SIMULATION MODELING IN THE CONSTRUCTION OF DYNAMIC INTEGRATED EXPERT SYSTEMS

Galina Rybina, Victor Rybin

Abstract: *Problems of integrating methods and simulation modeling tools with the technology of dynamic integrated expert systems (IES) developed on the basis of task-oriented methodology and AT-TECHNOLOGY workbench are investigated in the context of the modern stage of simulation modeling development. The current version of the simulation modeling subsystem, functioning as a part of the dynamic version of the AT-TECHNOLOGY workbench, is described. Basic subsystem components and the construction technology of simulation models of complex technical systems are considered, as well as examples of application of this subsystem in prototyping dynamic IES are provided.*

Keywords: *dynamic intelligent systems, dynamic integrated expert systems, task-oriented methodology, simulation, complex engineering systems, simulation modeling subsystem, simulation model, AT-TECHNOLOGY workbench.*

Introduction

Analysis of modern works in the scope of simulation modeling conducted in [1] shows a significant increase of interest in this area, which, experiencing a rebirth, is becoming more and more advanced computer simulation technology used in a wide range of new applications related to the control and decision-making of the technological, organizational, economic, social and other character in dynamic problem domains [2-5].

Today the integration technology with data mining (Data Mining) [6, 7], computer-aided design [8, 9], business information systems [10, 11] etc. are actively investigated in simulation modeling problems in addition to the traditional aspects (methodological, mathematical, technological and applied).

However, integration processes achieved the greatest topicality in dynamic intelligent systems (DIS) [12, 13], such as intelligent control systems [14, 15], intelligent decision support systems [16, 17], dynamic integrated expert systems [6, 12], multi-agent systems and intelligent agents [18-21], that connected with an constantly-expanded range of DIS application in modern postindustrial society - from the spacecraft and sophisticated automated manufacturing and robots to organizational and technical (socio-technical) systems with uncertain human factor.

It should be noted that the problems of integration of simulation modeling technology with dynamic integrated expert systems (IES) got the most complete development under task-oriented methodology and supporting software tools (AT-TECHNOLOGY workbench) [6]. Scientific and practical results in the theory and technology for constructing IES published in the monograph [6], as well as repeatedly described in various scientific journals [22-27], including the pages of some issues of this journal [28-30].

The focus of this article which is a continuation of the researches described in [6, 22-33] is technological and applied aspects connected with the expansion of the architecture of the dynamic version of the AT-TECHNOLOGY workbench by integrating the subsystem simulation modeling of the external world and the combined functioning of this subsystem with temporal solver and other basic workbench components in the development of dynamic IES prototypes.

Features of the Application of Simulation Modeling in the Construction of Dynamic IESS

As has been noted in a number of papers, for example [6, 22], long-term experience of practical use of task-oriented methodology and AT-TECHNOLOGY system has shown, that the applications in the which problem area is dynamic occur very often, and therefore a cycle of researches has begun and led to the creation of theory and software technology constructing dynamic IESs which operates in real time, i.e. IESs using dynamic representation of object domains and solving dynamic problems (main results were published in [1, 28-36]).

In the context of the further development of task-oriented methodology for other architectures of DIS, in particular, multi-agent systems also conducted researches related to the creation of simulation systems interaction of intelligent agents (IMVIA), which is currently used as a workbench developing the prototypes of multi-agent systems for dynamic problem fields (described in [12, 20] and other papers).

Most often, the most common areas of the application of dynamic IESs technology proved complex engineering systems (CES) and complex engineering and organizational systems (CEOS), for which the classical simulation focused on formal mathematical models was practically impossible to use. In [6] introduced a specific definition, under which CES and CEOS - the following objects of a technical nature: their parameters constantly vary (in real time); they comprise from several hundred to several thousand functionally and structurally interrelated components, subsystems, modules, units, etc.; the diagnostics of these objects can be considered as a specific control process with the goal of determining the technological state of objects at each current instant (the general task of diagnostics of the object status) and, in addition, the task of fault finding (as a special case of the general diagnostic task); the functioning of these objects is a complex technological process accompanied by a multitude of abnormal conditions, rapid changes in the environment, and the lack of time for decision-making in response to abnormal conditions; a high price is paid for errors made by operators.

