Modeling of Reasoning a Cognitive Agent with Significant Time Restrictions

1st Igor Fominykh
National Research University "MPEI"
111250 Moscow, Russia
igborfomin@mail.ru

2nd Alexander Eremeev
National Research University "MPEI"
111250 Moscow, Russia
eremeev@appmat.ru

3rd Nikolay Alekseev
National Research University "MPEI"
111250 Moscow, Russia
alekseevnp@list.ru

4th Natalia Gulyakina
Belarusian State University of Informatics and Radioelectronics
220013 Minsk, Belarus
guliakina@bsuir.by

***Abstract*—The problems of developing logical system to model the reasoning of cognitive agent are faced. Such an agent should be able to make conclusions based on its knowledge and observations in solving problems in the case of hard real-time. The hard real-time is characterized by setting a critical time threshold that considerably influences on agent's problem-solving.Exceeding this time threshold can lead to serious, sometimes catastrophic consequences and it is unacceptable for the agent. The formal basis of the modeling system is a logical system that integrates the concepts of active temporal logic and logical programming. Among the original approaches and methods proposed by the authors the following one should be mentioned. An integrated logical system that combines the concepts of active logic and logical programming has been built.. An approach to constructing a paraconsistent declarative semantics based on the concept of active logic has been introduced. The method of representing agent's temporal non-monotonous reasoning by active temporal logic has been proposed. The temporal granulation technique in logical system to formalize meta-reasoning has been suggested. Taking into account that the agent has to make decisions under the lack of time. the problem of the decision quality arises. In this context, it is useful to take branching time logics that allows us to infer various consequences of agent's decision. A subclass of such logics oriented to real-time systems applications has been considered. In general, the proposed methods and algorithms provide the conceptual and algorithmic bases for developing new generation intelligent systems able to function in the case of hard real-time.**

***Keywords*—cognitive agent, hard time constraints, active logic, step theory, temporal reasoning, logical programming**

I. Introduction

Various versions of active logic have been proposed for modeling reasoning in hard real time [1]- [4], which make it possible to observe the agent's reasoning process during its implementation. The creators of active logic emphasize its fundamental difference from traditional non-monotonic systems, such as default logic, auto-epistemic logic, etc. At present, there are dozens of different temporal logics, the purpose of which is to formalize the reasoning about time. The process of reasoning thus occurs as if out of time: the world as if stops while the system "thinks". For hard real-time systems, when solving problems, it is important to be able to estimate the amount of time available to them "to think" until it is too late to think. To do this, it is necessary to be able to correlate the steps and results of the conducted reasoning with the events occurring in the external environment. This type of reasoning is called reasoning situated in time. The General concept of active logic is described in [1]. As a model of deduction, active logic is characterized by language, many deductive rules, and many "observations". Reasoning situated in time is characterized by performing cycles of deduction called steps. Since the active logic is based on a discrete model of time, these steps play the role of a time standard – time is measured in steps. Agent knowledge is associated with the index of the step at which it was first obtained. The principal difference between active logic and other temporal epistemic logics is that temporal arguments are introduced into the language of agents own theories. A common drawback of most systems of Active Logic is the interpretation of time, in a sense, as the internal essence of these systems, the course of which is determined by the structure of the rules of inference used to obtain new formulas from existing ones. In all cases, the measure of time (standard) implicitly refers to the duration of the deductive cycle (=output step). Each execution of the deductive cycle corresponds to one "tick" of the virtual internal clock. It is also implicitly assumed that the duration of execution does not change from cycle to cycle, or that the changes are so small that they can be ignored. In reality, the duration of the deductive cycle is influenced by changes in the composition and structure of knowledge as a result of ongoing reasoning and

This work was supported by the Russian Foundation for Fundamental Research (RFFR) and Belarusian Republican Foundation for Fundamental Research (BRFR) (projects 18-51-0007 Bel_a, 20-57-00015 Bel_a)

observations of the external environment. In addition, the duration of deductive cycles can be influenced by random factors, such as power failures, in the operation of other technical systems, etc. in fact, the assumption of a constant duration of deductive cycles is akin to logical omniscience [5] and, like the latter, it often conflicts with reality. The report presents an approach in which time is treated as an external entity that is not related to the structure of knowledge and the speed of deductive cycles. Moreover, we propose a logical system (extended step theory [6], [7]) that integrates the concepts of active logic and logical programming, which allows us to optimize the relationship between the expressive capabilities of a cognitive agent and the complexity of calculations.

