TOOLS FOR DESIGNING COMPLEX SOFTWARE SYSTEMS BASED ON SPECIAL LINGUISTIC CONSTRUCTIONS AND ALGORITHMS FOR THEIR PROCESSING

O. Ja. Kravets [(1)], I. A. Aksenov [(2)], Yu. V. Redkin [(3)], P. A. Rahman [(4)], D. I. Mutin [(5)], Amoa Kouadio-kan Armel Geoffroy [(1)], M. A. Ermolova [(6)]*

[(1)] Voronezh State Technical University, Voronezh; [(2)] Vladimir State University named after Alexander and Nikolay Stoletovs, Vladimir; [(3)] Admiral Ushakov Maritime State University, Novorossiysk; [(4)] Ufa State Petroleum Technological University; [(5)] Moscow State University of Technology "STANKIN", Moscow; [(6)] Bauman Moscow State Technical University, Moscow
Russian Federation

* Corresponding Author, e-mail: csit@bk.ru

Abstract: The article presents an approach to the design of complex software systems based on special linguistic constructions in the form of Backus-Nour forms and syntactic diagrams. The goal is to create special mathematical and software that would formally describe the components of the designed software system for the study of a complex object, as well as visually form the structure and algorithmic component of the system. As a result, a formal language for describing discrete-continuous objects has been developed, which ensures the unification of the display of static and dynamic objects by introducing a generalized "element" construction; special mathematical software has been created for the formal description of complex systems, allowing to obtain syntactic constructions of minimal complexity due to the orthogonality of linguistic means; algorithms for processing language constructions have been created that take into account the parallelism of the processes of the studied objects and ensure the automated creation of a class system of special software.

Key words: formal description, static and dynamic objects, formal language, concurrency accounting, computer-aided design.

1. INTRODUCTION

When analyzing the design process of special software [1], it was found that the redundancy of conditional operators, the need to use an unconditional transition operator [2] and the lack of advanced debugging tools make it difficult to effectively build data processing algorithms. The separation of single-channel and multi-channel

devices into fundamentally different groups makes it difficult to change the functional purpose of a system element within a single process. When studying complex objects with a multicomponent composition [3], interconnections of various nature, as well as the hierarchical organization of elements, the lack of developed graphical means of representing the structure of the system makes it difficult to create software for conducting research.

It is necessary to create special mathematical and software that would formally describe the components of the designed software system for the study of a complex object, as well as visually form the structure and algorithmic component of the system. With the help of software interfaces, information from the designed components should be transmitted to research complexes, which use it to implement an effective assessment of the characteristics of complex objects. To do this, you need to:

1) analyze classes of complex systems for the complexity and the possibility of their research using existing special mathematical and software complexes;

2) create mathematical and linguistic support for the formal description of the process of functioning of complex systems;

3) design special software that implements the creation of software libraries for building algorithms for the study of complex objects;

4) implement software tools for automating the process of designing research complexes, conducting analysis and obtaining statistical characteristics.

The main task is defined as reproducing the behaviour of the system under study based on the results of the analysis of the most significant relationships between its elements.

2. THE STATE OF THE PROBLEM

The main classes of real systems are considered, to which one or another method of creating research algorithms is applicable. According to the class of real objects, the following types of algorithms are distinguished: agent-based, discrete-event and based on system dynamics.

The analysis of the most common tools for creating special mathematical and software using basic research algorithms and allowing the study of complex objects is carried out. These tools include:

1) specialized software complex Arena - developed by Rockwell Software, has many tools for building complex research algorithms based on an agent-based approach. The basis of this system is the SIMAN modelling language and the Cinema Animation system; there are no possibilities for effective visual design;

2) GPSS World system [7] - there is a built-in language that allows you to effectively set research algorithms. A discrete-event approach is used, but animation of the modelling process and graphical representation of the structure of the object under study are not provided;

3) the specialized mathematical and analytical system Simulink [8] is part of the MathWorks MatLab package, it is aimed mainly at the study of dynamic systems, the issues of studying interacting software objects are difficult.