Accordingly the dynamic IESs for discrete and continuous-discrete CES/CEOS must ensure, in the general case, support for the execution of the following tasks: the dynamic modeling of all processes of functioning the of CES/CEOS; monitoring the CES/CEOS operation, detection of deviations from the prescribed regime, pre-failure alerting and abnormal condition warning, emergency cut-out, etc.; studying the actions of the operators who control CES/CEOS and training of personnel; a convenient graphic user interface for monitoring variations in the basic parameters characterizing CES/CEOS operation, etc.

In the context of the use of the task-oriented methodology for constructing IESs [6], the additional functionality of described tasks entails a significant change of the IES architecture as all basic components of static IES are practically modified, especially, knowledge base and reasoning tools, and two new subsystems are added—subsystem modeling the external world (environment) and subsystem interfacing with the physical equipment, as well as the technology of constructing dynamic IESs is significantly changed. The subsystem interfacing with the external environment is necessary to obtain a constant data stream from external equipment and sensors, and the subsystem modeling the external world (environment) is intended to simulate the data stream at all stages of the life cycle of dynamic IES development.

In the context of this work, the subject of discussion is the subsystem modeling the external world, because the data that is transferred to working memory by the subsystem uses temporal and universal solvers of the AT- TECHNOLOGY workbench [6, 27, 31] to realize a deduction and to obtain recommendations. Basing on these objectives for computer simulation of the CES / CEOS behavior in time the simulation modeling concept using RAO-approach [37], which implements the process-oriented approach to construct simulation models (SM) achieved the most development and application. As the expansion experience of the AT-TECHNOLOGY workbench architecture by specialized tools in the form of the simulation modeling subsystem of the external world which is realized on the basis principles of the RAO-approach [31-36] has shown, this way was quite effective for deep integration of all components of the dynamic IES nucleus and combined functioning of the simulation modeling subsystem with the temporal solver, universal AT-SOLVER and other basic components of the dynamic version of AT- TECHNOLOGY workbench (individual results discussed in [1, 27-36]).

The basic principles of the simulation modeling subsystem implementation based on RAO-approach and task-oriented methodology requirements are considered in more detail.

General Characteristic of the Simulation Modeling Subsystem

- ✓ *Main Principles of the Construction of the Simulation Modeling Subsystem*

In the architecture of the simulation modeling subsystem, the functionality of the developed tools is divided between two global modules [6, 31] – the “SM development module” whose tasks are to support the development process and debugging of SM and other functions requiring the visual interface, and the "SM computation module" ensuring the computation of the conditions of SM in each time step (cycle) of the functioning process of the dynamic IES.

The development of a powerful full-featured high-level language to describe the SM and the creation of a corresponding compiler for that language are the unifying conceptual framework for the two basic modules. To implement this approach, at the first stage of the researches, formalism RAO is used as a language to describe the SM, the basic version of which is given in [37]. In the future, based on the analysis of current requirements to design models of CES / CEOS to create dynamic IESs developed a special language "RAOAT" including new conceptual changes associated with object -oriented language and significant technological expansions due to the addition of new instructions and data storage structures [31–36].

Let us briefly consider RAOAT language features. The basic concept of the creation of RAOAT language is its full object orientation, and therefore, such basic principles of object-oriented language, as encapsulation, inheritance and polymorphism are implemented. From the standpoint of encapsulation should be noted that into the language were entered modifiers for resource types that allow to describe the internal logic of resource types that change the state of the resource attributes according to some conditions (logical or temporary). Also entered the internal attributes of the resource types that are not available for changing by operation templates, that it significantly reduces the effort required in the design and development of SM containing a set of resource types that are represented by sets of an arbitrary number of instances by transferring the description of variation of resource conditions from the operation templates to the state modifiers of resource types.

Inheritance and polymorphism in the RAOAT language are represented with the ability to inherit one type of a resource from another and override the implementation of the state modifiers of a resource in the process of inheritance that can largely simplify the SM containing the sets of structurally similar resources.

It should be pointed out, that basic principles and implementation features of the several versions of simulation modeling subsystem in the context of the proposed approach have been repeatedly described in different papers, in particular [31–36]. In general, it should be noted that such basic requirements, as the ability to create new objects and resource types on the basis of finished, the ability of developing model base through the development of resource types library, the efficiency in developing and maintaining complex SM (hundreds or thousands of objects and resource types, well-developed logic), the suitable RAOAT language allowing to construct SM by specialists on simulation modeling rather than programmers and the efficient ways to integrate with third party software, were considered when developing the subsystem.