II. Main Results

Further, the main results in the field of modeling reasoning of a cognitive agent in the "hard" real-time mode, obtained by the authors to date, are considered.

An analytical review of studies has been carried out, including studies on the capabilities of existing logical systems for formalizing reasoning (meta-reasoning) of a cognitive agent in hard real time; by methods of granulation of time, methods of solving the problem of logical omniscience.

In the field of formalizing reasoning with limited time resources, there are several different directions, each of which is to one degree or another connected with solving the problem of logical omniscience, without which the conduct of reasoning (meta-reasoning), strictly limited in time, is not correct. Moreover, within the framework of existing epistemic logics, various restrictions on the ability of cognitive agents to logical inference (rational behavior) were proposed. This allowed us to solve the problem of logical omniscience, but at the same time there was a significant decrease in the capabilities of the cognitive agent, whose behavior was modeled. Today, only a few approaches to the creation of logical systems are known in which the problem of logical omniscience is solved without a serious limitation of the “mental abilities” of agents. One such example is the epistemic logic proposed by D. Ho [8]. It is based on the idea of introducing into the logical language special modal operators interpreted as “mental efforts” necessary to obtain any knowledge expressed in the form of logical formulas. This system overcomes the problem of logical omniscience. but it does not allow modeling the reasoning of a cognitive agent when it is necessary to determine whether it is able to solve a problem without going beyond certain time boundaries. Another example is active logic created by a team of specialists from Active Logic Grupp, and the like) [1]- [4]. It is a fairly general concept, which meets the logic presented in the report, a system based on the interpretation of reasoning as a process that proceeds in time. For this purpose, a temporal parameter is introduced into the metalanguage of logical systems that meet this concept. However, today there are a number of problems associated with active logic and other similar systems that hinder its actual practical application. Among the most important, it is necessary to highlight the absence of logical systems that meet the concept of active logic and have paraconsistent semantics, which makes it difficult to use active logic systems if there are contradictions in the information available; the lack of estimates of the computational complexity of reasoning and meta-reasoning, which are formalized by systems that meet the concept of active logic; lack of research results regarding the completeness and semantic consistency of systems that meet the concept of active logic. The logical system proposed in this report is largely free from these shortcomings due to the integration of formalisms of active logic and logical programming implemented in it.

The concept of time granulation as a special case of information granulation is developed, and like information granules, this representation of time in the form of granules - indistinguishable objects. Formally, granules can be represented as a neighborhood of points, intervals, fuzzy sets, etc. The concept of “granulation of time” was first introduced in the formalism of TLC (Temporal Logic with Clocks) [9]. In the Active Logic formalism, time granulation is introduced by analogy with TLC and reduces to the fact that the duration of deductive cycles, assumed constant in classical active logic, is not performed for hard real-time systems.

In reality, the duration of the deductive cycle is influenced by changes in the composition and structure of the agent’s knowledge as a result of his reasoning and observations of the external environment. In addition, random factors, such as power outages, other technical systems, etc., may affect the duration of deductive cycles. Also, “thinking abilities”, in this case, the duration of the computational cycles of different agents, ceteris paribus, can be different. To simulate the possibility of changing the duration, we propose a modification of the classical active logic - Step Logic (a lot of rules with a binary preference relation specified on it), which provides these capabilities by assigning the so-called. hours of model run, simulating the behavior of the system in various conditions (runs). A model run clock is a finite or infinite strictly increasing subsequence of a global clock whose members are interpreted as time instants (on a global clock) of the completion of deductive cycles, for example, <3, 5, 7, 10, ...>. By changing the model’s running hours, it is possible to simulate various operating conditions of the system and better reflect, for example, features such as an increase in the duration of deductive agent cycles as the amount of information known to him increases or in connection with the failure of part of his computing resources.

