The concept of intelligent tutoring for enterprise staff as a component of integrated manufacturing control system development

Rustem A. Sabitov, Gulnara S. Smirnova[1], Bulat. R. Sirazetdinov, Natalya. U. Elizarova, Alexander. V. Eponeshnikov

Kazan National Research Technical University named after A. N. Tupolev — KAI, Kazan, Russian Federation

Abstract. Modern industrial manufacturing has high requirements for employees. Workers must continually improve their competence. You must constantly test, train and retrain specialists of enterprises. In the light of manufacturing control problems and cognitive issues of identification, verification and other problems for processes of identification, modeling, and control, intelligent tutoring system for specialists training can be considered as an important part of manufacturing intelligent control and planning system. The development and a pilot implementation of adaptive approaches to the universal computer system of individual training, functioning on the basis of subject and meta knowledge bases and meta knowledge bases with the latest advances in information and communication technologies, and computing and artificial intelligence is a very important part of intelligent control and planning for manufacturing system. Staff must continuously learn novel the techniques and use of the learning system in the educational process for solving practical problems in individual courses within their existing production competence and future qualification requirements. To improve the system functionality and its compliance with the modern educational process needs it is necessary to take into account the comments and suggestions from users. This system also can be used for more productive generation of ideas to improve the system and eliminate bottlenecks. Actually an intelligent tutoring system helps to solve the broader problems, i.e. a learner understands and sees not only one option for solving the a problem task and working on a predetermined pattern, but begins to perceive the subject areas of the enterprise as a part of the whole system.

Key words: Learning Management Systems, Intelligent tutoring system, Model identification, Manufacturing control, Cognitive issues of identification, Positive-formed formulas.

1. INTRODUCTION

Modern industrial manufacturing has high requirements for employees. Workers must continually improve their competence. You must constantly test, train and retrain specialists of enterprises. An adaptive program of development of new competencies is needed to perform advanced enterprise plans. They should be formed depending on the program and production plans, as well as based on the analysis of skills of learners themselves.

In the light of manufacturing control problems and cognitive issues of identification, verification and other problems for processes of identification, modeling, and control, intelligent tutoring system for specialists training can be considered as an important part of manufacturing intelligent control and planning system [1].

The proposed approach can be considered as a component of the global problem of forecasting and management of the development of productive forces and production relations of society depends on the individual capabilities of each worker, and as a component of the inverse problem of advanced training of specialists, who is "needed" for the economy.

Training of highly skilled professionals that are competitive in the labor market, as well as competent and able to work effectively in the specialty, is a prerequisite for the successful development of any country. The recently formed lack of trained engineers in high-tech industries reflects the prevailing tension between the modern engineering complexity and the actual training level of technical specialists with higher education.

[1] Corresponding author. E-mail: seyl@mail.ru

Today it is a large and growing in scale problem of many businesses. There is a lack of education: instead of bringing innovative ideas to the production, investments are needed for further training and retraining of young professionals. Universities are not always able to train a sufficient quantity of decent level specialists in information technologies, data mining, machine-building design and technological specialties, integrated logistics, production planning and modeling, analysis and design of business processes, etc. [2].

The essence of the proposed approach in this paper is quite simple and can use well-known scientific and technical solutions. As necessary conditions for the successful introduction of the results and further development of a new training system are the rational use of modern and advanced information technologies, taking into account recent advances of cognitive science and developing of original high-tech methods for solving problems in the modern theory of Intelligent Control, the theory of “C cubed” (Control, Computation and Communication), etc.

2. STATEMENT OF THE PROBLEM AND THE MAIN RESULTS

Teaching many technical and mathematical subjects is impossible without learning problem solving. It is necessary not only to demonstrate the examples of reasoning in solving various problems, but also to offer the student a tool for problem solving and self-check of the results. Comparison of the numerical value of the response task obtained by a student with a reference value is not sufficient to control the level of the student’s assimilation of the problem-solving technique. Especially if he or she is asked to choose an answer from a fixed set of options. First, the student should find a solution in the general form (to obtain a formula), and only then perform the calculations. Thus, we need to be able to compare two formulas up to equivalence transformations (in the general case is algorithmically undecidable), but first it is necessary to get the formula from the user (in a machine-readable format). Second, to trace the student’s reasoning, you need to know at least what formulas and in what sequence he or she used (because we cannot count on the analysis of natural reasoning). That is, the task of getting the formula from the user and comparing it with the existing needs to be repeated many times.

