reducing Problem-Solving Variance to Improve Predictability, Advances in Real-Time Systems, IEEE Computer Society Press, p. 739-752, ISBN 0-8186-3792-7
35. E.I. Post, (1905-1970), Introduction to a general theory of elementary propositions, American Journal of Mathematics, 43, p. 165-185
36. H. Rasiowa, (1992), Toward fuzzy logic, Fuzzy Logic for the Management of Uncertainty, Edited by Lotfi Zadeh and Janusz Kacprzyk, Wiley Professional Computing, p. 5-25
37. H. Rasiowa, R. Sikorski, (1962), A proof of the completeness theorem of Gödel, Fundamenta Matematicae, 37
38. R. Turner, (1984), Logics for AI, John Wiley & Sons
39. R.R. Yager, (1992), Expert systems using fuzzy logic, An introduction to fuzzy logic applications in intelligent systems (edited by R. Yager, L. A. Zadeh), Kluwer Academic Publishers, p. 27-44