The analysis showed that there are factors that reduce the effectiveness of using specialized tools:

1) redundancy and bulkiness of conditional operators;

2) the need to use the unconditional transition operator for some algorithms;

3) separation of single-channel and multi-channel devices into fundamentally different groups;

4) inaccuracy of statistics on queues of limited length;

5) the inconvenience of debugging specialized algorithms;

6) the lack of means of graphical representation of the structure of the system under study, allowing to assess its complexity and the nature of the interrelationships of the elements.

The problem of increasing the efficiency of automation of the design of specialized research complexes is investigated. It is shown that existing tools can be improved through the use of new tools for the formal description of the functioning of complex objects, as well as through the refinement of algorithms.

The creation of specialized mathematical and linguistic tools that allow formally describing the process of functioning of discrete-continuous systems, as well as unifying the display of static and dynamic objects, is the subject of the article.

3. LINGUISTIC ELEMENTS

Formal descriptions of new structural means of model representation of the process of functioning of real objects are proposed:

$$|E| = N \tag{1}$$

$$Rm_{i,j} = Rf(e_i, e_j) = \begin{cases} 1, \text{if } e_j \text{ is related to } e_i; \\ 0, \text{otherwise} \end{cases} \tag{2}$$

$$Mm_{i,j} = Mf(e_i, m_j) = \begin{cases} 1, \text{if } m_j \text{ belongs to } e_i; \\ 0, \text{otherwise} \end{cases} \tag{3}$$

$$Km_{i,j} = Kf(m_i, k_j) = \begin{cases} 1, \text{if } k_j \text{ belongs to } m_i; \\ 0, \text{otherwise} \end{cases} \tag{4}$$

$$Pm_{i,j} = Pf(e_i, p_j) = \begin{cases} 1, \text{if } p_j \text{ belongs to } e_i; \\ 0, \text{ otherwise} \end{cases} \tag{5}$$

$$Hm_{i,j} = Hf(e_i, h_j) = \begin{cases} 1, \text{if } h_j \text{ belongs to } e_i; \\ 0, \text{otherwise} \end{cases} \tag{6}$$

where E is the set of all elements of the system, N is the total number of elements; R, M, K, P, H are matrices of connections, messages, message parameters, properties of elements and events; Rm is an element of the bond matrix, Rf is a function defining the bond matrix; Mm is an element of the message matrix, Mf is a function defining the message matrix; Km is an element of the message parameter matrix, Kf is a function defining the message parameter matrix; Pm is an element of the matrix of element properties, Pf is a function defining the matrix of element properties; Hm is an element of the event matrix, Hf is a function defining the event matrix.

Using a formal description, a means of transition from a meaningful description of the behaviour of a complex object to a formalized representation of the process of its functioning has been created:

$$M =< E, Rm, M, Mm, K, Km, P, Pm, H, Hm > . \quad (7)$$

The functions Rf, Mf, Kf, Pf, Hf are implicitly involved in (7).

During the creation of specialized linguistic tools, the decomposition of the process of studying a complex object based on research complexes was carried out. Two stages of this process are highlighted. The first of them is related to the description of the functioning of a real system and includes the following tasks: dividing the entire system into separate elements; describing the relationships between the identified elements; determining ways of interaction between the objects of the system; identifying conditions and restrictions on the interaction of individual parts of the object under study; creating rules for calculating the necessary statistical characteristics. The second stage consists in the implementation of the identified rules of operation in the form of algorithms or programs.

Based on the analysis of the first stage, new objects of linguistic support for the formal description of specialized research tools have been created. The created objects make it possible to describe the real system with the required accuracy, and the formality of the syntactic representation provides an automated study of a complex object. The implementation of the second stage is not required, provided that new objects of linguistic support are used. The objects are presented in the Table 1 and 2.