Modern information technologies training mainly provides students with electronic versions of educational hypertext material, possibly using multimedia tools, as well as remote access to these information resources and remote interaction with the teacher and other students, including real time. The achievements of artificial intelligences are still used in a much lesser extent. This conclusion applies to the Learning Management Systems and learning content (LCMS), both commercial and open source systems (e.g., Moodle).

Content of this work is the development and pilot implementation of adaptive approaches to the universal computer system of individual training, functioning on the basis of subject knowledge and meta knowledge bases with the latest advances in information and communication technologies, computing and artificial intelligence as an important part of intelligent control and planning manufacturing system. The work includes the presentation of the basic theoretical foundations, detailed solutions of typical examples; explanations of practical applications of the mathematical results, as well as challenges for independent solutions. It is not simple tests that require calculate a numeric value, or select one of several options, but it must provide control of the correct course of solving the problem, review and updating of the results.

We are talking about the formation of the individual learning program material based on real opportunities to learn using the level of training dynamic selection and feedback. The entire process can be analyzed and logged, while training the next lesson; you will have more efficiency using the accumulated experience of the system.

3. BRIEF DESCRIPTION OF THE ALGORITHM OF THE MODULE "UNIVERSAL TASKS SOLVER ON THE BASIS OF POSITIVE-FORMED FORMULAS"

The main components of the pilot software package [3] is the module "Universal tasks solver on the basis of positive- formed formulas". The basis of knowledge representation here is a classic first-order language *L* of positively constructed formulas. Positively formed formula - the formula is derived from *N* concluding statements ($N \geq 1$) using standard quantifiers and logical connectives &, □∨, and which do not limit (in its syntax) representability of various properties. Moreover, the presentation of knowledge in the form of language *L* using only standard existential and universal quantifiers, logical connectives conjunction and disjunction is not explicitly recorded and assumed in conjunctive and disjunctive branching-formulas [4]. Output is to focus on the method of contradiction. Inference rule in the calculus *J* positively constructed formulas denoted as ω.

Thus, an algorithm for finding solutions using calculus *J*, based on the language *L* can be represented as follows:

- have a starting base *B* (which is given), i.e. many well-known facts;
- have the relations of computability in the knowledge base that can be used to form the basis of questions to *B*;
- by the operation ω find answers to some questions and expand the facts base *B*;
- if the answer is made by the target issue, i.e. refutation of contradictions, we come to a successful conclusion;
- if the answer to the question is made with disjunctive subformula branching, the fact base is split by the number of bases corresponding to the number of elements in the disjunctive formula;
- in case of splitting database operations using ω again is searched for the new database response and so on, to successfully complete location solutions or not.

The algorithm of the universal problem solvers in this case can be summarized as follows:

- The Raw data (given, goal) enter the solver from the graphical user interface (interactive equation editor);
- The Raw initial data base is formed by the facts;
- Loop solver associated with the Knowledge Base (which contains the ratio of computability) and selects the required ratio of computability;
- On the basis of computability relations the solver generates a design according to the structure - formulas, applies the rule ω and given the potential ramifications of the facts base (raw data), through all the branches in the original conjunctive branching;
- Solver adds facts to the base;
- The process continues as long as the option is not found/solutions in the form of a refutation of the contradictions, otherwise - the message about the absence of the possibility of solving the problem with this facts and knowledge database.

Schematically, the process of finding a solution to the problem can be represented by the following relationships between the graphical user interface, solver of problems, the original data and knowledge base (Fig. 1-2)