Another important feature is related to the “SM development module” ensuring the user (knowledge engineer and / or specialist on simulation modeling) by convenient visual editor that allows *operating* graphical objects and relationships between them within the concept of RAO-method. Meanwhile, a code generation of the model in RAOAT language occurs (established resource types, resource templates operations, etc.). Result of the visual design of the model is the description of the model in RAOAT language. The tools of this module allow you to construct a model of the outside world for dynamic IES using visual, flexible, extensible, repeated-used objects (resource types). The ability to increase the life cycle of SM by rapid adjustment to changing conditions, when solving of which require both high and low levels of abstraction, is achieved due to an unlimited nesting of resource types and inheritance .

Composition and structure of the “SM development module” and the “SM computation module” functioning in the structure of the current version of simulation modeling subsystem (dynamic version of the AT-TECHNOLOGY workbench) [1, 31-36] are briefly considered bellow.

- ✓ *Architecture, Composition and Structure of the Basic Components of the Simulation Modeling Subsystem*

General architecture, composition and structure are of the basic components of the current version of simulation subsystem, detailed description of which is given below.

Visual Objects Editor: The component “Visual objects editor” allows you to create objects, setting their properties and attributes, as well as establish relationships between the model objects in graphical mode. Knowledge engineer and / or specialist on simulation modeling, operating with the models editor, designs SM on graphic canvas containing a predetermined number of various objects, at that the ability to create, delete, copy objects and set relationships between them is ensured. The values of a particular set of properties are able to change for each object. The created visual representation of SM and properties of all objects are stored in the memory of the editor, as well as in a separate text file, which is further processed by the RAOAT language compiler. This visual tool allows you to load a saved SM to update the visual presentation and insert the changes manually to obtained code which is described in RAOAT language.

Models Synthesis Component: The models synthesis component, interacting with the visual objects editor by processing the stored collections of objects, generates a description of SM in RAOAT language in XML format which is passed to the RAOAT language compiler.

Component of Visualizing SM: Knowledge engineer, if necessary, with the support of the “Animation frames and displayed rules editor” selects the description of the model in RAOAT language and makes animation frames for corresponding objects, and the tool “Visualizer" based on the values of resource parameters, descriptions of animation frames and displayed rules ensures drawing the animation frames.

RAOAT Language Compiler: The obtained description of SM in RAOAT language is passed to the input of the “SM computation module” where the compilation from RAOAT language to C# language occurs and the further interpretation and run the developed model. The kernel of the “SM computation module” is the “RAOAT language compiler” which structure has a standard form for the syntax-driven three-pass compiler. This compiler consists of an analyzer which includes components of the lexical, syntactic and semantic analysis, and synthesis component including a generation component of output code. Features of the implementation of several versions of the RAOAT language compiler are described in detail in [31, 32, 34]. It should be noted that the availability of such objects in the RAOAT language, as irregular events and temporary resources, requires the time coordination with each object of the model. Each object has its own internal timer showing within the modeling time scale, how much time obtains an object. In addition, this timer is associated with the total time of an activity of the SM for a corresponding pause during transferring and obtaining the data from the working memory of the temporal solver.

Supporting Component of Computing the SM States: The “Supporting component of computing the SM states” ensures the generation of a discrete modeling time, as well as the generation of the control actions used to start or stop the activity of SM in the form of messages on each discrete time step. Computation of the new state of SM is based on the state at the previous time step and an allowance of executed operations.

Technology of Constructing Simulation Models Using Simulation Modeling Subsystem Tools

A technology for constructing SM described as five suitable interacting stages is considered in detail.

Stage 1. Description of CES/CEOS. First, the knowledge engineer and/or specialist on simulation modeling (expert) selects the simulation objects (resources) and resource parameters, and then the rules for interactions between resources (events) are determined and the detachment of events to regular and irregular is executed with the indication of the relevant event resources.

Stage 2. Constructing the description of SM in RAOAT language. A knowledge engineer and / or a specialist on simulation modeling constructs the description of RAOAT objects (resource types, resources, templates of operations, operations, functions), and a program control of the correctness of the description of RAOAT objects (object parameter blank, infringement of the logic of object, etc.) is realized. Each described object is added to the corresponding collections, and then models synthesis component generates a conversion of the object collections to the description of SM in the RAOAT format (saving in corresponding XML-structure is realized).