Table 1. The main objects of the formal description language

Title	Description
"Hierarchy"	A construction that allows you to represent the system under study in the form of a hierarchical structure
"Element"	A construction that describes a part of the selected system as a single whole
"Property"	A construction that describes any characteristic of a real object
"Message"	The design for representing signals by which individual parts of system interact
"Parameter"	The design describes any values transmitted during the exchange of specific messages
"Event"	It is a linguistic tool for formally describing the reaction of a separate part of the system in response to receiving a specific
"Communication message"	The design allows you to describe the relationships inherent in different parts of a real object

Table 2. Additional objects of the formal description language

Title	Description
"Property list"	Contains one or more constructs "Property"
"Message list"	Contains one or more constructs "Message"
"Parameter list"	Contains one or more constructs "Parameter"
"Event list"	Contains one or more constructs "Event"
"Link list"	Contains one or more "Connection" constructs

The formality of the representation of the functioning process with the help of created objects is ensured by defining all constructions in the form of Backus-Naur forms and syntactic diagrams. For the main objects, these forms are given below. The syntax diagrams for them are shown in Figures 1-6.

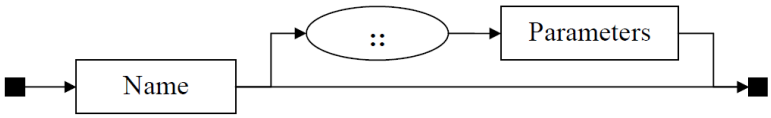

Figure 1. Syntactic diagram for the "Message" construction

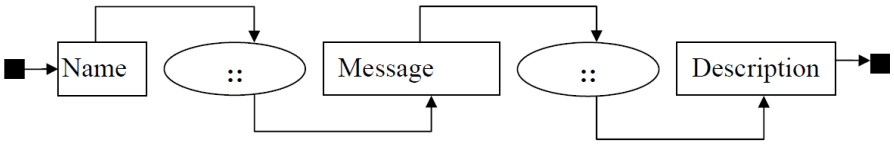

Figure 2. Syntactic diagram for the "Event" construction

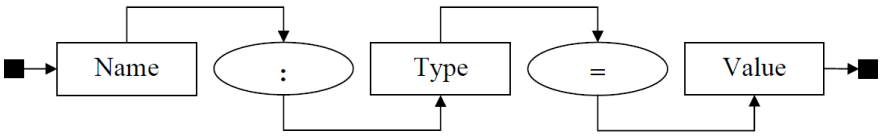

Figure 3. Syntactic diagram for the "Property" construction

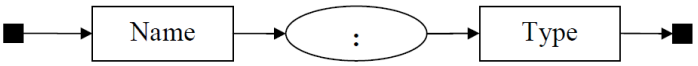

Figure 4. Syntactic diagram for the "Parameter" construction

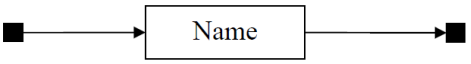

Figure 5. Syntactic diagram for the "Connection" construction

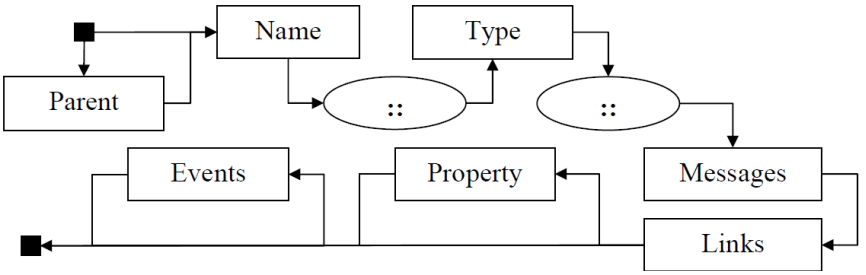

Figure 6. Syntactic diagram for the "Element" construction

The creation of linguistic means for the formal description of the process of functioning of complex objects requires the design of specialized algorithms for processing language structures that allow for the parallelism of processes [9] in heterogeneous systems [10] and ensure the automated creation of a class system of special software.

