EDITED BY
YURII ZHURAVSKYI

INTELLIGENT DECISION SUPPORT SYSTEMS: METHODS FOR OPTIMIZING AND SUPPORTING MANAGEMENT DECISIONS

Collective monograph

Technology Center

2026

UDC 004.7
Z62

Published in 2026
by TECHNOLOGY CENTER PC®
Kharkiv, Ukraine

Z62 **Authors:**
Edited by Yurii Zhuravskyi
Yurii Zhuravskyi, Dmytro Fedorchuk, Oleksandr Perehuda, Mykola Romanchuk, Roman Stavisiuk, Dmitro Stupak, Serhii Neronov, Ganna Plekhova, Olena Feoktystova, Igor Shostak, Anastasiia Voznytsia, Andrii Shyshatskyi, Danylo Pliekhov, Oleksii Nalapko, Yuliia Vakulenko, Andrii Lebedynskyi, Oksana Dmytriieva, Ivan Starynskyi, Oleksandr Zhuk, Pavlo Zhuk, Andrii Veretnov, Olena Shaposhnikova, Yaroslav Melnyk, Oleh Shknai, Illia Dmytriiev, Oleg Sova, Olesia Zhuk, Bohdan Molodetskyi
Intelligent decision support systems: methods for optimizing and supporting management decisions: monograph / Yu. Zhuravskyi and others. — Kharkiv: TECHNOLOGY CENTER PC, 2026. — 186 p.

A collective monograph "Intelligent decision support systems: methods for optimizing and supporting management decisions" is offered for your attention. This work is the result of many years of painstaking work of researchers on solving optimization problems in decision support systems using the theory of artificial intelligence. Structurally and logically, the monograph is divided into chapters that are structured according to a certain thematic direction of research using the theory of artificial intelligence.
This scientific research is not complete, does not claim to be complete and comprehensive in this direction, but is only a separate view of the authors on this direction of research.
The monograph will be useful for researchers dealing with issues of solving optimization problems, using the theory of artificial intelligence, developing new (improving existing) approaches to solving complex technical problems in various fields of human activity.
The monograph is also useful for practitioners – designers, developers implementing modern solutions in the field of information technologies, engaged in the development of information, information-analytical, as well as automated systems.
Figures 1, Tables 17, References 101 items.

DOI: 10.15587/978-617-8360-23-8
ISBN 978-617-8360-23-8 (on-line)

Cite as: Zhuravskyi, Yu. (Ed.) (2026). Intelligent decision support systems: methods for optimizing and supporting management decisions: monograph. Kharkiv: TECHNOLOGY CENTER PC, 186. http://doi.org/10.15587/978-617-8360-23-8

9 786178 360238

AUTHORS

CHAPTER 1

YURII ZHURAVSKYI
Doctor of Technical Sciences, Professor
Department of Computer Technologies in Medicine and Telecommunications
Zhytomyr Polytechnic State University
ORCID: https://orcid.org/0000-0002-4234-9732

DMYTRO FEDORCHUK
PhD, Senior Researcher, Deputy Head of Institute for Scientific Work
Korolyov Zhytomyr Military Institute
ORCID: https://orcid.org/0000-0003-2896-3522

OLEKSANDR PEREHUDA
PhD, Senior Researcher, Head of Scientific Center
Korolyov Zhytomyr Military Institute
ORCID: https://orcid.org/0000-0001-8802-0740

MYKOLA ROMANCHUK
PhD, Senior Researcher, Deputy Head of Scientific Center for Scientific Work
Korolyov Zhytomyr Military Institute
ORCID: https://orcid.org/0000-0002-0087-8994

ROMAN STAVISIUK
PhD, Senior Researcher
Korolyov Zhytomyr Military Institute
ORCID: https://orcid.org/0000-0002-0087-8994

DMYTRO STUPAK
PhD, Associate Professor
Department of Electrical Engineering and Electronics
Korolyov Zhytomyr Military Institute
ORCID: https://orcid.org/0000-0001-7638-3982

CHAPTER 2

YURII ZHURAVSKYI
Doctor of Technical Sciences, Professor
Department of Computer Technologies in Medicine and Telecommunications
Zhytomyr Polytechnic State University
ORCID: https://orcid.org/0000-0002-4234-9732

SERHII NERONOV
PhD, Senior Lecturer
Department of Computer Science and Information Systems
Kharkiv National Automobile and Highway University
ORCID: https://orcid.org/0000-0003-2381-1271

GANNA PLEKHOVA
PhD, Associate Professor, Head of Department
Department of Computer Science and Information Systems
Kharkiv National Automobile and Highway University
ORCID: https://orcid.org/0000-0002-6912-6520

OLENA FEOKTYSTOVA
PhD, Associate Professor
Department of Software Engineering
National Aerospace University "Kharkiv Aviation Institute"
ORCID: https://orcid.org/0000-0001-8490-3108

IGOR SHOSTAK
Doctor of Technical Sciences, Professor
Department of Software Engineering
National Aerospace University "Kharkiv Aviation Institute"
ORCID: https://orcid.org/0000-0002-3051-0488

ANASTASIIA VOZNYTSIA
Graduate Student
State non-profit enterprise State University "Kyiv Aviation Institute"
ORCID: https://orcid.org/0009-0004-3767-7354

CHAPTER 3

ANDRII SHYSHATSKYI
Doctor of Technical Sciences, Professor, Senior Researcher
Department of Computer Science and Information Systems
Kharkiv National Automobile and Highway University
ORCID: https://orcid.org/0000-0001-6731-6390

GANNA PLEKHOVA
PhD, Associate Professor, Head of Department
Department of Computer Science and Information Systems
Kharkiv National Automobile and Highway University
ORCID: https://orcid.org/0000-0002-6912-6520

DANYLO PLIEKHOV
Assistant
Department of Automation and Computer-Aided Technologies
Kharkiv National Automobile and Highway University
ORCID: https://orcid.org/0009-0004-7873-1716

OLEKSII NALAPKO
PhD, Head of Research Department
Research Department for Development of Electronic Warfare Equipment
Central Scientifically-Research Institute of Armaments and Military Equipment of the Armed Forces of Ukraine
ORCID: https://orcid.org/0000-0002-3515-2026

YULIIA VAKULENKO
PhD, Associate Professor
Department of Information Systems and Technologies
Poltava State Agrarian University
ORCID: https://orcid.org/0000-0002-6315-0116

ANDRII LEBEDYNSKYI
PhD, Associate Professor
Department of Computer Science and Information Systems
Kharkiv National Automobile and Highway University
ORCID: https://orcid.org/0000-0002-5086-8209

CHAPTER 4

ANDRII SHYSHATSKYI
Doctor of Technical Sciences, Professor, Senior Researcher
Department of Computer Science and Information Systems
Kharkiv National Automobile and Highway University
ORCID: https://orcid.org/0000-0001-6731-6390

GANNA PLEKHOVA
PhD, Associate Professor, Head of Department
Department of Computer Science and Information Systems
Kharkiv National Automobile and Highway University
ORCID: https://orcid.org/0000-0002-6912-6520

DANYLO PLIEKHOV
Assistant
Department of Automation and Computer-Aided Technologies
Kharkiv National Automobile and Highway University
ORCID: https://orcid.org/0009-0004-7873-1716

OLEKSII NALAPKO
PhD, Head of Research Department
Research Department for Development of Electronic Warfare Equipment
Central Scientifically-Research Institute of Armaments and Military Equipment of the Armed Forces of Ukraine
ORCID: https://orcid.org/0000-0002-3515-2026

OKSANA DMYTRIIEVA
Doctor of Economic Sciences, Professor, Head of Department
Department of Economics and Entrepreneurship
Kharkiv National Automobile and Highway University
ORCID: https://orcid.org/0000-0001-9314-350X

IVAN STARYNSKYI
Senior Researcher
Research Department
Institute of Information and Communication Technologies and Cyber Defense
The National Defense University of Ukraine
ORCID: https://orcid.org/0000-0003-2001-7718

CHAPTER 5

ANDRII SHYSHATSKYI
Doctor of Technical Sciences, Professor, Senior Researcher
Department of Computer Science and Information Systems
Kharkiv National Automobile and Highway University
ORCID: https://orcid.org/0000-0001-6731-6390

OLEKSANDR ZHUK
Doctor of Technical Sciences, Professor, Head of Department
Department of Communication Technologies and Cyber Protection
The National Defense University of Ukraine
ORCID: https://orcid.org/0000-0002-3546-1507

PAVLO ZHUK
PhD, Associate Professor, Head of Institute
Professional Military Education Institute "Leadership School"
The National Defense University of Ukraine
ORCID: https://orcid.org/0000-0002-9628-8074

ANDRII VERETNOV
PhD, Leading Researcher
Research Department
Central Scientifically-Research Institute of Armaments and Military Equipment of Armed Forces of Ukraine
ORCID: https://orcid.org/0000-0003-0160-7325