The concepts of metacognition, counting the time spent on conducting reasoning, paraconsistency of agent metareasoning based on active logic formalisms are developed.

The term metacognition was proposed in the works of J. Flavell [10] and is defined by him as the individual's awareness of his cognitive processes and related strategies, or, in other words, as reasoning about reasoning, meaning "cognition second order". The difference between cognitive and metacognitive strategies should be noted. The former helps the individual to achieve a specific cognitive goal (for example, to understand the text), and the latter are used to monitor the achievement of this goal (for example, self-inquiry for understanding this text). Metacognitive components are usually activated when cognition fails (in this case, it may be a misunderstanding of the text from the first reading). Such failure activates metacognitive processes that allow the individual to correct the situation. Thus, metacognition is responsible for the active control and sequential regulation of cognitive processes. The concept of "metacognitive cycle" was proposed by M. Anderson [11] in the context of using the principles of metacognition to improve resistance to anomalies of a rational agent with a limited time resource. It is defined as the cyclical implementation of the following three stages: self-observation (monitoring); self-esteem (analysis of the revealed anomaly), self-improvement (regulation of the cognitive process). At the self-observation stage, meta-reasoning comes down to checking for the presence in the argument of an agent that solves a certain problem, formal signs of the presence of anomalies. These formal features are often direct contradictions in the agent's knowledge (the presence of a counter pair of formulas expressing the agent's current knowledge). At the self-assessment stage, the degree of threat to the quality of the agent's functioning that the identified anomaly bears is established, and at the self-improvement stage, if the threat is real, a new strategy for solving the problem faced by the agent is selected.

The countdown is achieved using the special predicate now (.) Introduced in the rules. Moreover, now (t) takes the value "true" if and only if t is the time moment of completion of the last of the deductive cycles already completed, that is, in other words, when t is the current time. At the same time, the time counting principle used in this project is free from the unrealistic assumption of a constant duration of deductive cycles inherent in other existing approaches to solving the problem of modeling metaraguments. As you know, logic is called paraconsistent if it can be the basis of conflicting, but not trivial theories. In turn, a contradictory theory is a logical theory in which a certain proposition and its negation are simultaneously provable, and a trivial theory is a contradictory logical theory in which any proposition is provable or formally: for any formulas A and B,$\{A, \neg A \vdash B\}$. As a result of the analysis, it was found that the paraconsistency of classical active logic has not yet received a theoretical justification. At the same time, it was shown that the proposed step theory, based on the integration of the concepts of active logic and logical programming, is paraconsistent in the sense that the existence of contradictions in step theories does not lead to their destruction, as is the case in standard logical systems.

A method has been developed for evaluating the time resource available for a cognitive agent based on the proposed logical system.

It seems obvious that for agents with a strictly limited time resource, it is impossible to control this resource without correlating the results obtained in the course of the cognitive process (the process of solving the problem) with the times when these results were obtained. In accordance with the concept of step logic, this process, which proceeds in time, is characterized by the execution of deduction cycles (output steps).

As noted earlier, the time is counted using the special single predicate now (.). The following inference rule applies to this predicate:

$$\frac{t : now(t)}{t+1 : now(t+1)}, \tag{1}$$

moreover, the $now(t)$ formula is not inherited at the time instant (at the output step) t + 1, as is the case with "ordinary" formulas due to the inference rule

$$\frac{t : A}{t+1 : A}, \tag{2}$$

Also, a formula of the form $resource(t)$ is not inherited, obtained using the following inference rule, which allows you to evaluate a temporary resource at any time:

$$\frac{t : resource(t_1)}{t+1 : resource(t_1-1)}, \tag{3}$$

where t_1 is the time resource available to the agent at time t.