4. ALGORITHMIZATION OF THE DESIGN PROCESS OF AUTOMATION TOOLS FOR BUILDING RESEARCH COMPLEXES USING THE UML LANGUAGE

The stages of creating special software for the implementation of research complexes are defined: building the structure of the object under study; creating a list of messages; setting events and properties; writing event handlers; defining the rules for collecting statistical data; analysis of the identified statistical characteristics.

An algorithm for creating special software that allows for the study of a complex object is constructed (Figure 7).

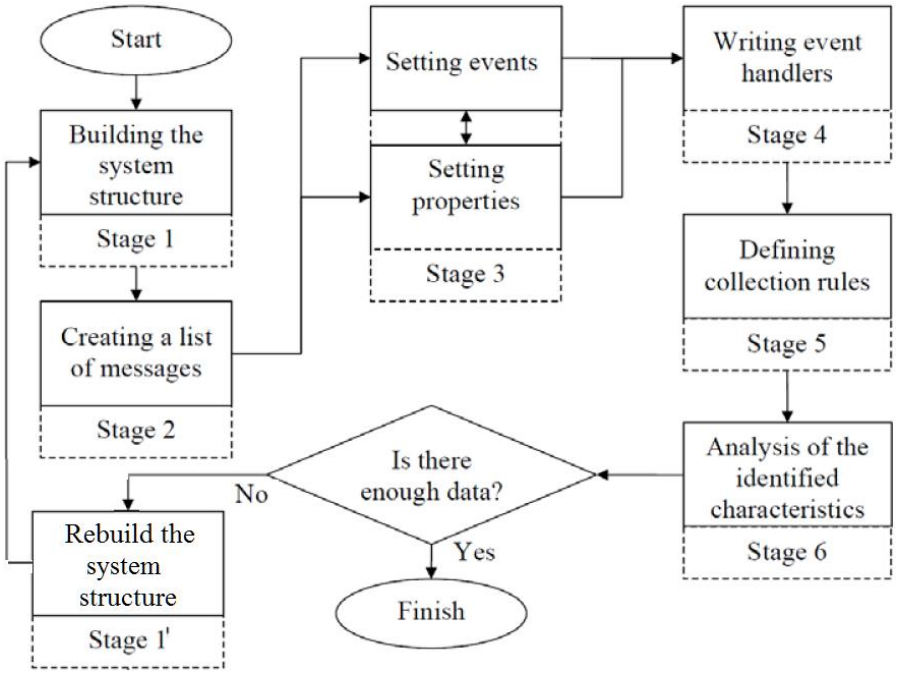

Figure 7. An algorithm for building specialized software for conducting research

The decomposition of the general usage diagram has been performed. As a result, specialized diagrams were obtained for constructing the structure of an object, creating a list of messages; creating a list of properties, creating events and their handlers; defining rules for collecting statistical information. This made it possible to clarify the requirements for the designed software.

Based on the identified requirements for the functionality of the designed software, its structural parts are allocated, each of which is designed to solve specific tasks.

1) System editor – designed to create, modify and delete special research algorithms in the developed tool;

2) Structure editor – allows you to set the structure of an object consisting of interrelated elements;

3) Message editor – designed to work with messages;

4) the element properties editor – provides work with the characteristics and events of the elements;

5) Event handler code editor – is a text editor with syntax highlighting, allows you to edit code in a high-level language;

6) the editor of the rules for collecting statistical data is designed to create rules according to which various statistical characteristics will be obtained in the future.