OLENA SHAPOSHNIKOVA
PhD, Associate Professor
Department of Computer Science and Information Systems
Kharkiv National Automobile and Highway University
ORCID: https://orcid.org/0000-0002-0405-8205

YAROSLAV MELNYK
Deputy Head of Center
Center for Simulation Modeling
The National Defense University of Ukraine
ORCID: https://orcid.org/0000-0002-2919-9119

CHAPTER 6

OLEH SHKNAI
PhD, Senior Researcher, Leading Researcher
Research Department
Scientific-Research Institute of Military Intelligence
ORCID: https://orcid.org/0000-0002-5572-4917

ILLIA DMYTRIIEV
Doctor of Economic Sciences, Professor
Department of Management
Kharkiv National Automobile and Highway University
ORCID: https://orcid.org/0000-0001-8693-3706

OLEG SOVA
Doctor of Technical Science, Professor, Head of Center
Simulation Modeling Center
The National Defense University of Ukraine
ORCID: https://orcid.org/0000-0002-7200-8955

ANDRII SHYSHATSKYI
Doctor of Technical Sciences, Professor, Senior Researcher
Department of Computer Science and Information Systems
Kharkiv National Automobile and Highway University
ORCID: https://orcid.org/0000-0001-6731-6390

OLESIA ZHUK
PhD, Associate Professor, Leading Researcher
Strategic Communications Institute
The National Defense University of Ukraine
ORCID: https://orcid.org/0000-0002-8974-0309

BOHDAN MOLODETSKYI
PhD, Chief Specialist
Scientific-Research Institute of Military Intelligence
ORCID: https://orcid.org/0000-0002-2704-7963

ABSTRACT

The high dynamism of the development of social processes and phenomena determines the formation of new systems of humanity's worldview, modification (change) of the hierarchy of needs and values, challenges to the pace and quality of development.

Addressing the over-complicated problems associated with meeting the demands of modern times requires the application of innovative scientific approaches. Today, the use of modern intelligent technologies such as artificial neural networks, deep learning and artificial intelligence is a prerequisite for the active development of all spheres of human activity: medicine, technology, business, environmental protection, education, transport and communication, etc.

Thus, it is the intellectualization of technical and management systems that can be considered one of the key bases of the new paradigm of science and technology. The phrase "of the artificial intelligence system is clear and known to everyone today. The context of this term is associated with concepts such as robotics, forecasting, processing of large information flows, expert systems, diagnostics, projects "smart house" or "smart tools", cyber-physical space and cyber-physical systems, computer translation, etc.

There is a positive dynamics of the development and implementation of elements of artificial intelligence in most types of software: mobile applications, information systems, electronic devices, etc.

This process of "intellectualization" allows to talk about the gradual increase in the intelligence of modern computer systems capable of performing functions that are traditionally considered intellectual: understanding language, logical inference, using accumulated knowledge, learning, pattern recognition, as well as learning and explaining your decisions.

The monograph contains methods of processing heterogeneous data, a scientific-methodical apparatus for assessing the state of complex technical systems, as well as a scientific-methodical apparatus for ensuring the functional reliability of special purpose information systems.

Separate chapters present mathematical models of the functioning of information systems under the influence of a set of destabilizing factors, as well as a polymodel complex of resource management of intelligent decision support systems. The specified polymodel complexes allow modeling of the functioning process of intelligent decision support systems and information systems. This is important when modeling the processes of solving complex calculation tasks in the specified systems due to the conceptual description of the specified processes.

The authors' research is supported by appropriate analytical expressions, graphic dependencies, as well as tabular values.

The monograph will be useful for researchers dealing with issues of solving optimization problems, using the theory of artificial intelligence, developing new (improving existing) approaches to solving complex technical problems in various fields of human activity.

The monograph is also useful for practitioners – designers, developers implementing modern solutions in the field of information technologies, engaged in the development of information, information-analytical, as well as automated systems for the purpose of creating new schemes and algorithms, their adaptation to non-stereotypical conditions of use, including for the implementation of artificial intelligence methods in conditions of autonomous operation, limitation of computing resources, remote control, etc.

KEYWORDS

Intelligent systems, decision support systems, artificial intelligence, modeling.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

CIRCLE OF READERS AND SCOPE OF APPLICATION

The monograph will be useful for researchers dealing with issues of solving optimization problems, using the theory of artificial intelligence, developing new (improving existing) approaches to solving complex technical problems in various fields of human activity.

The monograph is also useful for practitioners – designers, developers implementing modern solutions in the field of information technologies, engaged in the development of information, information-analytical, as well as automated systems for the purpose of creating new schemes and algorithms, their adaptation to non-stereotypical conditions of use, including for the implementation of artificial intelligence methods in conditions of autonomous operation, limitation of computing resources, remote control, etc.

INTRODUCTION

The high dynamism of the development of social processes and phenomena determines the formation of new systems of humanity's worldview, modification (change) of the hierarchy of needs and values, challenges to the pace and quality of development.

Solving the complex problems associated with meeting the demands of modernity requires the application of innovative scientific approaches. Today, the use of modern intelligent technologies such as artificial neural networks, deep learning and artificial intelligence is a prerequisite for the active development of all spheres of human activity: medicine, technology, business, environmental protection, education, transport and communication, etc. Thus, it is the intellectualization of technical and management systems that can be considered one of the key bases of the new paradigm of science and technology. The phrase "of the artificial intelligence system" is known to everyone today. The context of this term is associated with concepts such as robotics, forecasting, processing of large information flows, expert systems, diagnostics, projects "smart house" or "smart tools", cyber-physical space and cyber-physical systems, computer translation, etc.

There is positive dynamics of the development and implementation of elements of artificial intelligence in most types of software: mobile applications, information systems, electronic devices, etc.

The process of "intellectualization" allows to assert the gradual increase in the intelligence of modern computer systems capable of performing functions that are traditionally considered intellectual: language understanding, logical inference, use of accumulated knowledge, learning, pattern recognition, as well as learning and explaining your decisions.

Structurally and logically, the monograph is divided into 6 interconnected parts.

In the first chapter of the monograph, a scientific and methodological apparatus for processing heterogeneous data in decision support systems is proposed. This study is based on the theory of artificial intelligence, namely evolving artificial neural networks, basic procedures of the genetic algorithm, as well as improved combined bio-inspired algorithms. The specified scientific and methodological apparatus makes it possible to increase the efficiency of processing heterogeneous data while maintaining the required level of reliability.

The second chapter of the monograph presents a set of mathematical models of resource management of information systems. The proposed polymodel complex allows to describe the processes of functioning of information systems of various types, and also allows to model the process of managing their resources.

In the third chapter of the monograph, the scientific and methodological apparatus of intellectual assessment of parameters in decision support systems is given. An example of the use of the proposed scientific and methodological apparatus was carried out using the example of the evaluation of the parameters of decision support systems, which showed an increase in the reliability of the evaluation of its parameters at the level of 25% due to the use of additional procedures while maintaining the given level of efficiency.

In the fourth part of the monograph, a polymodel complex of resource management of intelligent decision support systems is proposed. This polymodel complex allows modeling the processes of solving complex calculation tasks in intelligent decision support systems due to the conceptual description of the specified processes.

In the fifth chapter of this monograph, the development of methods for evaluating complex organizational and technical systems using the theory of artificial intelligence was carried out. The proposed methods provide an average increase in efficiency from 16% to 23%, while ensuring high convergence of the obtained results at the level of 93.17%.

In the sixth chapter of the monograph, the development of a scientific and methodological apparatus for ensuring the functional reliability of special purpose information systems was carried out. The originality of the study consists in the construction of multidimensional dependencies of the state of functional reliability of special purpose information systems, which achieves an assessment of their functional reliability based on an arbitrary number of indicators.

DOI: 10.15587/978-617-8360-23-8.CH1

1 SCIENTIFIC AND METHODOLOGICAL FRAMEWORK FOR PROCESSING HETEROGENEOUS DATA IN DECISION SUPPORT SYSTEMS

Yurii Zhuravskyi, Dmytro Fedorchuk, Oleksandr Perehuda, Mykola Romanchuk, Roman Stavisiuk, Dmitro Stupak

ABSTRACT

This section of the study presents a scientific and methodological framework for processing heterogeneous data within decision support systems. The research is grounded in the theory of artificial intelligence, specifically focusing on evolving artificial neural networks, fundamental procedures of genetic algorithms, as well as advanced hybrid bio-inspired algorithms.

In the course of the study, the authors propose the following:

– a method for processing heterogeneous data in organizational and technical systems;

– a method for evaluating the reliability of special-purpose radio communication systems using artificial intelligence theory.