An important feature of step logic systems is the introduction of temporal parameters into the metalanguage, which determines their operational semantics, and the output steps play the role of a temporary reference. Agent knowledge is associated with the index of the step at which it was first acquired. This illustrates the inference rule, which is the "active" analogue of the modus ponens rule:

$$\frac{t : A, A \vdash B}{t+1 : B}, \tag{4}$$

This rule "says" that if at an instant t, an agent derived from reasoning or obtained from observing the external environment formulas A and $A \rightarrow B$, then at time t +1, formula B will be derived.

Moreover, the assumption that the duration of deductive cycles is always the same is unfair. The time moments of completion of the output steps form a sequence (“clock”), which is a subsequence of a sequence of natural numbers, for example, clock = <1, 3, 5, 7, 10, 14, ...>.

Methods have been developed to control the intermediate results and the time of their receipt on the basis of the proposed logical system. In the conditions of severe time constraints, it is extremely important to control the course of the reasoning process, primarily identifying the anomalies that arise. To do this, it is necessary for the agent to be able to realize not only what he knows at a given moment in time, but also what he does not know at this moment. In accordance with the concept of step logic, such an ability (which can be called self-cognition) is achieved thanks to two rules of inference:

$$\frac{t : C}{next(t) : K(t, C)}, \frac{t : C, sub(A, C), [A]}{next(t) :\vdash K(t, A)}, \tag{5}$$

where A is any formula that is not known to the agent at time t, but is a subformula of some known formula C, i.e. recognized by the agent, $sub(.,.)$ is a double meta-predicate expressing the relation “to be a subformula”, $[A]$ is a notation meaning that the formula A is absent in the agent’s current knowledge at time t. $K(.,.)$ Is a double meta-predicate (and not a modal operator!), Expressing the fact that the agent knows some formula at some point in time.

The above rules are used in order to be able to compare the current state of knowledge of the agent with those expectations that he had regarding the reasoning process carried out by the agent. If these expectations conflict with reality, this indicates a possible anomaly and the need to take measures to eliminate its consequences.

The syntax and declarative semantics of the language of the logical system are developed, combining the concepts of active logic and logical programming (LP) - the Extended Stepping Theory. The theory got its name by analogy with the extended logical programs introduced by A. Lifshitz and M. Gelfond [12] as applied to the logic programming paradigm. The language of this logical system includes two types of negation. One of them corresponds to the usual ("strong") logical negation, while the second, called "subjective", in a sense is similar to the default negation (negation as failure) in the LP, but has the following important difference. While in LP the meaning of negation by default lies in the fact that the negated formula (in the LP is always a literal) could not be deduced using the given logical program, the subjective negation in the considered logical system means that the negated literal could not be deduced by the current moment in time. Thus, in the system under consideration, the principle of self-knowledge is implemented, which consists in the fact that an agent whose behavior is modeled by a logical system is able to recognize and express explicitly not only what he knows, but also what he does not know at the moment. Note that such an opportunity is especially in demand when managing the cognitive process in the conditions of severe time constraints. This allows you to make managing the process of solving the problem more efficient compared to using other existing meta-reasoning formalisms in which this principle is not implemented.

Extended step theories are pairs of the form $T = (R, Ck)$, where R is a finite set of named rules, Ck is a clock of step theory, which is a finite subsequence of a sequence of natural numbers. The members of this subsequence characterize the duration of sequentially performed deductive cycles that determine the process of reasoning in all systems of active logic. At the same time, in this project, in contrast to classical active logic systems, the original principle of time granulation is used, which is implemented using the concept of clocks of step theories. The latter allow one to take into account the difference in the duration of deductive cycles and increase the temporal sensitivity of the step theory (i.e., the dependence of the results of the argument on how quickly the available time resource is spent). At the same time, it is interpreted as an external entity, independent of the internal structure of the set of rules of step theory, while in early versions of active logic systems, time is rigidly tied to the internal structure of the knowledge base used, which is a drawback.