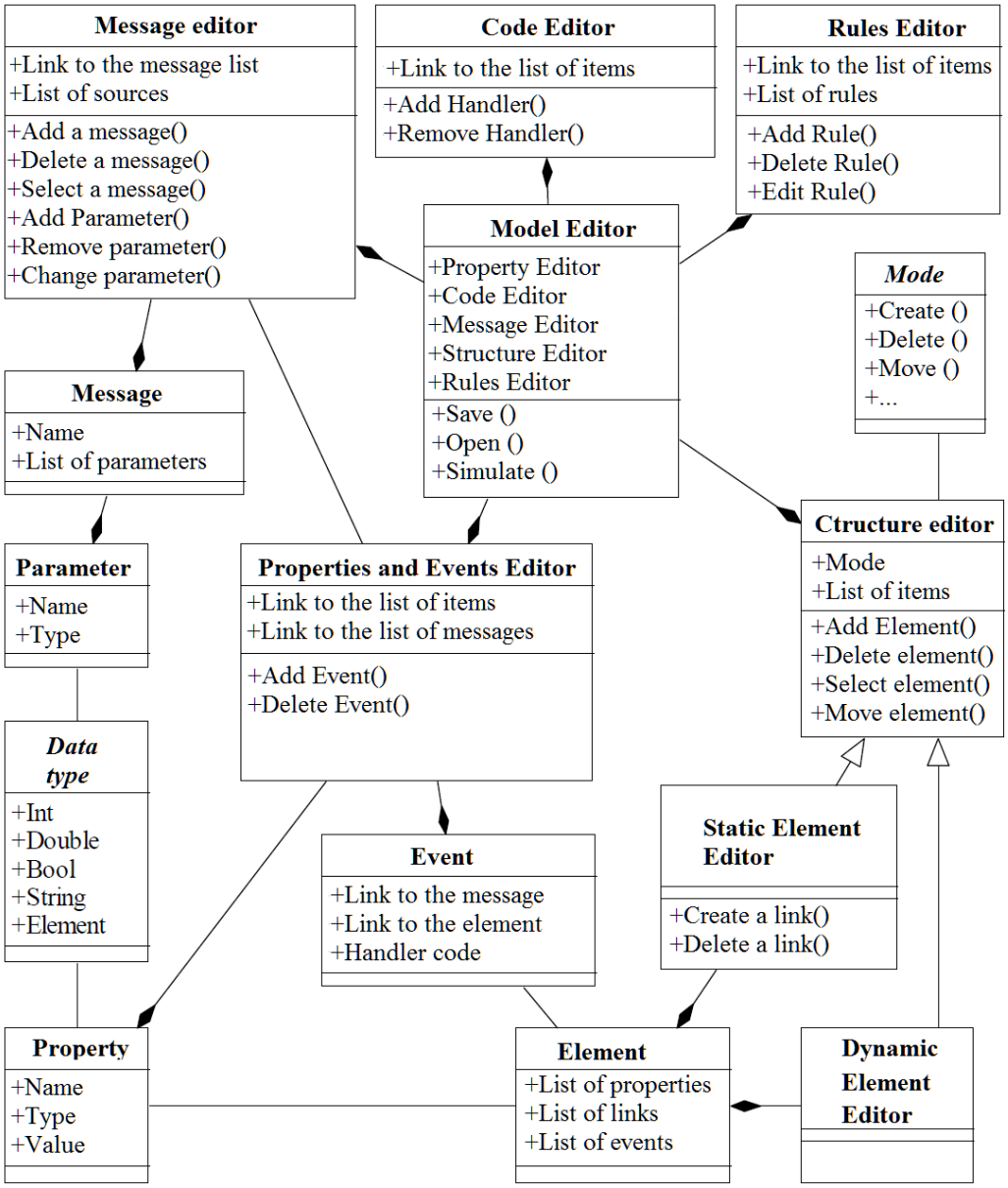

Figure 8. Detailed class diagram for specialized software

Various characteristics of the relationships between classes of the designed specialized application are defined. In accordance with the UML methodology, the links are plotted on a detailed class diagram. It is established that the classes of editors of static and dynamic elements are inheritors of the class of the structure editor. It is determined that the classes of all editors are part of the system editor class. It was decided to make the mode and data type classes static. A detailed class diagram reflecting the main objects of the designed specialized software, as well as the nature of the relationship between them, is shown in Figure 8.