The implementation of the proposed scientific and methodological framework enables the following:

– reduction of the probability of premature convergence of the metaheuristic algorithm within decision support systems;

– maintenance of a balance between convergence speed and diversity of the metaheuristic algorithm during decision-making processes;

– consideration of the type of uncertainty and data noise in the metaheuristic algorithm when operating within decision support systems;

– accounting for available computational resources of the decision support system;

– prioritization of search processes by agents within the swarms of the metaheuristic algorithm;

– initialization of swarm individuals with consideration of the type of uncertainty present in the system;

– precise training of individuals in metaheuristic algorithms;

– execution of both local and global searches considering the level of noise in the data describing the analyzed object;

– application as a universal tool for assessing the state of analysis objects through hierarchical object representation;

– verification of the reliability of the obtained results;

– enhancement of the reliability of object state assessments by constructing object-oriented and relational models of the object's state with varying levels of hierarchy;

– avoidance of the local optimum problem.

KEYWORDS

Bio-inspired algorithms, multi-agent systems, hybrid systems, reliability, efficiency, decision support systems.

The problem of enhancing the efficiency of heterogeneous data processing in decision support systems (DSS) has become increasingly urgent in modern information and automated systems of various functional purposes. The experience of recent conflicts involving the use of advanced information and automated systems demonstrates that existing methods of processing heterogeneous data allow for the processing of only 5 to 10% of the data circulating within these systems [1–5]. This limitation is attributed to several factors:

– the significant role of the human factor in the processing of heterogeneous data circulating in information and automated systems [6–10];

– the large number of heterogeneous information sources integrated into information and automated systems [11–16];

– the processing of heterogeneous data under conditions of uncertainty, which intr– oduces delays in data handling [17–21];

the presence of a substantial amount of destabilizing data that adversely affects the speed of heterogeneous data processing [22–27];

– the coexistence of structured and unstructured data within these systems, both of which require processing [29–33].

Given the diversity, the abundance of destabilizing factors, and the varying dimensionality of the indicators describing them, the need to process large volumes of heterogeneous data necessitates the development of novel approaches. One such approach is the use of metaheuristic algorithms [34–38]. While the use of canonical metaheuristic algorithms improves the efficiency of heterogeneous data processing, further improvements are limited if only their standard forms are employed [39–44]. This necessitates the introduction of diverse strategies to enhance the convergence speed and accuracy of core metaheuristic algorithms when processing heterogeneous data [44–49]. One way to improve the efficiency of heterogeneous data processing using metaheuristic algorithms is through their further refinement—by combining, comparing, and developing new procedures for their integrated application [50–54]. An analysis of previous studies [1–71] reveals several common shortcomings:

– lack of capability to construct a hierarchical system of indicators for evaluating heterogeneous data processing;

– lack of consideration for the computational resources of the system managing the data processing;

– absence of mechanisms to adjust the set of indicators governing the data processing;

– lack of mechanisms for deep learning of knowledge bases;

– high computational complexity;

– insufficient consideration of available computational (hardware) resources in DSS;

– absence of search prioritization mechanisms in specific directions.

1.1 DEVELOPMENT OF A METHOD FOR PROCESSING HETEROGENEOUS DATA IN ORGANIZATIONAL AND TECHNICAL SYSTEMS

The objective of this study is to develop a method for processing heterogeneous data in organizational and technical systems. This will enhance the efficiency of data processing within such systems while ensuring a predetermined level of reliability and enabling subsequent managerial decisions based on the processed heterogeneous data. It will also serve as a foundation for the development (or improvement) of software tools tailored for heterogeneous data processing in organizational and technical systems.

To achieve this objective, the following tasks were defined:

– to determine the implementation algorithm of the proposed method;

– to present a practical example of applying the method to process heterogeneous data in organizational and technical systems.

The object of the study is heterogeneous data within organizational and technical systems.

The problem addressed in the research is increasing the speed of processing heterogeneous data in organizational and technical systems while maintaining a specified level of reliability, regardless of the data volume received at the system's input.

The subject of the research is the process of processing heterogeneous data, which involves:

– an improved hybrid algorithm that increases processing efficiency through a competition strategy among individuals within the hybrid algorithm;

– evolving artificial neural networks used for deep learning of the knowledge base in a multi-agent system, enabling the training of both parameters and architecture of the artificial neural networks.

The hypothesis of the study posits that it is possible to increase the speed of decision-making during the processing of heterogeneous data – while maintaining a specified reliability level – through the application of an improved hybrid algorithm.

The proposed method was modeled in the Microsoft Visual Studio 2022 (USA) software environment. The simulation task focused on determining the composition of a military (force) grouping. The hardware used for the research was based on the AMD Ryzen 5 processor.

Parameters of the improved algorithm:

1. Number of iterations: 50.

2. Number of individuals in the swarm: 25.

3. Feature space range: [–150, 150].

The method for processing heterogeneous data in organizational and technical systems includes the following sequence of steps:

Step 1. Input of initial data.

At this stage, the initial data regarding the organizational and technical system are entered, including: the number and types of components within the system, the type of data circulating through the system, available computational resources, the quantity and nature of interconnections between system elements, technical specifications of control and data transmission channels, environmental application parameters, and more.

Step 2. Initialization and formation of each agent group within the hybrid algorithm.

At this stage, initial random solution sets are generated to represent agent groups of the hybrid algorithm.

The mathematical representation of a randomly selected group of agents from the set of possible agents within a defined area is described as follows

$$P_{i,j}=P_{i,j}^{\min}+\left(\lambda\left(P_{i,j}^{\max}-P_{i,j}^{\min}\right)\right)\gamma, \tag{1.1}$$

where λ – a random number within the interval [0, 1], $P_{i,j}$ – i-th element of the j-th agent group within the hybrid algorithm. The population of hybrid algorithm agents is arranged in ascending order based on $f\left(P_i\right)$, the best and worst solutions are selected $\left(P_i^{best}\right)$ and $\left(P_i^{worst}\right)$, γ – represents the degree of uncertainty in the data circulating within the organizational and technical system. At this stage, the objective function for heterogeneous data processing *f(P)* is defined, along with the population size *m* of the hybrid algorithm's swarm, the number of variables *n*, bounds for the variables *(LB, UB)*, and the termination criterion (FE_{max}') of the algorithm.

Step 3. Numbering of hybrid algorithm agents within the population, i, where $i \in [0, S]$.

Each agent in the hybrid algorithm's swarm is assigned a sequential identifier.

Step 4. Initialization of the agents' initial velocities.

The initial velocity v_0 for each agent in the population is computed using the following expression

$$v_i=\left(v_1,v_2...v_S\right),\ v_i=v_0. \tag{1.2}$$

Step 5. Preliminary assessment of the search area by the hybrid algorithm's swarm agents.

In this step, the search area is defined in natural language as the aura surrounding each group of the hybrid algorithm's swarm.

Step 6. Classification of food sources for the swarm agents.

The best food source (i.e., the one with minimal fitness value) is labeled as (FS_{ht}), representing nearby and energy-efficient resources. High-cost, desirable food sources are labeled as $FS_{at,}$ while survival-level, non-priority resources are marked as $FS_{nt.}$:

$$FS_{ht}=FS\left(sorte\_index\left(1\right)\right), \tag{1.3}$$

$$FS_{at}\left(1:4\right)=FS\left(sorte\_index\left(1:3\right)\right), \tag{1.4}$$

$$FS_{nt}\left(1:NP-4\right)=FS\left(sorte\_index\left(6:NP\right)\right). \tag{1.5}$$

Step 7. Execution of the cheetah swarm algorithm procedures.

Step 7.1 Search behavior of cheetah agents.

This behavior simulates the process of prey scouting. Cheetahs apply two main strategies: waiting and active exploration. In this context, the search space represents the problem-solving space for heterogeneous data processing. The behavior is modeled by

$$X_{i,j}^{t+1}=X_{i,j}^{t}+\hat{r}_{i,j}^{-1}\alpha_{i,j}^{t}, \tag{1.6}$$

where $X_{i,j}^{t}$ – denotes the current position of agent *i* or individual cheetah agents *i* (*i*=1, 2..., *n*) in population *j* or in the search space *j* (*j*=1, 2..., *d*), where *n* – the number of cheetah agents in the population, and d is the dimensionality of the optimization problem. $X_{i,j}^{t+1}$ – the next position of the cheetah agent, t – the cheetah agents' hunting time, T – the maximum hunting time, $\hat{r}_{i,j}^{-1}$ – randomization parameter, $\alpha_{i,j}^{t}$ – the step size for cheetah agent i in population j. The randomization parameter $\hat{r}_{i,j}$ follows a standard normal distribution. The step size $\alpha_{i,j}^{t} > 0$ is set to 0.001 x t/T, as cheetahs are slow searchers.

Step 7.2. Waiting strategy of cheetah agents.