Stage 3. Compilation of SM in RAOAT language to the internal representation of C# language. With the help of RAOAT compiler generates a compilation of SM in RAOAT language of XML- structure. The description of SM in RAOAT language passes lexical analysis stage, result of which is the formation of object lists, then the stage of parsing, at which the resulting lists of objects are converted to the parse tree, and at the stage of semantic analysis realized the checking of correctness of the description of SM in accordance with the logic of RAOAT language. Further, the synthesis component of target text realizes a transformation of original description of SM to the code in C# language, and the result description is compiled into .dll.

Stage 4. Parameters configuration and computing of SM states

4.1. Configuration of SM parameters. The obtained SM in the form of .dll enters the input of the "Supporting component of computing SM states" to be realized the configuration of SM parameters using specialist on simulation modeling (time step length, number of time steps). After completing the configuration of SM parameters, SM is executed, and then control of the SM can be realized by stopping or pausing it.

4.2. Computation of SM states. Computation of the states of SM is executed based on the state of SM at the previous time step, with an allowance for the operations executed by SM. If the time step is not the last, then computing the states of SM to the next time step is realized on the basis of resource parameter values in the current tact. The activity of the component comes to complete after obtaining the last time step and the file which contains the values of resource parameters at all time-step is generated. In case of if the visual examination of the state of SM is not required, and result of the activity of the SM is suited by the specialists, then the first iteration is passed, i.e. the first version of SM is obtained. If the visual examination of SM is needed, then transition to the next stage 5 is realized.

Stage 5. Visualization of SM. The "Component of visualizing SM", which consists of two units – the "Animation frames and displayed rules editor" and the "Visualizer", is activated when the visual examination of the state of SM is necessary. Editor obtains a description of SM in RAOAT language as an input, and the knowledge engineer and / or a specialist on simulation modeling makes adding animation frames (the path to the image, width, height, etc.) and displayed rules that determine the behavior of the animation frames. Added animation frames and displayed rules are written in the corresponding XML- structure and passed to the input of "Visualizer", together with the values of resource parameters at all time-step for the graphical representation of SM. If the obtained graphical representation on the results of the activity of SM is suited by the specialists, then the first iteration is passed, i.e., the first version of SM is obtained.

Technological cycle is completed at this and the achieved version of SM in the form of .dll which is intended for further use in the process of prototyping of dynamic IESs is transferred to the working memory by special tools of AT- TECHNOLOGY workbench which obtain the SM in the form .dll ensure the synchronization of the processes of interacting between simulation modeling subsystem and reasoning tools (temporal solver and AT-solver). As the temporal solver, as well as the SM, operates on time step, the interaction is realized through the exchange of data and commands in asynchronous mode [35]. At the end of each time step of the activity of SM, data transferring to the solver through the working memory is realized. The obtained data initializes a new reasoning cycle. From now the solver has a certain time quantum allowing realizing a deduction for urgent situations. The resulting urgent commands (the deduction of which had time to execute within the time limit) are used at the next time step of SM. Not considered urgent and common situations are processed before the end of the next time step and used through a time step of the activity of SM. In case of obtaining new data from SM a deduction is processed and delivered an interim solution. The next time step of the activity of temporal solver does not depend on the place where the deduction of previous time step has broken down.

Conclusion

Thus, the further development and deep integration of the methods of simulation modeling with the methods of constructing dynamic IESs in the context of task- oriented methodology as a unified framework ensuring the complex decision for scientific and applied problems related to the construction of different classes of dynamic IESs, allowed to create an approach that does not have analogues today. This approach has been successfully implemented in the framework of the dynamic version of AT- TECHNOLOGY system and has already shown its efficiency in implementation of several prototypes of applied dynamic IESs.

Acknowledgements

This work was supported in part by the Russian Foundation for Basic Research under Grant No. 15-01-04696

Bibliography

1. Рыбина Г.В., Рыбин В.М., Со Ти Ха Аунг. Некоторые аспекты применения имитационного моделирования в динамических интегрированных экспертных системах // Приборы и системы. Управление, контроль, диагностика. 2014. № 3. С. 35-45.
2. Плотников А.М., Рыжиков Ю.И., Соколов Б.В. Современное состояние и тенденции развития имитационного моделирования в Российской Федерации // Пятая всероссийская научно-практическая

конференция по имитационному моделированию и его применению в науке и промышленности «Имитационное моделирование. Теория и практика». Сборник докладов. Т.1. – СПб.: ОАО «Центр технологии и судостроения», 2011. С.55-61.