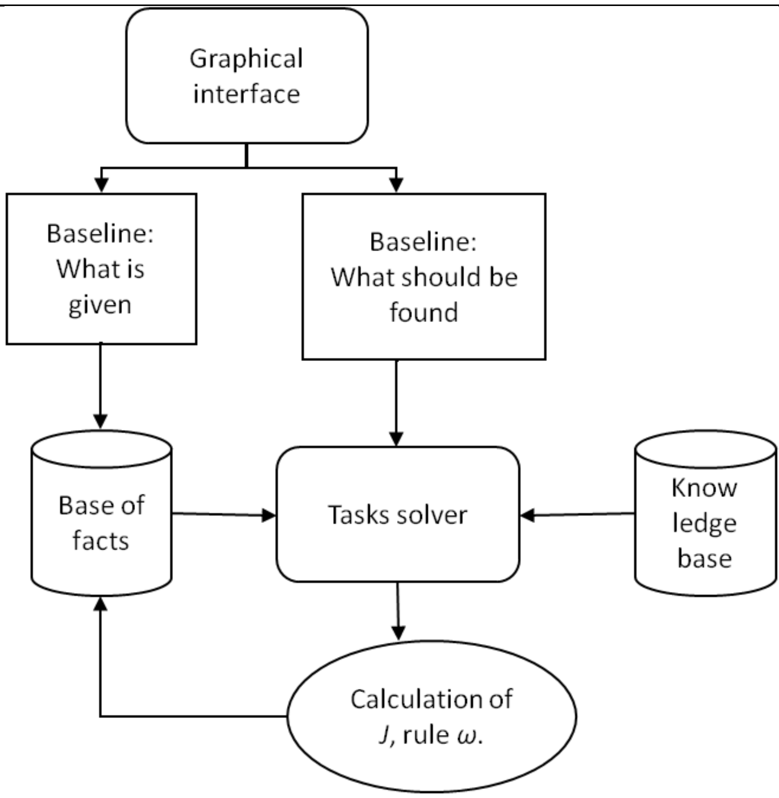

Fig. 1. Process of finding a solution to the problem

4. PRINCIPLES OF FORMATION OF THE FORMULA EDITOR FOR INTERACTIVE INTELLIGENT TUTORING SYSTEM

Computer software formula editors are of great importance in teaching technical subjects. They are needed when you create training material, as well as directly in the learning process. It offers ideas for the construction of the Formula Editor you would really want to use. On the basis of these ideas a working formula editor has already been created. This equation editor is used in the interactive learning system of intellectual training of engineers [7].

These ideas are as follows. Form formula by typing the name of the keyboard characters, operators, etc., that you want to insert. Moreover, it made directly in the text of the formula. The system then replaces the typed text to the desired mark, operator or member. Enough variants of the name that you use it (integral matrix, matrix, etc.). Also, the system tries to guess what you want, with the first letter typed, and offers options for items to choose from. As you type, a list of options varies.

This approach works well in integrated development environments used by programmers. Here the system guesses what the programmer is going to write on, and offers options for continuing or replacement text. This feature is known as code completion. Our experiments showed that the use of this approach for formulas reduces the formula typing time compared with the traditional interface to enter a formula using the mouse. Revealed the following features.

- Beginners do not need to study the menu and the editor of formulas to find the right icon. They already know names of the needed elements.

- The student, typing formula, can continue to think about the formula itself. He or she doesn’t need to transfer attention to finding the right button, the window or the icon. Inserting the name of the element is made directly in the text of the formula, and not in a separate window.

- Interface to enter a formula should contain the necessary elements to enter and control the formula using the mouse as in traditional systems, since it also has certain advantages.