The waiting strategy is formally defined as follows

$$X_{i,j}^{t+1} = X_{i,j}^{t}, \tag{1.7}$$

where $X_{i,j}^{t+1}$ $X_{i,j}^{t}$ – represents the new position of the cheetah agent i in population j, respectively. The proposed strategy introduces a specific coordination approach in the optimization algorithm, where all agents in the group attack simultaneously. This significantly increases the probability of successful hunting and reduces the risk of premature convergence toward food sources.

Step 7.3. Active attack behavior of cheetah agents.

This procedure outlines two key characteristics of the active hunting stage: speed and flexibility. It is important to note that cheetahs do not use group tactics while hunting, and all their attack strategies can be mathematically represented as

$$X_{i,j}^{t+1} = X_{B,j}^{t} + \hat{r}_{i,j} \cdot \beta_{i,j}^{t}, \tag{1.8}$$

where $X_{B,j}^{t}$ – the current prey position in search space j; $\hat{r}_{i,j}$, $\beta_{i,j}^{t}$ – a rotation and interaction factors associated with cheetah i in search space j, $\hat{r}_{i,j}$ – a random value described by $\left|\hat{r}_{i,j}\right|^{\exp\left(r_{i,j}/2\right)} \sin\left(2\pi r_{i,j}\right)$ and $r_{i,j}$ – a random number from the normal distribution.

Step 8. Execution of particle swarm algorithm procedures.

8.1. Velocity update of particle swarm agents.

The velocity generation of particle swarm agents is calculated based on two parameters: the global best particle G_{best} and the local best particle L_{best}. Velocities are updated using the following equation

$$V_i^{t+1} = \omega V_i^{t} + c_1 r_{1i}^{t}\left(L_{best}^{t} - x_i^{t}\right) + c_2 r_{2i}^{t}\left(G_{best}^{t} - x_i^{t}\right), \tag{1.9}$$

where V_i^{t+1} – the particle's velocity at iteration (t+1), V_i^{t} – the velocity at the previous iteration, x_i^{t}, r_{1i}^{t}, r_{2i}^{t} – d-dimensional vectors of uniformly distributed random numbers between 0 and 1, representing the position of particle i; c_1 and c_2 – learning coefficients, and ω – the inertia weight, typically set to 1.

8.2. Particle movement in the search space.

The new position of the particles is generated according to equation (1.10), where x_i^{t+1} – the new particle position, x_i^{t} – the previous position, and V_i^{t+1} – the velocity calculated using equation (1.9)

$$x_i^{t+1} = x_i^{t} + V_i^{t+1}. \tag{1.10}$$

Step 9. Integration of search strategies from both algorithms.

After generating the initial population, each agent receives a subpopulation size equal to half of the original population, as defined in this study. The application process of metaheuristic operators is simplified by the sequential execution of cheetah swarm behavior and particle swarm algorithm procedures. The integration procedure of both behavioral strategies is modified as follows

$$x_{i+1}^k = x_i^k \alpha + 1 - \alpha x_{best}^k M_i^k. \tag{1.11}$$

where x_{i+1}^k – the new candidate solution position x_i^k. The scaling factor α set to 0.1 in this study, x_{best}^k – the best solution at iteration k; M_i^k – the modulation variable of the candidate from the agent swarm. Equation (1.8) defines the combined swarm population of the hybrid algorithm, which exhibits the best performance.

Step 9.1. Modulation of metaheuristic operators.

In this study, the influence modulation of each metaheuristic operator is determined not only by comparing it to the best candidate solution but also by analyzing its elite behavior. The competition begins by identifying the solution x_c^k based on the actual solution obtained x_i^k. The solution x_c^k must only satisfy one condition: x_c^k it must be different from x_i^k.

Step 9.2. Pairwise competition of agent groups in the hybrid algorithm.

The group competition procedure in the hybrid algorithm is defined by equation (1.12):

$$\begin{aligned} &\text{if } f\left(x_i^k\right) < f\left(x_c^k\right) \text{ then } x_i^k = x_c^k \text{ and } M_i^k; \\ &\text{if } \left(x_i^k\right) > f\left(x_c^k\right) \text{and } Pr > r \text{ then } x_i^k = G\left(x_c^k\right) \text{ and } M_i^k = M_c^k. \end{aligned} \tag{1.12}$$

Additionally, the probabilistic threshold is determined by the performance difference between the obtained solution and the best-known solution, which varies during iterations. This threshold is computed as

$$Pr = \left| \frac{f\left(x_i^k\right) - f\left(x_c^k\right)}{BF} \right|, \tag{1.13}$$

where Pr – the probabilistic threshold, x_i^k – the actual solution, x_c^k – the reference solution, and BF – the objective value of the obtained solution. The new position x_i^k is determined based on the Euclidean distance between x_i^k and x_c^k and is updated using equation (1.14)

$$r \cdot dist - x_c^k, \tag{1.14}$$

where r – a normally distributed random number, and $dist$ – the Euclidean distance between x_i^k and x_c^k. It is worth noting that the described procedure facilitates the exploration of new regions within the solution search space x_i^k. This approach prevents premature convergence and ensures a thorough evaluation of the algorithm's computational capabilities.

Step 10. Termination condition check for hybrid swarm agents.

The algorithm terminates when the maximum number of iterations is reached. Otherwise, new candidate positions are generated, and the process is repeated.

Step 11. Knowledge base learning for hybrid swarm agents.

In the proposed study, the knowledge base of each agent in the hybrid swarm algorithm is trained using the evolving artificial neural network method described in [2]. This method modifies the movement behavior of each agent in the hybrid algorithm, contributing to more accurate analytical outcomes.

Step 12. Determination of required computational resources for the intelligent decision support system.

To prevent computational loops through *Steps 1–11* and to improve computational efficiency, the system workload is monitored. If a defined computational complexity threshold is exceeded, the number of additional hardware and software resources required is determined using the method proposed in [23].

End of Algorithm.

Proposed method for processing heterogeneous data in organizational-technical systems.

The efficiency of the proposed method for processing heterogeneous data in organizational-technical systems is compared using a set of benchmark functions, the structure of which is presented in **Table 1.1**.

Table 1.1 Evaluation of the efficiency of the proposed method for processing heterogeneous data according to the criterion of information processing speed

Function name	Metric	Canonical particle swarm algorithm	Ant colony algorithm	Black widow algorithm	Gray wolf pack algorithm	Cheetah pack algorithm	Proposed method
1	2	3	4	5	6	7	8
U22-1	Average value	300.000	300.000	300.000	300.000	300.000	300.000
	Standard value	2.17547E-07	1.94448E-07	1.73866E-07	1.73121E-07	1.51021E-07	1.72168E-07
B22-2	Average value	400	400.265772	400.7973158	400.265772	400.3986579	400.5315439
	Standard value	4.9898E-08	1.011427534	1.621892282	1.011427535	1.216419212	1.378343398
B22-3	Average value	600.0071815	600.0644622	600.0240021	600.012832	600.031303	600.0449987
	Standard value	0.021632777	0.184980091	0.115606243	0.053463097	0.147011513	0.101164243
B22-4	Average value	826.5653461	827.3281442	823.8789639	826.3000191	826.2668486	825.7693662
	Standard value	9.13817552	8.364210734	11.30806963	8.186625055	9.136107323	10.05991317
B22-5	Average value	900.743876	900.9504411	900.9726169	900.8007883	900.5452042	901.2016312
	Standard value	0.781626306	1.424558753	1.275779755	0.903385622	0.635781924	1.598982565
B22-6	Average value	1888.524629	1874.869967	1876.294359	1847.184924	1888.926953	1842.878175
	Standard value	127.2561383	91.22185049	69.00003268	32.76980351	140.693674	31.32108747

Continuation of Table 1.1

1	2	3	4	5	6	7	8
H22-7	Average value	2027.479588	2030.758499	2029.556604	2032.238674	2028.177978	2029.128603
	Standard value	6.106897592	8.027195324	5.81348717	7.446489204	8.003968446	8.197733191
H22-8	Average value	2223.108804	2223.537417	2222.070633	2223.140251	2220.888475	2220.690533
	Standard value	4.749655105	2.963408213	4.895282849	3.995669404	5.451654006	6.337353983
H22-9	Average value	2510.930321	2510.930321	2536.358938	2498.216012	2523.644629	2498.216012
	Standard value	65.93880108	65.93880108	85.778947	48.38585173	77.58997694	48.38585173
C22-10	Average value	2594.615905	2596.833927	2585.256107	2591.210109	2605.304194	2619.308989
	Standard value	48.2013289	49.71807546	57.1034079	56.36586785	42.57395199	34.10382553
C22-11	Average value	2695.981932	2685.587394	2733.855734	2710.621315	2700.168413	2715.332781
	Standard value	116.3652035	110.1475838	146.333679	118.5098748	113.7913849	109.3008673
C22-12	Average value	2857.067086	2858.742176	2854.959949	2861.414681	2859.407788	2860.718769
	Standard value	9.364347909	14.88960231	5.539104327	17.96133754	15.00545163	16.34731781

Table 1.2 presents the results of the reliability assessment of decisions made by each of the optimization methods for processing heterogeneous data in organizational-technical systems.