3. Лычкина Н. Н. Инновационные парадигмы имитационного моделирования и их применение в сфере управленческого консалтинга, логистики и стратегического менеджмента // Современные технологии управления логистической инфраструктурой - III: Сборник научных статей, – М.: Изд-во Эс-Си-Эм Консалтинг, 2012. С. 9-28.
4. Averill M., Kelton W. Simulation modeling and analysis. 3rd ed. McGraw-Hill,Inc., 2000. – 760 p.
5. Banks J. Handbook of Simulation: Principles, Methodology, Advances, Applicationsand Practice. – Wiley, 1998. – 864 p.
6. Рыбина Г.В. Теория и технология построения интегрированных экспертных систем. Монография. – М.: Научтехлитиздат, 2008. − 482с.
7. Rybina G.V., Deineko A.O., Distributed Knowledge Acquisition for the Automatic Construction of Integrated Expert Systems // Scientific and Technical Information Processing. 2011.V. 38. № 6. P.1-7.
8. Таратухин В.В. Интеграция систем имитационного моделирования и экспертных систем в САПР // КИИ-98. Шестая национальная конф. с международным участием. Сб. научных трудов в 3-х томах. – Пущино: РАИИ, 1998. Т.2. С. 461-465.
9. Seidel S., Donath U., Haufe J. Towards an integrated simulation and virtual commissioning environment for controls of material handling systems //Proceedings of the 2012 Winter Simulation Conference. 2012. P. 2852 -2863.
10. Лычкина Н.Н. Современные технологии имитационного моделирования и их применения в информационных бизнес-системах и системах поддержки принятия решений // Вторая всероссийская научно-практическая конференция по имитационному моделированию и его применению в науке и промышленности «Имитационное моделирование. Теория и практика». Сборник докладов. Т.1. – СПб.: ФГУП ЦНИИ технологии судостроения, 2005. С. 26-31.
11. Pidd M. Computer simulation in Management Science. 5th ed. Chichester:Wiley, 2004. – 328 p.
12. Рыбина Г.В., Паронджанов С.С. Технология построения динамических интеллектуальных систем: Учебное пособие. М.: НИЯУ МИФИ, 2011. − 240с.
13. Осипов Г.С. Методы искусственного интеллекта. – М.: ФИЗМАТЛИТ, 2011. – 296с.
14. Макаров И.М., Лохин В.М., Манько С.В., Романов М.П. Искусственный интеллект и интеллектуальные системы управления. – М.: Наука, 2006. − 333с.
15. Рыбин В.М. Интеллектуальное управление на основе динамических интегрированных экспертных систем. // Информационно-измерительные и управляющие системы. 2011. №6. С. 16-19.
16. Вагин В.Н., Еремеев А.П. Некоторые базовые принципы построения интеллектуальных систем поддержки принятия решений реального времени // Известия РАН. ТиСУ. 2001. №6. С. 114-123.