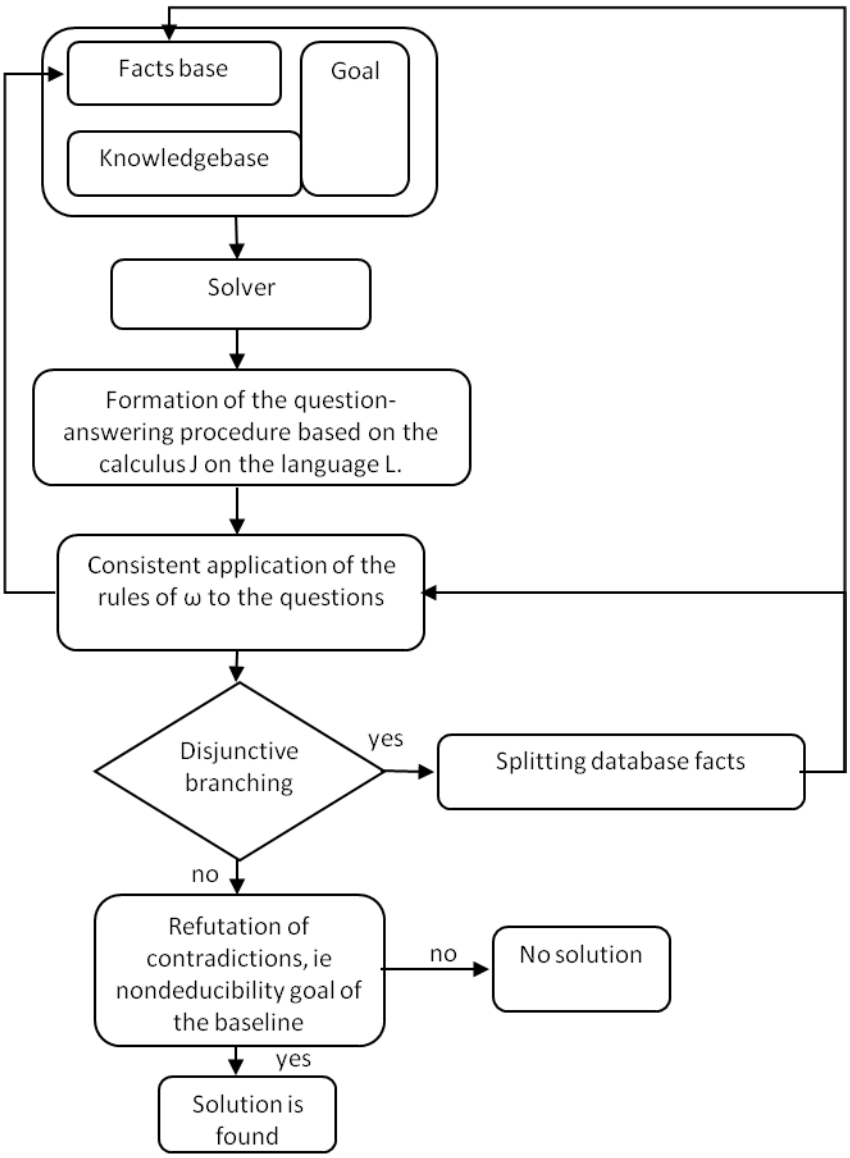

Fig. 2. Algorithm for finding the solution of the problem

- Allow you to specify the semantics of typed formulas. That is, allow the user to specify exactly what he got – a superscript or degree opening bracket arithmetic expression or a list of function parameters, vector, matrix, etc.

- Automatic guessing what is typed by the user, based on the context surrounding the formula. That is, if a function is assumed in calculations of *v*, but there is a variable with the same name, then the bracket for the symbol *v* rather marks the beginning of the list of its parameters, than anything else. Also in the opposite direction, it is possible to use semantic information from a typed formula to automatically generate a list of user-entered or used variables, functions, and other objects. This is true for both research and training systems. For example, to request the user to provide the meaning of an introduction of a variable already present in the context of the problem being solved.

- A formula may be managed fully with the keyboard, waiting for the user's next action. For example, adding new parameters of a function by pressing the comma. Inserting new rows and columns by pressing the Enter key and the space bar, respectively.

- Add to a user the possibility to undo and redo actions and also the opportunity to return to any of the formulas in the timeline, copy from an old version of the desired piece and insert it into the current version. Further some of the implementation details are provided.

It is proposed to move away from the standard approach - abstraction and unification of structures and methods. For a software implementation does not use the formula editor arithmetic data types (ADT) and / or parse trees formats for mathematical formulas (MathML, etc.) and reflects the perception of the structure of the formula written by the user. For example, a formula does not seem tree operators /, operations on arguments (ADT), and an ordered sequence of its constituent elements (variables, operators, etc.), see Fig. 3.

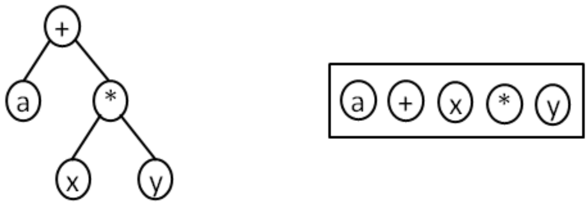

Fig. 3. Representation of formulas in the computer memory

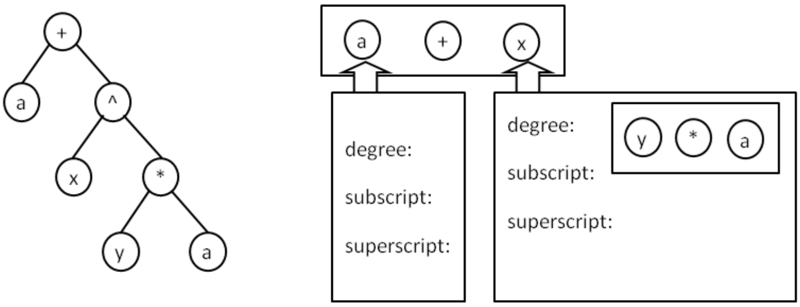

Fig. 4. Presentation exponentiation operation in the computer memory

Exponentiation is represented not as a transaction with two arguments (MathML, etc.). It is a characteristic of an element (variable, function, etc.) - the extent to which he was elevated (Fig. 4). Division operation will then be presented in two ways (Fig. 5).