Table 1.2 Evaluation of the proposed management method's efficiency based on the information processing reliability criterion

Function name	Metric	Canonical particle swarm algorithm	Ant colony algorithm	Black widow algorithm	Gray wolf pack algorithm	Cheetah pack algorithm	Proposed method
1	2	3	4	5	6	7	8
U22-1	Average value	0.66	0.73	0.67	0.68	0.8	0.9
	Standard value	0.7	0.73	0.68	0.69	0.83	0.91
B22-2	Average value	0.7	0.73	0.7	0.71	0.77	0.89
	Standard value	0.71	0.73	0.72	0.72	0.76	0.9
B22-3	Average value	0.68	0.73	0.7	0.71	0.76	0.92
	Standard value	0.69	0.73	0.69	0.73	0.77	0.91
B22-4	Average value	0.67	0.74	0.7	0.72	0.78	0.93
	Standard value	0.67	0.72	0.67	0.72	0.79	0.92

Continuation of Table 1.2

1	2	3	4	5	6	7	8
B22-5	Average value	0.6	0.71	0.64	0.73	0.8	0.91
	Standard value	0.61	0.72	0.64	0.74	0.88	0.92
B22-6	Average value	0.64	0.73	0.66	0.77	0.85	0.93
	Standard value	0.66	0.75	0.66	0.78	0.83	0.92
H22-7	Average value	0.67	0.72	0.68	0.75	0.81	0.9
	Standard value	0.68	0.71	0.69	0.74	0.83	0.9
H22-8	Average value	0.68	0.74	0.69	0.75	0.84	0.93
	Standard value	0.65	0.74	0.67	0.77	0.81	0.91
H22-9	Average value	0.64	0.75	0.66	0.69	0.83	0.91
	Standard value	0.7	0.72	0.71	0.71	0.84	0.93
C22-10	Average value	0.69	0.71	0.7	0.72	0.8	0.94
	Standard value	0.68	0.71	0.7	0.73	0.8	0.91
C22-11	Average value	0.67	0.71	0.69	0.71	0.82	0.91
	Standard value	0.67	0.72	0.68	0.74	0.91	0.91
C22-12	Average value	0.63	0.73	0.65	0.75	0.82	0.91
	Standard value	0.62	0.74	0.66	0.76	0.83	0.91

An analysis of **Tables 1.1** and **1.2** shows that the proposed method ensures stable algorithm performance for both unimodal and multimodal benchmark functions.

As illustrated in **Tables 1.1, 1.2**, the improvement in decision-making speed reaches 14–20% due to the application of additional procedures and the reliability of the obtained decisions being maintained at a level of 0.9.

The advantages of the proposed method are as follows:

– the initial population of agents in the hybrid swarm algorithm and their starting positions in the search space are determined considering the uncertainty degree of the initial data circulating in the organizational-technical system (1.1), through the application of correction coefficients. This contrasts with approaches in [9, 14, 20] and allows for reduced time in configuring the heterogeneous data processing subsystem during its initial setup;

– the initial velocity of each agent in the hybrid swarm is taken into account (1.2), enabling the prioritization of searches in specific dimensions of the search space (by elements and components of the organizational-technical system), compared to methods in [9–15];

– the suitability of the decisions made during heterogeneous data processing is evaluated considering the aggregate of external factors, thereby reducing the overall decision search time (*Step 5*), compared to [14, 16, 17];

– the search strategies of food source localization for agents in the hybrid swarm algorithm are versatile, allowing classification of the conditions and factors influencing heterogeneous data processing (*Step 6*), compared to [14, 16, 17]. This enables the identification of the most suitable decision-making options according to the defined optimization criterion;

– the method allows exploration of solution spaces defined by non-standard functions by using a step selection procedure for cheetah agents within the hybrid swarm algorithm (*Step 7*), in contrast to [9, 12–18];

– replacement of unfit agents is possible via population updating mechanisms of the hybrid swarm algorithm (*Steps 8–10*), compared to [9, 12–18];

– the method supports comparative assessment of heterogeneous data processing efficiency using the metaheuristic operator modulation procedure (*Step 9.1*), as compared to [20].

– the capability of simultaneously searching for solutions in multiple directions is supported (*Steps 1–12*, **Tables 1.1** and **1.2**).

– the method provides for deep learning of the knowledge bases of agents in the hybrid swarm algorithm (*Step 10*), compared to [9–20];

– the required number of computational resources can be estimated in cases where available computing capacity is insufficient (*Step 12*), compared to [9–20].

The drawbacks of the proposed method include:

– loss of informativeness when processing heterogeneous data due to the construction of a membership function;

– lower accuracy in evaluating individual parameters of the heterogeneous data processing state;

– decreased reliability of the obtained decisions when searching in multiple directions simultaneously;

– lower evaluation precision compared to other methods for processing heterogeneous data.

The proposed method enables:

– determination of the optimal efficiency indicator for heterogeneous data processing based on the selected optimization criterion;

– identification of effective measures to improve the efficiency of heterogeneous data processing;

– increased speed of heterogeneous data processing while maintaining the required decision-making reliability;

– reduced use of computational resources in decision support systems.

The limitations of the study include the requirement for information on the degree of uncertainty in the data circulating within organizational-technical systems, and the need to consider delays in the collection and dissemination of information from the system components.

The proposed approach is recommended for solving problems related to the processing of heterogeneous data characterized by a high degree of complexity.

1.2 METHOD FOR ASSESSING THE RELIABILITY OF SPECIAL-PURPOSE RADIO COMMUNICATION SYSTEMS USING ARTIFICIAL INTELLIGENCE THEORY

Radio communication systems currently serve as the core of the transmission environment and are used to transfer all types of traffic. The method for assessing the reliability of special-purpose radio communication systems (SP-RCS) using artificial intelligence theory consists of the following sequence of steps:

Step 1. Input of initial data.

At this stage, the available initial data concerning the SP-RCS and the enemy's electronic warfare (EW) assets are entered, specifically:

– the quantity and types of radio communication assets included in the system;

– the quantity and types of enemy EW assets;

– technical characteristics of the SP-RCS;

– technical characteristics of the EW assets;

– architecture (topology of connections) of the SP-RCS;

– architecture (topology of connections) of the EW assets;

– types of data circulating within the system;

– available computational resources;

– information about the operating environment, etc.

Step 2. Initialization and formation of the general agent population for the hybrid swarm algorithm.

At this stage, initial random sets of solutions are generated, representing groups of agents in the hybrid swarm algorithm.

The mathematical representation of a randomly selected group of agents from the hybrid algorithm, taken from the set of all possible agents within a defined area, is described as follows

$$P_{i,j} = P_{i,j}^{\min} + \left(\lambda\left(P_{i,j}^{\max} - P_{i,j}^{\min}\right)\right)\gamma, \tag{1.15}$$

where λ – a random number in the range [0, 1], $P_{i,j}$ – *i*-th identifier of the *j*-th agent group of the hybrid algorithm. The agents of the hybrid algorithm are arranged in ascending order of values $f(P_i)$, with the best $\left(P_i^{best}\right)$ and the worst solutions selected $\left(P_i^{worst}\right)$. γ represents the degree of uncertainty regarding the enemy's electronic warfare (EW) assets. At this stage, the target reliability function *f(P)* is also defined, as well as the population size *(m)* of the hybrid swarm, the number of variables *(n)*, bounds on variable values *(LB, UB)*, and the termination criterion for the algorithm ($FE_{\max}'$).

Step 3. Numbering of hybrid algorithm agents in the population, i, $i \in [0, S]$.

Each agent in the hybrid swarm algorithm is assigned a sequential identifier within the population.

Step 4. Determination of initial agent velocity in the population.

The initial velocity v_0 of each agent in the population is calculated using the following expression

$$v_i = \left(v_1, v_2, \ldots, v_s\right),\ v_i = v_0. \tag{1.16}$$

Step 5. Preliminary evaluation of the search area by hybrid swarm agents.

In this step, the search area is linguistically defined as the *aura* surrounding each group of agents in the hybrid swarm algorithm.

Step 6. Classification of food sources for hybrid swarm agents.

The location of the best food source (i.e., the one with the lowest fitness value) is denoted as (FS_{ht}) a source that is nearby and requires minimal energy to locate and acquire. Delicacy food sources, which demand the most effort to obtain, are marked as FS_{at}.

Other non-priority food sources (required only for individual survival) are denoted as FS_{nt}, defined as follows:

$$FS_{ht} = FS\left(sorte\_index\left(1\right)\right), \tag{1.17}$$

$$FS_{at}\left(1:4\right) = FS\left(sorte\_index\left(1:3\right)\right), \tag{1.18}$$

$$FS_{nt}\left(1:NP-4\right) = FS\left(sorte\_index\left(6:NP\right)\right). \tag{1.19}$$

Step 7. Execution of calculations by individual hybrid swarm agent groups.