The properties of completeness and correctness of the declarative semantics of a logical system for formalizing meta-reasoning in relation to various existing semantics of logical programs, including the semantics of stable models, the semantics of response sets, and others are investigated. The conditions and restrictions imposed on the language of the logical system for formalizing meta -reasoning, in which its semantics is correct and / or complete in relation to the semantics of logical programs listed above. The concepts of correctness and completeness of two semantics are refined using the relationship of logical sequence. It has been established that the declarative semantics of the logical system are correct and incomplete with respect to all the semantics listed above, except when the extended logical program is stratified. It was also established that the declarative semantics of a logical system are not only correct, but also complete with respect to the semantics of the set of answers of stratified extended logical programs.

The paraconsistency of the declarative semantics of the logical system for formalizing meta-reasoning is proved. The consistency of semantics informally means that the presence of contradictions in the theory does not lead to its destruction (i.e., it turns out to be trivial in a sense). The proof of the paraconsistency theorem for the declarative semantics of step theories with two types of negations is constructed similarly to the previously proved theorem on the consistency of semantics of step theories with only strong negation.

Studies have been carried out to determine the relationship of step theories of the logical system to formalize meta-reasoning and advanced logical programs. Formalisms of step theories with two types of negation and advanced logical programs were created for various purposes. The main difference between these formalisms is that in step theories, reasoning is treated as a process that develops over time, while from the point of view of the semantics of extended logical programs, reasoning is static. A comparison of the formalisms under consideration was reduced to a comparison of the logical sequence relations that they define and was carried out using some translation that transforms the rules of an extended logical program into the rules of step theory.

The temporal logic of branching time (branching temporal logic) can be used to solve the problems of training, forecasting and modeling in intelligent systems, when it is necessary to consider time branching into the future. The application of this logic allows us to simulate, as noted earlier, the possible consequences of the solution (or solutions) found by the agent under rather tight time constraints. In work [13], various temporal logics were considered in terms of their application in intelligent real-time systems (IRS). As a basis for use in the IRS, the recommended BPTL (Branching-Time Propositional Temporal logic), proposed in [14] and is an extension of propositional temporal logic (PTL). PTL is a modal temporal logic built on the basis of classical logic with added modal operators for discrete linear time.

The PTL syntax is defined as follows. The L_p PTL language is a countable set of propositional symbols $p, q, r, s, \ldots$ Formulas are constructed using the following symbols:

- A variety of propositional characters of L_p;
- Classic ligaments: $T, F, \neg, \wedge, \vee, \Rightarrow, \Leftrightarrow$;
- Temporal operators of the future tense: unary - $\boldsymbol{O}, \Box, \Diamond$; binary - U, W;
- Temporal operators of the past: unary - $\otimes, \bullet, \blacklozenge, \blacksquare$; binary J, Z;

A variety of well-formed formulas (wffs) PTL are:

- All propositional characters of L_p are wffs;
- If A and B are wffs, then
  $T, F; \neg A, A \wedge B, A \vee B, A \Rightarrow B, A \Leftrightarrow B$;
  $\boldsymbol{O}A, \Box A, \Diamond A, AUB, AWB$;
  $\otimes A, \bullet A, \blacklozenge A, \blacksquare A, AJB, AZB$ are wffs too;

The intuitive (informal) meaning of modal operators is as follows. Unary: $\boldsymbol{O}$ - next, $\bullet$ - last, $\otimes$ - last), $\Box$ - always in the future, $\blacksquare$ - always in the past, $\Diamond$ - sometime in the future, $\blacksquare$ - sometime in the past; binary: U - until, W - unless, J - since, Z - zince. If A and B are propositional formulas, then the intuitive meaning of modal formulas is defined as follows: "wff $\boldsymbol{O}A$ is true at the moment (in a given state) if wff A is true at the next moment; wff $\Box A$ ("always" A) is true at the moment, if and only if A is true at all future moments (states, including the current one); wff $\Diamond A$ ("eventually - finally" A) is true at the present moment if and only if A is true at some future moment.The strict wff until AUB is true at the moment if and only if the wff B is finally true, i.e. at the moment $s > n$, where n is the current moment, and the wff A is true for all moments t such that $n \leq t < s$. The operator W is a weak version of the operator U when it is not guaranteed that wff B is true at some future moment. Temporal operators of the past tense are defined as a strict version of the past tense for the corresponding operators of the future ("future" twins), i.e. the past does not include the present.