This approach simplifies the system. System behavior when navigating and editing a formula corresponds to the user's expectations, and not abstract structures in computer memory. It then becomes easier to predict what the user wants to do next, but has not yet made, and to help the user to perform these actions.

The program realizes the possibility of undo and redo actions. Changes are made through the design pattern (Command). In this case a formula itself does not change (Fig. 5). If you want to make changes to the formula, the program creates a special object (command object) with two methods – "apply changes" and "undo the changes." This object allows you to make changes to the formula These objects are stored in a special stack. If necessary, they in turn are popped from the stack, used for undo and placed in another stack to repeat actions.

The command object, which inserts a new element into the formula, when recalling the method "apply changes", it inserts the same object as the first call. Objects that removes an element after canceling of the removal return to the formula the object that has been deleted, and do not generate a new identical object.

This allows you to insert, delete, and modify the formula to implement elements directly via pointers to the object in memory (since reached the immutability of pointers), and not only use the location element in the formula with respect to its root / start. Application of direct pointers to mutable object simplifies writing and empowering online and accelerates the work itself online.

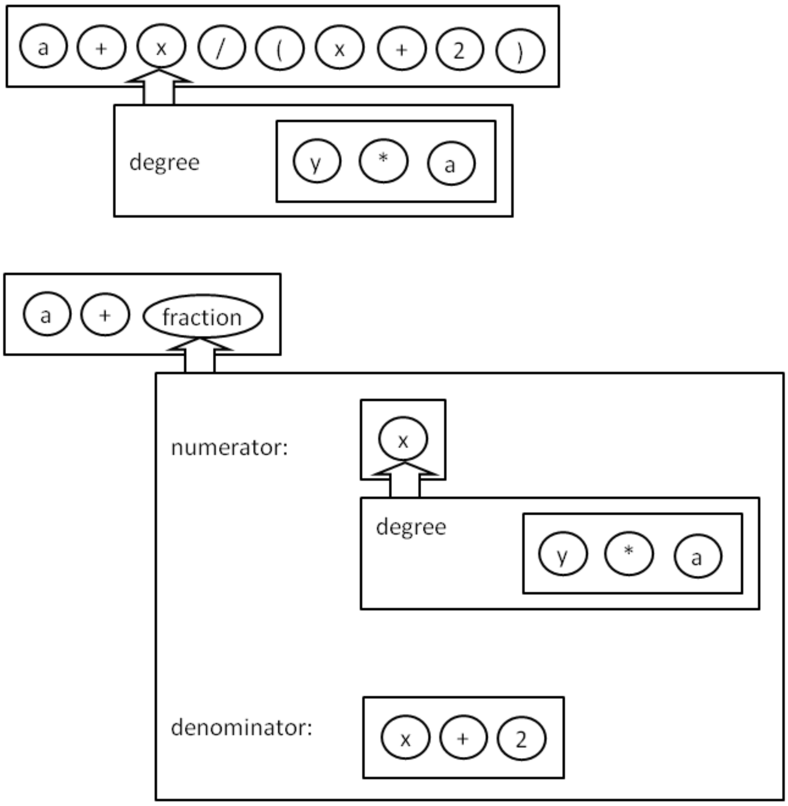

Fig. 5. Substitution the division operation in the computer memory

Implementation of these ideas has already shown tremendous speed gain in typing a set of formulas and invaluable assistance to concentrate on the essence of the case that the user is working on.

Actually, the same formula editor can be used for scientific research and for writing articles. Together with our colleagues, we continue to develop it.

Thus, the role of an editor of mathematical formulas is twofold: on the one hand, it should allow a student to be comfortable enough to express their thoughts, and on the other – should allow effective handling of user's input. In doing so, it would be desirable to reduce the possibility of typing errors. Moreover, the editor's usage should not assume that a user has special knowledge, representation of formulas should be familiar and intuitive.