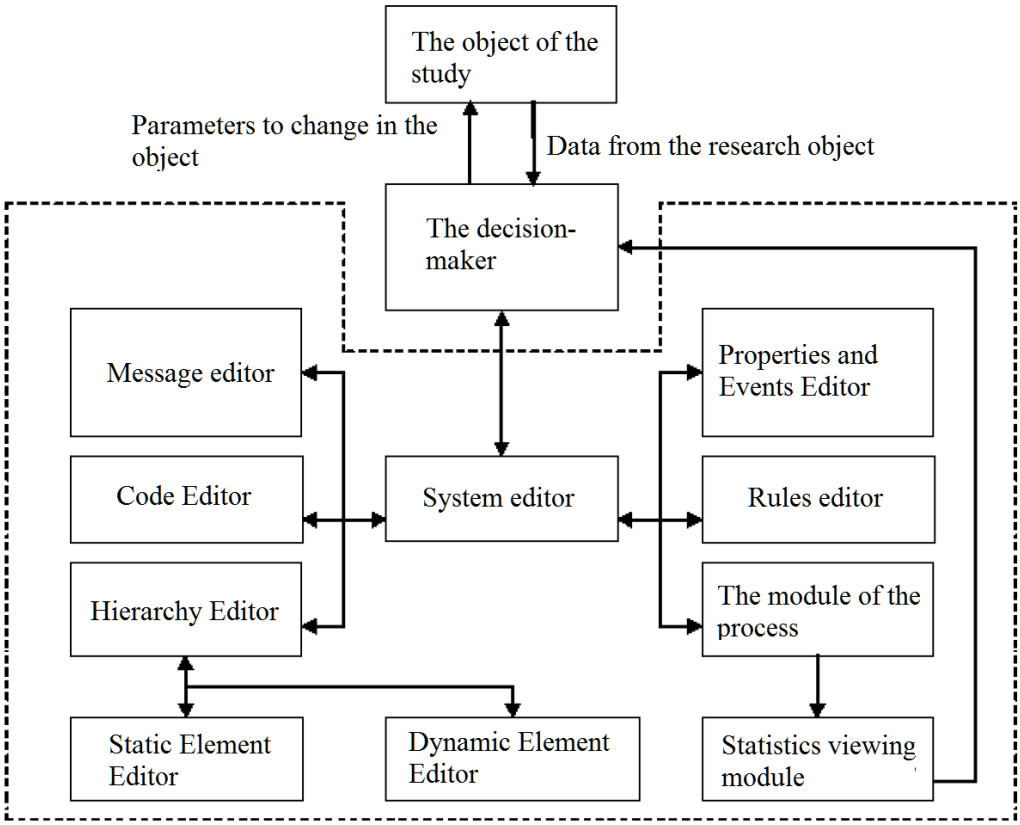

Figure 9. The general structure of the implemented specialized software

An algorithm for the timer operation reflecting the system time has been created. This makes it possible to organize parallel processing of signals and data within the framework of a single research algorithm between related elements.

Thus, the means of processing language constructions are proposed, taking into account the parallelism of the functioning processes and ensuring the automated creation of a class system of special software.

5. IMPLEMENTATION OF COMPUTER-AIDED DESIGN ALGORITHMS FOR SPECIAL SOFTWARE FOR CONDUCTING RESEARCH ON COMPLEX OBJECTS

The general structure of specialized software is shown in Figure 9.

The object-oriented C# language and Microsoft Visual Studio visual design tools are used to implement specialized software.

The modular structure of the created specialized software consists of 12 files: StructureBuilder – contains the code of the structure editors and the code of related classes; GraphicElement – contains classes responsible for the graphical representation of the structure of the object under study, as well as for the relationship between the element and its properties, messages, events, etc.; ElementProperties – contains the code of the property editors and events, as well as additional related classes; EventsBuilder – includes the code of the message editor; CodeEditor – contains the code of the code editor; SyntaxHighlighting – includes a class for highlighting the syntax of the event handler code; CodeCompiller – contains the code of classes responsible for compiling special algorithms; RulesBuilder – includes the code of the statistical information collection rules editor; ViewResults – this module contains the code of tools for viewing results; WorkSpace – contains the class code, combining all editors into a single software unit; Additional – this module contains static classes, as well as common classes necessary for other modules; ModelElement – includes templates for building a class system that allows you to create a specialized research algorithm.