Step 7.1. Execution of calculations by dung beetle agents.

This algorithm mimics the natural behavior of dung beetles. The dung beetle algorithm (DBA) divides the entire population into four segments based on this behavior.

Step 7.1.1. Ball rolling procedure.

When dung beetles roll their dung balls, they ensure the path is linear, aligned with celestial cues. To simulate this behavior, agents in the algorithm move directionally through the entire search space. It is assumed that sunlight intensity affects the beetles' paths. The agent's position is updated as follows:

$$\begin{aligned} &X_n^{i+1} = X_n^i + a.k.X_n^{i-1} + b.\Delta x; \\ &\Delta x = \left|X_n^i - X^{\omega}\right|, \end{aligned} \tag{1.20}$$

where X_n^{i+1} – the position of the n-th dung beetle at the i-th iteration, $k \in \left(0, 0.2\right]$ – the deviation coefficient (assigned a value of 0.1 in the code), $b \in \left(0,1\right)$ – a natural coefficient (assigned a value of 0.3 in the code), x – the change in illumination intensity, and X^{ω} – the worst agent position in the current population. a – a natural coefficient assigned a value of either 1 or –1, where a = 1 indicates no effect from environmental interference on the beetle's direction, and a = –1 indicates deviation. In this study, a characterization of the level of SP-RCS suppression by EW assets.

In nature, when dung beetles encounter an obstacle, they rotate their bodies to alter direction and bypass the barrier. In our case, this simulates the reconfiguration of the information transmission route when an SP-RCS element is disabled by disruptive factors.

This is described by:

$$X_n^{i+1} = X_n^i + \tan\alpha \left| X_n^i - X_n^{i-1} \right|;$$
$$0 \le \alpha \le \pi, \tag{1.21}$$

where α – the deflection angle between the new direction of the beetle and its original path.

Step 7.1.2. Beetle reproduction procedure.

When the dung ball returns to the nest, the beetles must choose a suitable location for egg-laying to ensure a safe environment for their offspring. Based on the discussion above, the algorithm models this behavior by identifying a boundary region where females deposit eggs. This is defined as follows:

$$LB_1 = max\left(X^t.(1-T), LB\right),$$
$$UB_1 = min\left(X^t.(1+T), UB\right), \tag{1.22}$$

where X^t – the current local optimum, LB_1 and UB_1 – the lower and upper boundaries of the egg-laying region; Lb and Ub – the lower and upper bounds of the overall search space, respectively; and the inertia weight is $R = 1 - t/T_{max}$, where T_{max} – the maximum number of iterations during the algorithm's runtime.

In the context of this study, the procedure identifies the SP-RCS elements that are least suppressed.

The boundary range of the egg-laying region dynamically changes to prevent the algorithm from falling into a local optimum. Thus, during the iteration process, the positions of the laid dung balls can also shift. This process is described as

$$X_n^{i+1} = X^t + B_1.\left(X_n^i - LB_1\right) + B_2.\left(X_n^i - UB_1\right), \tag{1.23}$$

where X_n^i – the position of the n-th dung ball to be laid at iteration t, B_1 and B_2 – two independent random matrices, and D is the algorithm's dimensionality.

Step 7.1.3. Hatching of dung beetles.

Once the young dung beetles successfully hatch, they move out in groups to search for food. Their food-searching behavior is constrained by a limited range. For these young beetles, the optimal foraging area is determined as:

$$LB_2 = max\left(X^{tb}.(1-T), LB\right),$$
$$UB_2 = min\left(X^{tb}.(1-T), UB\right), \tag{1.24}$$

where X^{tb} – the global optimum, and LB_2 and UB_2 – the lower and upper bounds of the region, respectively. Once the location for the young dung beetle is determined, its position can be updated as follows

$$X_i^{n+1} = X_i^n + C_1.\left(X_i^n - LB_2\right) + C_2.\left(X_i^n - UB_2\right), \tag{1.25}$$

where X_i^{n+1} – represents the location information of the i-th young beetle at iteration t, C_1 is a Gaussian-distributed random number, and C_2 is a value in the interval (0, 1).

Step 7.1.4. Execution of the stealing strategy.

In dung beetle populations, some individuals steal dung balls from others. These beetles are referred to as "thieves". As mentioned earlier, X^s is the globally optimal location – that is, the best location for accessing food.

Therefore, it is assumed that the surroundings of X represent the most competitive region for food. The position of the dung beetle thieves is updated during the iteration process as follows

$$X_n^{i+1} = X^s + P.d.\left(\left|X_n^{i} - X^t\right| + \left|X_n^{i} - X^-\right|\right), \tag{1.26}$$

where d – a random vector of size $1.D$, following a normal distribution, and P – a constant.

Step 7.2. Execution of calculations by osprey swarm agents.

Step 7.2.1. Global exploitation.

To model the first stage of population updating for osprey agents, the natural behavior of ospreys was simulated. Equation (1.27) is used to determine the location of each osprey

$$OS_n = \{X_t \mid t \in \{1, 2, ..., N\} \wedge O_t < O_n\} \cup \{X^*\}, \tag{1.27}$$

where OS_n – the set of positions occupied by the n-th osprey, and X^* – the precise location of the optimal osprey. The osprey independently identifies the location of a fish and initiates its attack. The new position of the osprey relative to the fish is calculated based on modeled movement behavior, as described in equation (1.28). If the fitness function at the new position yields a better value, the previous position is replaced:

$$\begin{gathered} x_{i,j}^{P1} = xi,j + ri,j.\left(SF_{i,j} - I_{i,j}.x_{i,j}\right), \\ x_{i,j}^{P1} = \begin{cases} x_{i,j}^{P1}, lb_j \le x_{i,j}^{P1} \le ub_j; \\ lb_j, x_{i,j}^{P1} \langle lb_j; \\ ub_j, x_{i,j}^{P1} \rangle ub_j. \end{cases} \\ X_i = \begin{cases} X_i^{P1}, F_i^{P1} \langle F_i; \\ X_i, \text{also}, \end{cases} \end{gathered} \tag{1.28}$$

where $x_{i,j}^{P1}$ – the new position in the j-th dimension for the i-th osprey at stage one; $F_{i,j}$ – the fitness value of the current osprey position; $SF_{i,j}$ – a random number in the range [0, 1]; and $I_{i,j}$ – a random integer from the set 1, 2.

Step 7.2.2. Local exploitation.

When an osprey catches a fish, it transports it to a safe zone to consume it. The second stage of population updating uses simulation-based modeling to replicate the osprey's natural behavior. Guiding the fish to a suitable location leads to minor adjustments in the osprey's position within the search space.

This enhances the local search capability of the procedure and allows the algorithm to converge toward a more optimal solution in the vicinity of a previously defined result. This position is considered "suitable for fish consumption" and is determined using the following equation (1.29). Subsequently, if the fitness function value improves at this new position, the previous position of the corresponding offspring is replaced:

$$x_{i,j}^{P2} = x_{i,j} + \frac{lb_j + r \cdot \left(ub_j - lb_j\right)}{t},$$

$$i = 1,2,\ldots,N,\ j = 1,2,\ldots,m, t = 1,2,\ldots,T,$$

$$x_{i,j}^{P2} = \begin{cases} x_{i,j}^{P2}, lb_j \le x_{i,j}^{P2} \le ub_j; \\ lb_j, x_{i,j}^{P2} \langle lb_j; \\ ub_j, x_{i,j}^{P2} \rangle ub_j. \end{cases}$$

$$X_i = \begin{cases} X_i^{P2}, F_i^{P2} \langle F_i; \\ X_i, \text{also}, \end{cases} \tag{1.29}$$

where $x_{i,j}^{P2}$ – the new position of the j-th dimension for the i-th osprey during the second stage; $F_{i,j}^{P2}$ – the corresponding fitness value of the updated position; r is a random number in the range [0, 1]; t and T are the current and maximum number of iterations, respectively.

Step 8. Integration of the search strategies from both algorithms.

After generating the initial population, each agent receives a population size equal to half of the original population, as defined in the referenced study. The process of applying metaheuristic operators is simplified by sequentially executing the behavior of the osprey swarm algorithm and the dung beetle swarm algorithm, according to their respective procedures. The procedure for integrating both behavioral strategies is modified as follows

$$x_{i+1}^k = x_i^k \alpha + 1 - \alpha x_{best}^k M_i^k, \tag{1.30}$$

where x_{i+1}^k – the new candidate solution position x_i^k. The scaling coefficient α in this study is set to 0.1; x_{best}^k – the best solution at iteration k; M_i^k – the modulation variable of the candidate from the swarm. Equation (1.30) defines the merged population of agents in the hybrid algorithm that exhibits optimal performance.

Step 8.1. Modulation of metaheuristic operators.