These requirements are met by the flexible editor. Formulas in the editor are presented in a familiar graphical form, as they would be printed in a textbook. The editor uses the concept of a template formula – a sort of a "canonical" representation in which this type of formula is written (which makes them easy to compare). A user selects a pattern corresponding to the desired formula, and then fills it with slots of necessary constants, variables and functions. Symbols of constants, variables and functions are presented by graphic primitives, roaming through the mechanism of drag-n-drop. In addition, the editor allows you to further adjust the difficulty of a task, varying the amount of "designer details" available to the user.

Templates in the system are generated automatically from text formulas. In a formula (generated by parsing the text representation) variables, functions, constants are indicated. They are not inserted in the template automatically, instead of that cells for substitution variables, functions and constants are put in their places. Other elements (operators, etc.) are inserted into the template unchanged.

6. CONCLUSIONS

The development and a pilot implementation of adaptive approaches to the universal computer system of individual training, functioning on the basis of subject and meta knowledge bases with the latest advances in information and communication technologies, computing and artificial intelligence is a very important part of intelligent control and planning for manufacturing system. Staff must continuously learn novel techniques and use a learning system in the educational process for solving practical problems in individual courses within their existing production competence and future qualification requirements. To improve the system functionality and its compliance with the modern educational process needs it is necessary to take into account comments and suggestions from users [5-6].

Staff can also use a complex learning system to have a holistic picture for more productive generation of ideas which can be used for improving the production system and eliminating bottlenecks. Actually, an intelligent tutoring system helps to solve the broader problems, i.e. a learner understands and sees not only one option for solving a task and working on a predetermined pattern, but begins to perceive the subject areas of the enterprise as a part of the whole system. This happens largely because using the latest achievements of artificial intelligence, telecommunications and information technologies it is possible to study fast and find efficient solutions of a various options with the learner’s active participation. This also significantly simplifies the process of domain knowledge modeling. A similar effect can also be achieved with conventional training methods, i.e. using a constant direct contact between a teacher and a learner.

Obviously, to achieve a similar effect in the case of traditional teaching methods in average much more time and money is required than with the learning process of the developed software system. Here the learning system actually allows you to simulate the process of individual learning. The effect of the individual approach in this case is very significant and convenient. You can continue with your work at any time that is not practicable at the standard training scheme. In the case of a group of traditional forms of education it is extremely difficult for a teacher to give sufficient time for an individual approach for each group member.

The system can be used very effectively in groups where there are different levels of competence of learners. Weaker learners will not hinder the work of strong learners. All motivated learners have an opportunity to improve their performance through their individual work.

References

[1] Vassilyev, S.N., Sabitov R.A., Degtyarev, G.L., Kozlov, V.V, Malivanov, N.N., Sabitov, Sh.R. and Sirazetdinov, R.T. *Adaptive Approach to Developing Advanced Distributed E-learning Management System for Manufacturing.* Preprints of the 13th IFAC Symposium on Information Control Problems in Manufacturing, Moscow, Russia, pp. 2198-2203, 2009

[2] Vasiliev S.N. and Sabitov R.A. *The Knowledge Economy and the intelligent management.* Proceedings of the X International Conference Chetaev, Kazan, 2012.

[3] Vassilyev, S.N., Sabitov Sh.R., Sirazetdinov B.R., Smirnova, G.S., Sukonnova A.A. *Architecture and functions tracking intelligent tutoring system "Volga" №4*, KNRTU-KAI named after A.N. Tupolev, Russia, Kazan, 2012

[4] Vasiliev S. N., Zherlov A. K., Fedosov E. A., B. E. Fedunov (2000) *Intelligent control of dynamic systems.* – Moscow: Physical and mathematical literature

[5] Smirnova, G.S., Sabitov, R.A., Sabitov, Sh.R., Korobkova, E.A, Kislov, A.S. *Program-methodical complex of operations management №4*, KNRTU-KAI named after A.N. Tupolev, Russia, Kazan, 2012

[6] Smirnova, G.S., Sabitov, R.A., Elizarova, N.Y., Sh.R. Sabitov *Operational management system of enterprise production processes "1C: MES: cloudy production management*. Journal “Automation in Industry”, 2014, № 8, Russia, Moscow

[7] Sirazetdinov B.R., Smirnova, G.S., Korobkova E.A. *Principles of formation equation editor for interactive intelligent tutoring system for training engineers in aerospace disciplines.* International scientific-practical conference "Modern technologies and materials – the key point in revival of domestic aircraft building", Kazan, 2010