17. Башлыков А.А., Еремеев А.П. Семиотические системы реального времени для интеллектуальной поддержки принятия решений при управлениями сложными технологическими объектами // Информационно-измерительные и управляющие системы. 2013. №5. С.49-57.
18. Тарасов В.Б. От многоагентных систем к интеллектуальным организациям: философия, психология, информатика. – М.: Эдиториал УРСС, 2002. − 352с.
19. Fougères A. Modelling and simulation of complex systems: an approach based on multi-level agents // IJCSI International Journal of Computer Science Issues, November 2011.Vol. 8, Issue 6, No 1, P. 8-17.
20. Rybina G.V., Parondzhanov S.S. Modeling of Intelligent Agent Interactions for Multiagent Systems // Scientific and Technical Information Processing. 2010. V. 37. № 5. P. 318-327.
21. Аксенов К.А., Смолий Е.Ф. Мультиагентное имитационное моделирование бизнес-процессов и организационно-технических систем, среда BPSIM // Пятая всероссийская научно-практическая конференция по имитационному моделированию и его применению в науке и промышленности «Имитационное моделирование. Теория и практика». Сборник докладов. Т.1. – СПб.: ОАО «Центр технологии и судостроения», 2011. С.11-19
22. Рыбина Г.В. Практическое применение задачно-ориентированной методологии построения интегрированных экспертных систем (обзор приложений в статических и динамических проблемных областях) // Приборы и системы. Управление, контроль, диагностика. 2011. № 12. С. 10-28.
23. Рыбина Г.В. Интегрированные экспертные системы // Приборы и системы. Управление, контроль, диагностика. 2011. №4. С. 22-47.
24. Рыбина Г.В. Технология построения интегрированных экспертных систем на основе задачно-ориентированной методологии // Приборы и системы. Управление, контроль, диагностика. 2011. №6. С. 28-42.
25. Рыбина Г.В. Теоретические основы задачно-ориентированной методологии построения интегрированных экспертных систем. // Приборы и системы. Управление, контроль, диагностика. 2011. №7. С. 27-42.
26. Рыбина Г.В. Модели и методы реализации интеллектуальной технологии построения интегрированных экспертных систем. // Приборы и системы. Управление, контроль, диагностика. 2011. №10. С. 27-37.
27. Рыбина Г.В. Инструментальный комплекс АТ-ТЕХНОЛОГИЯ для поддержки построения интегрированных экспертных систем: общая характеристика и перспективы развития. // Приборы и системы. Управление, контроль, диагностика. 2011. №11. С.17-40.
28. Рыбина Г.В., Демидов Д.В., Шанцер Д.И., Мозгачев А.В. Динамические интегрированные экспертные системы: новые возможности инструментального комплекса АТ-ТЕХНОЛОГИЯ // Информационно-измерительные и управляющие системы. 2011. №6. С. 7-15.
29. Рыбина Г.В., Мозгачев А.В., Со Ти Ха Аунг. Построение динамических интегрированных экспертных систем на основе задачно-ориентированной методологии // Информационно-измерительные и управляющие системы. 2012. №8. Т.10. С. 4-12.

30. Рыбина Г.В. Экспертные системы и инструментальные средства для их разработки: некоторые итоги // Информационно-измерительные и управляющие системы. 2013. Т.11. № 5. С. 35-48.
31. Рыбина Г.В. Инструментальные средства для построения динамических интегрированных экспертных систем: развитие комплекса АТ-ТЕХНОЛОГИЯ // Искусственный интеллект и принятие решений. 2010. №1. С. 41-48.
32. Rybina G., Shantzer D.I., Mozgachev A.V. Some aspects of evolution of tools for support dynamic integrated expert systems development // Proceedings of First Russia and Pacific Conference on Computer Technology and Applications (RPC 2010). 6-9 September 2010, Vladivostok (Russia). Vladivostok: IACP FEB RAS, 2010. P. 136-139.
33. Рыбина Г.В., Шанцер Д.И., Мозгачев А.В., Блохин Ю.М. Динамические интеллектуальные системы на основе интегрированных экспертных систем // Приборы и системы. Управление, контроль, диагностика. 2012. №5. С. 13-19.
34. Рыбина Г.В., Шанцер Д.И., Липатов А.А. Имитационное моделирование внешнего мира при построении динамических интегрированных экспертных систем// Тринадцатая национальная конференция по искусственному интеллекту с международным участием. Труды конференции. Т.3. – Белгород: Изд-во БГТУ, 2012.С.179-185
35. Рыбина Г.В., Мозгачев А.В., Паронджанов С.С., Со Ти Ха Аунг Динамические интегрированные экспертные системы: методы представления и обработки темпоральных знаний // Приборы и системы. Управление, контроль, диагностика, 2013. №6. С.63-73
36. Rybina G.V., Rybin V.M., Parondzhanov S.S., SoeThi Ha Aung. Some aspects of simulation application in dynamic integrated expert systems // Life Science Journal. 2014. V.11. №8s. P.145-149.
37. Емельянов В.В., Ясиновский С.И. Введение в интеллектуальное имитационное моделирование сложных дискретных систем и процессов. Язык РДО. − М.: АНВИК, 1998. – 533с.

Authors' Information

Galina Rybina *– Doctor of Technical Science, Professor cybernetics department of NRNU MEPhI. RF President education award winner. Accents on intelligent systems and technologies, static, dynamic and integrated expert systems, intelligent dialogue systems, multi-agents systems, workbenches; e-mail: galina@ailab.mephi.ru*

Victor Rybin *- Doctor of Technical Science, Professor department of Automation of NRNU MEPhI. Accents on automation and electronics, electro physical complex, automatic control system, intelligent control systems, dynamic intelligent systems; e-mail: vmrybin@yandex.ru*