Semantics of PTL. To define the semantics of PTL, the semantics of the possible worlds of Kripke are used. A possible world is considered as a set of states in time, connected by temporal relations from a set of permissible relations R. Formally, the world is defined by a pair $A = (S, R)$, where S is a nonempty set of possible states, R is a binary relation, $R \subseteq S \times S$.

Considering L_p as a set of atomic statements, the model of the world can be defined as $M = (R, S, V)$, where V is the valuation function defining the map $V : S \times L_p \to \{T, F\}$, that is, calculating a propositional value for each state $s \in S$. By introducing various constraints on the relation R, various model structures are obtained. For example, if we introduce the restriction of antireflexivity (<), then we obtain a discrete model. For discrete linear models, the set S can be considered as a sequence of states, R - as the relation of following or successor. The interpretation is given by the pair $\langle M, i \rangle$. where M is the model and i is an integer indexing the states $s_i \in S$ in the model.

The semantics for temporal wff are defined using the relationship $\models$ between interpretation and wff. Thus, the statement $\langle M, i \rangle \models A$ means that the wff A is interpreted in the model M as a state with index i. The axiom system for linear PTL is consistent and complete. In a linear discrete PTL, the time model is an ordered sequence of natural numbers, i.e. each state has one and only one successor.

In the branching BPTL logic, a single successor is not necessary for each state and there can be many possible paths from any given state and, therefore, several different "future" ones are possible. The time model is an infinite tree, each vertex of which has a finite number of successors. The top of the tree is regarded as a possible state, and a branch or path is considered as the history of a possible world. The semantics of BPTL are defined in terms of the model structure $M = (S, R, V)$, where S, R and V are defined similarly to PTL. The concept of branching time requires the introduction of the linearity condition into the past and the transitivity of R. BPTL wffs are state formulas, and path formulas are auxiliary objects introduced in order to facilitate the determination of the semantics of state formulas. A system of axioms for BPTL is defined and BPTL is proved to be complete with respect to all branching time structures. Inference algorithms for BPTLs with a focus on IRS were proposed in [13].

On the whole, the obtained results create the necessary conceptual basis for constructing promising systems for modeling the reasoning of a cognitive agent that functions under strict time constraints by combining the concepts of active logic and logical programming.

Conclusion

The principal differences of active logic from traditional nonmonotonic logics such as default logic, auto-epistemic logic, etc. are formulated. (rejection of logical omniscience, the presence of temporal sensitivity and self-knowledge). The advantages of the step theory are formulated in comparison with other Active Logic systems (improved characteristics of computational complexity, paraconsistency, implementation of the principle of time granulation). The consistency of the step logic allows one to avoid the destruction of the entire system of reasoning, despite the presence of conflicting information. To further improve the management of the process of reasoning, formalisms of extended step theories are used, which differ from standard step theories by the introduction, along with a strong negation of subjective negation, which allows the cognitive agent to recognize and express explicitly not only what he knows, but also what he does not knows at the moment. This improves the expressive capabilities of the theory and, in particular, the property of temporal sensitivity.

The use of temporal logic of branching time is proposed, which allows modeling (deriving) various consequences of a solution found by an agent. A subclass of such logic, oriented to application in real-time systems, is considered.

The results can be used in the design of complex dynamic systems of hard real-time, including the design of control systems for vehicles (ships, aircraft), power systems, power plant units and their simulators.