In this study, the influence of each metaheuristic operator is modulated not only by the traditional comparison to the best candidate solution but also by analyzing its elite behavior. The competition begins by identifying solution x_c^k in comparison to the actual obtained solution x_i^k. The only requirement x_c^k is that it must differ from x_i^k.

Step 8.2. Pairwise competition of agent groups in the hybrid algorithm.

The group competition procedure in the hybrid algorithm is defined by Equation (1.31):

$$\text{if } f\left(x_i^k\right) < f\left(x_c^k\right) \text{ then } x_i^k = x_c^k \text{ and } M_i^k;$$
$$\text{if } \left(x_i^k\right) > f\left(x_c^k\right) \text{and } Pr > r \text{ then } x_i^k = G\left(x_c^k\right) \text{ and } M_i^k = M_c^k. \tag{1.31}$$

Additionally, the probabilistic threshold is defined as the performance difference between the obtained solution and the best solution, and it varies across iterations.

The threshold is calculated as follows

$$Pr = \left|\frac{f\left(x_i^k\right) - f\left(x_c^k\right)}{BF}\right|, \tag{1.32}$$

where Pr – the probabilistic threshold, x_i^k – the actual solution, x_c^k – the benchmark solution, and BF – the cost (fitness) of the obtained solution. The new position x_i^k is determined by calculating the Euclidean distance between x_i^k and x_c^k. A position is updated using the following formula

$$r \cdot dist - x_c^k, \tag{1.33}$$

where r – a randomly distributed number and *dist* is the Euclidean distance between x_i^k and x_c^k. It is worth noting that this procedure facilitates the exploration of new regions within the solution search space x_i^k. It prevents premature convergence and ensures a more comprehensive analysis of the algorithm's computational capabilities.

Step 9. Stopping criterion check for the hybrid swarm agents.

The algorithm terminates when the maximum number of iterations is reached. Otherwise, new positions are generated, and the condition check is repeated.

Step 10. Learning of knowledge bases for hybrid swarm agents.

In this study, the knowledge base of each agent in the hybrid swarm algorithm is trained using the evolving artificial neural network method developed in [2]. This method adjusts the movement patterns of each agent in the hybrid swarm, contributing to more accurate analytical results in future iterations.

Step 11. Determination of required computational resources for the intelligent decision support system.

To avoid computational loops through *Steps 1–10* and to increase computational efficiency, system load is monitored. If the defined computational complexity threshold is exceeded, the required number of additional software and hardware resources is determined using the method proposed in [23].

The efficiency of the proposed method for assessing the reliability of special-purpose radio communication systems using artificial intelligence theory is evaluated using a set of benchmark functions, as presented in **Table 1.3**.

Table 1.3 Evaluation of the efficiency of the proposed method for assessing the reliability of special-purpose radio communication systems using artificial intelligence theory

Function name	Metric	Canonical particle swarm algorithm	Ant colony algorithm	Black widow algorithm	Gray wolf pack algorithm	Cheetah pack algorithm	Proposed method
U22-1	Average value	300.000	300.000	300.000	300.000	300.000	300.000
	Standard value	2.17547E-07	1.94448E-07	1.73866E-07	1.73121E-07	1.51021E-07	1.72168E-07
B22-2	Average value	400	400.265772	400.7973158	400.265772	400.3986579	400.5315439
	Standard value	4.9898E-08	1.011427534	1.621892282	1.011427535	1.216419212	1.378343398
B22-3	Average value	600.0071815	600.0644622	600.0240021	600.012832	600.031303	600.0449987
	Standard value	0.021632777	0.184980091	0.115606243	0.053463097	0.147011513	0.101164243
B22-4	Average value	826.5653461	827.3281442	823.8789639	826.3000191	826.2668486	825.7693662
	Standard value	9.13817552	8.364210734	11.30806963	8.186625055	9.136107323	10.05991317
B22-5	Average value	900.743876	900.9504411	900.9726169	900.8007883	900.5452042	901.2016312
	Standard value	0.781626306	1.424558753	1.275779755	0.903385622	0.635781924	1.598982565
B22-6	Average value	1888.524629	1874.869967	1876.294359	1847.184924	1888.926953	1842.878175
	Standard value	127.2561383	91.22185049	69.00003268	32.76980351	140.693674	31.32108747
H22-7	Average value	2027.479588	2030.758499	2029.556604	2032.238674	2028.177978	2029.128603
	Standard value	6.106897592	8.027195324	5.81348717	7.446489204	8.003968446	8.197733191
H22-8	Average value	2223.108804	2223.537417	2222.070633	2223.140251	2220.888475	2220.690533
	Standard value	4.749655105	2.963408213	4.895282849	3.995669404	5.451654006	6.337353983
H22-9	Average value	2510.930321	2510.930321	2536.358938	2498.216012	2523.644629	2498.216012
	Standard value	65.93880108	65.93880108	85.778947	48.38585173	77.58997694	48.38585173
C22-10	Average value	2594.615905	2596.833927	2585.256107	2591.210109	2605.304194	2619.308989
	Standard value	48.2013289	49.71807546	57.1034079	56.36586785	42.57395199	34.10382553
C22-11	Average value	2695.981932	2685.587394	2733.855734	2710.621315	2700.168413	2715.332781
	Standard value	116.3652035	110.1475838	146.333679	118.5098748	113.7913849	109.3008673
C22-12	Average value	2857.067086	2858.742176	2854.959949	2861.414681	2859.407788	2860.718769
	Standard value	9.364347909	14.88960231	5.539104327	17.96133754	15.00545163	16.34731781

Table 1.4 presents the results of the reliability assessment of decisions obtained using each of the optimization methods for processing heterogeneous data in organizational-technical systems.

Table 1.4 Evaluation of the proposed method's efficiency based on the information processing reliability criterion

Function name	Metric	Canonical particle swarm algorithm	Ant colony algorithm	Black widow algorithm	Gray wolf pack algorithm	Cheetah pack algorithm	Proposed method
U22-1	Average value	0.66	0.73	0.67	0.68	0.8	0.9
	Standard value	0.7	0.73	0.68	0.69	0.83	0.91
B22-2	Average value	0.7	0.73	0.7	0.71	0.77	0.89
	Standard value	0.71	0.73	0.72	0.72	0.76	0.9
B22-3	Average value	0.68	0.73	0.7	0.71	0.76	0.92
	Standard value	0.69	0.73	0.69	0.73	0.77	0.91
B22-4	Average value	0.67	0.74	0.7	0.72	0.78	0.93
	Standard value	0.67	0.72	0.67	0.72	0.79	0.92
B22-5	Average value	0.6	0.71	0.64	0.73	0.8	0.91
	Standard value	0.61	0.72	0.64	0.74	0.88	0.92
B22-6	Average value	0.64	0.73	0.66	0.77	0.85	0.93
	Standard value	0.66	0.75	0.66	0.78	0.83	0.92
H22-7	Average value	0.67	0.72	0.68	0.75	0.81	0.9
	Standard value	0.68	0.71	0.69	0.74	0.83	0.9
H22-8	Average value	0.68	0.74	0.69	0.75	0.84	0.93
	Standard value	0.65	0.74	0.67	0.77	0.81	0.91
H22-9	Average value	0.64	0.75	0.66	0.69	0.83	0.91
	Standard value	0.7	0.72	0.71	0.71	0.84	0.93
C22-10	Average value	0.69	0.71	0.7	0.72	0.8	0.94
	Standard value	0.68	0.71	0.7	0.73	0.8	0.91
C22-11	Average value	0.67	0.71	0.69	0.71	0.82	0.91
	Standard value	0.67	0.72	0.68	0.74	0.91	0.91
C22-12	Average value	0.63	0.73	0.65	0.75	0.82	0.91
	Standard value	0.62	0.74	0.66	0.76	0.83	0.91

From the analysis of **Tables 1.3** and **1.4**, it can be concluded that the proposed method ensures stable algorithm performance for key unimodal and multimodal test functions.

As evident from **Tables 1.3, 1.4**, the increase in decision-making speed reaches 16–20% due to the use of additional procedures and the achievement of decision reliability at the 0.9 level.