6. CONCLUSION

The following results were obtained:

A formal language for describing discrete-continuous objects has been developed,

2. Special mathematical software has been created for the formal description of complex systems.

3. Algorithms for processing language constructions have been created that take into account the parallelism of the processes of the studied objects and ensure the automated creation of a class system of special software.

4. The means of creating special software for the study of complex objects have been implemented based on the integration of high-level language constructs to reduce the complexity of the process of creating research complexes.

REFERENCES

[1] Otieno M., Odera D., Ounza Ja.E. Theory and practice in secure software development lifecycle: A comprehensive survey. *World Journal of Advanced Research and Reviews*, vol. 3, no. 18, 2023, pp. 53–78.

[2] Doria S. Coherent Upper Conditional Previsions Defined through Conditional Aggregation Operators. *Mathematics*, vol. 24, no. 10, 2022, article no. 4728. DOI: 10.3390/math10244728.

[3] Li Y., Ramli D.A. Fractional synchrosqueezing transform for enhanced multicomponent signal separation. *Scientific Reports*, vol.14, no.1, 2024. DOI: 10.1038/s41598-024-68216-x.

[4] Ji Ruyi et al. Decomposition-based Synthesis for Applying Divide-and-Conquer-like Algorithmic Paradigms. *ACM Transactions on Programming Languages and Systems*, vol. 46, no. 2, 2024, pp. 1-59. DOI: 10.1145/3648440.

[5] Alzalg B. *Combinatorial and Algorithmic Mathematics: From Foundation to Optimization*. John Wiley & Sons, Inc., 2024 (506 p.)

[6] Allen T.T. Introduction to ARENA Software. In book *Introduction to Discrete Event Simulation and Agent-based Modeling*. Springer, London, pp. 145-160.

[7] Cox S.W. GPSS World: A brief preview, *Winter Simulation Conference Proceedings*, Phoenix, USA, 1991, pp. 59-61.

[8] Eshkabilov S. *Beginning MATLAB and Simulink: From Novice to Professional*. Springer, 2019. 524 p.

[9] Kravets O.Ja. et al. Algorithms and methods for managing request flows in a distributed service system. *International Journal on Information Technologies and Security*, vol. 15, no.4, 2023, pp. 73-80.

[10] Spitsyn A.A. et al. Multi-objective Selection of Structure Variants for a Corporate Heterogeneous Integrated System of Information Management. *International Journal on Information Technologies and Security*, vol. 13, no. 1, 2021, pp. 27-38.

Information about the authors:

Oleg Jakovlevich Kravets – Dr. Sci (IT), professor of Voronezh State Technical University, areas of scientific research - system analysis, optimization, simulation of complex objects

Ilia Antonovich Aksenov – PhD in economics, associate professor of Vladimir State University named after Alexander and Nikolay Stoletovs, areas of scientific research -economic information systems

Yuriy Viktorovich Redkin – PhD, associate professor of Admiral Ushakov Maritime State University, Novorossiysk, areas of scientific research - wireless networks, digital signal processing, control and data processing systems

Pavel Azizurovich Rahman – PhD in Technical Sciences, Associate professor, Department of Computer Science, Mathematics and Physics, Ufa State Petroleum Technological University, areas of scientific research information technologies, computer networks, reliability modeling and engineering

Denis Igorevich Mutin – Dr. Sci (IT), professor of Moscow State University of Technology "STANKIN", Moscow, areas of scientific research – system analysis, optimization, simulation of complex objects

Amoa Kouadio-kan Armel Geoffroy – researcher, Voronezh State Technical University, areas of scientific research - optimization, simulation of complex objects, monitoring systems

Maria Alekseevna Ermolova– PhD in Technical Sciences, Bauman Moscow State Technical University, Moscow, areas of scientific research – Modeling Methods, Planning, Equipment, Components Parts, Conveyor Assembly, Control Algorithms, Schedule Theory

Manuscript received on 06 September 2024