References

[1] Elgot-Drapkin J. Step Logic: Reasoning situated in time. PhD thesis. Department of computer science, University of Maryland, Colledge-Park, Maryland, 1988

[2] Purang K., Purushothaman D., Traum D., Andersen C., Perlis D. Practical Reasoning and Plan Executing with Active Logic. Proceedings of the IJCAI'99 Workshop on Practical Reasoning and Rationality, 1999

[3] Perlis D., Purang K., Purushothaman D., Andersen C., Traum D. Modeling time and metareasoning in dialog via active logic. Working Notes of AAAI Fall Symposium on Psychological Models of Communication, 2005

[4] Hovold J. On a semantics for active logic. MA Thesis, Department of Computer Science, Lund University, 2011

[5] Vinkov M.M., Fominykh I.B. Discussions about knowledge and the problem of logical omniscience. Part 1. Modal approach. Artificial Intelligence and Decision Making. vol. 4, 2011, pp. 4-12 (In Russian)

[6] Vinkov M.M., Fominykh I.B. Stepping Theories of Active Logic with Two Kinds of Negation. Advances in electrical and electronic engineering. 2017, vol. 15, no. 1,pp. 84-92.

[7] Vinkov M.M., Fominykh I.B. Extended Stepping Theories of Active Logic: Paraconsistent Semantics. Proceedings of the 16th Russian Conference RCAI 2018 Moscow, Russia, September 24-27, 2018. Communications in Computer and Information Science v.934 Springer, pp.70-78, 2018

[8] D. N. Ho, Logical omniscience vs. logical ignorance. On a dilemma of epistemic logic. Progress in Artificial Intelligence. Proceedings of EPIA'95, LNAI, vol. 990, Springer Verlag, 1995, pp. 237-248

[9] C.Liu, M.A. Orgun. Verification of reactive systems using temporal logic with clocks. Theoretical Computer Science 220 (1999) 377—408

[10] Flavell, J. H. Metacognition and cognitive monitoring: A new era in cognitive- developmental inquiry. American Psychologist, 34(10), 1987. - pp. 906-911

[11] Michael L. Anderson and Tim Oates and Waiyian Chong and Don Perlis. The metacognitive loop I: Enhancing reinforcement learning with metacognitive monitoring and control for improved perturbation tolerance. Journal of Experimental and Theoretical Artificial Intelligence. 2006. 18. 3. 387-411

[12] Gelfond, M., Lifschitz, V. Classical Negation in Logic Programs and Disjunctive Databases. New Generation Computing 9, 365–385 (1991)

[13] Eremeev A.P., Kurilenko I.E. An Implementation of Inference in Temporal Branching Time Models. Journal of Computer and Systems Sciences International, 2017, Vol. 56, No. 1, pp. 105–124

[14] I.S. Torsun, Foundations of Intelligent Knowledge-Based Systems (Academic, London, 1998).

Моделирование рассуждений когнитивного агента при существенных временных ограничениях

Фоминых И. Б., Еремеев А. П.,
Алексеев Н. П., Гулякина Н. А.

Рассматриваются вопросы проектирования системы моделирования рассуждений когнитивного агента, способного на основе своих знаний и наблюдений за внешней средой делать умозаключения, решая задачи в режиме «жёсткого» реального времени. Для работы в таком режиме характерно существование критического временного порога, установленного для решения стоящей перед агентом задачи. Превышение порога чревато тяжёлыми, подчас катастрофическими последствиями и для агента является неприемлемым. Формальной основой системы моделирования является логическая система, объединяющая концепции активной темпоральной логики и логического программирования. Среди предлагаемых авторами в работе оригинальных методов следует отметить подход к объединению концепций активной логики и логического программирования в одной логической системе; подход к построению паранепротиворечивой декларативной семантики логической системы, имеющей в основе концепцию активной логики; метод формализации темпоральных, немонотонных рассуждений агента средствами активной темпоральной логики; метод грануляции времени в логической системе для формализации мета-рассуждений. Учитывая, что в системах жесткого реального времени агенту часто приходится принимать решения в условиях недостатка времени, то возникает вопрос о качестве найденного решения. В этом плане полезно использование темпоральной логики ветвящегося времени, позволяющей смоделировать (вывести) различные последствия найденного агентом решения. Рассматривается подкласс такой логики, ориентированный на применение в системах реального времени. В целом, предлагаемые методы создают концептуальные и алгоритмические основы для построения перспективных интеллектуальных систем жёсткого реального времени нового поколения.

Received 17.01.2020