The advantages of the proposed method are as follows:

– the initial population of agents in the hybrid algorithm swarm and their initial positions in the search space are determined considering the degree of uncertainty in the input data circulating in the organizational and technical system (equation 1.15) through the use of appropriate correction coefficients, in comparison with studies [9, 14, 20]. This reduces the time required for the initial setup of the heterogeneous data processing subsystem;

– the initial velocity of each agent in the hybrid swarm is taken into account (equation 1.16), allowing prioritization of the search in the corresponding search space (by elements and components of the organizational and technical system), compared to studies [9–15];

– the suitability of decisions made during heterogeneous data processing is assessed by considering external factors, reducing the time needed to find a solution (*Step 5*), compared to studies [14, 16, 17];

– the universality of the food source search strategies among the agents in the hybrid algorithm swarm allows for classification of the conditions and factors influencing the heterogeneous data processing process (*Step 6*), compared to studies [14, 16, 17]. This makes it possible to determine the most suitable solutions according to the specified optimization criterion;

– the ability to explore solution spaces defined by atypical functions due to the use of the cheetah agents' step-size selection procedure in the hybrid swarm (*Step 7*), compared to studies [9, 12–18];

– the replacement of ineffective individuals is carried out by updating the population of agents in the hybrid algorithm swarm (*Steps 8–10*), compared to studies [9, 12–18];

– the ability to conduct comparative evaluation of the effectiveness of heterogeneous data processing using the procedure for modulating metaheuristic operators (*Step 9.1*), compared to study [20];

– the ability to search for a solution in multiple directions simultaneously (*Steps 1–12*, **Tables 1.3** and **1.4**);

– the ability for deep learning of the knowledge base of agents in the hybrid algorithm swarm (*Step 10*), compared to studies [9–20];

– the ability to calculate the required amount of computational resources that need to be involved if it is impossible to perform calculations with the available computational resources (*Step 12*), compared to studies [9–20].

Disadvantages of the proposed method include:

– loss of informativeness during heterogeneous data processing due to the construction of a membership function;

– lower accuracy in evaluating individual parameters of the heterogeneous data processing state;

– loss of decision reliability when searching in multiple directions simultaneously;

– lower evaluation accuracy compared to other heterogeneous data processing methods.

The proposed method allows:

– determining the optimal performance indicator for heterogeneous data processing according to a defined optimization criterion;

– identifying effective measures to increase the efficiency of heterogeneous data processing;

– increasing the processing speed of heterogeneous data while ensuring the specified decision-making reliability;

– reducing the use of computational resources in decision support systems.

Limitations of the study include the need for information on the degree of uncertainty in the data circulating in organizational and technical systems, and the need to consider delays in data collection and delivery from components of these systems.

CONCLUSIONS

The algorithm for implementing a method for processing heterogeneous data in organizational and technical systems has been developed.

Through the introduction of additional and improved procedures, the following capabilities have been achieved:

– initialization of the initial population of agents in the hybrid algorithm swarm and their positioning within the search space, accounting for the uncertainty level of the input data circulating within the organizational and technical system, made possible through the application of correction coefficients. This significantly reduces the time required for the initial setup of the data processing subsystem;

– consideration of the initial velocity of each agent in the hybrid algorithm swarm, enabling prioritization of the search process across the system's components and elements;

– assessment of the suitability of the decisions made during data processing, taking into account the aggregate influence of external factors, which reduces the time required to find a solution;

– classification of the conditions and factors affecting the heterogeneous data processing process through the universality of the food-source search strategies of the hybrid swarm agents. This allows for the selection of the most suitable processing solutions according to the defined optimization criterion;

– exploration of complex solution spaces represented by non-standard functions, supported by the dynamic step-size adjustment procedure for cheetah agents within the hybrid algorithm swarm;

– replacement of unfit individuals by updating the hybrid algorithm swarm's population;

– comparative evaluation of the efficiency of heterogeneous data processing through metaheuristic operator modulation;

– capability to search for solutions in multiple directions simultaneously;

– capability for deep learning of the hybrid algorithm swarm agents' knowledge base;

– capability to estimate the required amount of computational resources necessary in cases where the existing computational capacity is insufficient.

A case study demonstrating the application of the proposed method for processing heterogeneous data in a military (forces) operational grouping confirmed an improvement in decision-making efficiency by approximately 14–20%, achieved through the integration of additional procedures and maintenance of decision reliability at a level of 0.9.

The algorithm for implementing a method for assessing the reliability of special-purpose radio communication systems using artificial intelligence theory has been developed. With the inclusion of additional and enhanced procedures, the following improvements have been achieved:

– initialization of the initial agent population and positioning within the search space, taking into account the uncertainty in the input data related to the operational environment of the radio communication systems via appropriate correction coefficients, reducing setup time;

– incorporation of each agent's initial velocity, which allows for prioritization in the respective search space;

– evaluation of decision suitability during data processing, accounting for multiple external influences, thereby accelerating solution discovery;

– classification of conditions and influencing factors on data processing based on the hybrid swarm agents' search strategy universality, improving solution suitability according to the defined optimization criterion;

– study of solution spaces described by non-standard functions via the cheetah agents' step-size adjustment mechanism;

– population update procedures that replace ineffective individuals in the swarm;

– performance comparison using metaheuristic operator modulation techniques;

– capability for multi-directional solution search;

– capability for deep training of the agents' knowledge bases;

– ability to calculate the required amount of computational resources needed when current resources are insufficient.

Another case study applying the proposed method to heterogeneous data processing in a military (forces) operational grouping demonstrated a 16–20% increase in decision-making efficiency, with a maintained decision reliability level of 0.9, attributable to the implementation of the described enhancements.

CONFLICT OF INTEREST

The authors declare that they have no conflict of interest in relation to this research, whether financial, personal, authorship or otherwise, that could affect the research and its results presented in this paper.

USE OF ARTIFICIAL INTELLIGENCE

The authors confirm that they did not use artificial intelligence technologies in creating the submitted work.

REFERENCES

1. Dudnyk, V., Sinenko, Y., Matsyk, M., Demchenko, Y., Zhyvotovskyi, R., Repilo, I. et al. (2020). Development of a method for training artificial neural networks for intelligent decision support systems.

Eastern-European Journal of Enterprise Technologies, 3 (2 (105)), 37–47. https://doi.org/10.15587/1729-4061.2020.203301

2. Sova, O., Shyshatskyi, A., Salnikova, O., Zhuk, O., Trotsko, O., Hrokholskyi, Y. (2021). Development of a method for assessment and forecasting of the radio electronic environment. EUREKA: Physics and Engineering, 4, 30–40. https://doi.org/10.21303/2461-4262.2021.001940
3. Pievtsov, H., Turinskyi, O., Zhyvotovskyi, R., Sova, O., Zvieriev, O., Lanetskii, B., Shyshatskyi, A. (2020). Development of an advanced method of finding solutions for neuro-fuzzy expert systems of analysis of the radioelectronic situation. EUREKA: Physics and Engineering, 4, 78–89. https://doi.org/10.21303/2461-4262.2020.001353
4. Zuiev, P., Zhyvotovskyi, R., Zvieriev, O., Hatsenko, S., Kuprii, V., Nakonechnyi, O. et al. (2020). Development of complex methodology of processing heterogeneous data in intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 4 (9 (106)), 14–23. https://doi.org/10.15587/1729-4061.2020.208554
5. Kuchuk, N., Mohammed, A. S., Shyshatskyi, A., Nalapko, O. (2019). The Method of Improving the Efficiency of Routes Selection in Networks of Connection with the Possibility of Self-Organization. International Journal of Advanced Trends in Computer Science and Engineering, 8 (1.2), 1–6.
6. Shyshatskyi, A., Zvieriev, O., Salnikova, O., Demchenko, Ye., Trotsko, O., Neroznak, Ye. (2020). Complex Methods of Processing Different Data in Intellectual Systems for Decision Support System. International Journal of Advanced Trends in Computer Science and Engineering, 9 (4), 5583–5590. https://doi.org/10.30534/ijatcse/2020/206942020
7. Pozna, C., Precup, R.-E., Horvath, E., Petriu, E. M. (2022). Hybrid Particle Filter–Particle Swarm Optimization Algorithm and Application to Fuzzy Controlled Servo Systems. IEEE Transactions on Fuzzy Systems, 30 (10), 4286–4297. https://doi.org/10.1109/tfuzz.2022.3146986
8. Yang, X.-S., Deb, S. (2013). Cuckoo search: recent advances and applications. Neural Computing and Applications, 24 (1), 169–174. https://doi.org/10.1007/s00521-013-1367-1
9. Mirjalili, S. (2015). The Ant Lion Optimizer. Advances in Engineering Software, 83, 80–98. https://doi.org/10.1016/j.advengsoft.2015.01.010
10. Yu, J. J. Q., Li, V. O. K. (2015). A social spider algorithm for global optimization. Applied Soft Computing, 30, 614–627. https://doi.org/10.1016/j.asoc.2015.02.014
11. Mirjalili, S., Mirjalili, S. M., Lewis, A. (2014). Grey Wolf Optimizer. Advances in Engineering Software, 69, 46–61. https://doi.org/10.1016/j.advengsoft.2013.12.007
12. Koval, V., Nechyporuk, O., Shyshatskyi, A., Nalapko, O., Shknai, O., Zhyvylo, Y. et al. (2023). Improvement of the optimization method based on the cat pack algorithm. Eastern-European Journal of Enterprise Technologies, 1 (9 (121)), 41–48. https://doi.org/10.15587/1729-4061.2023.273786
13. Gupta, E., Saxena, A. (2015). Robust generation control strategy based on Grey Wolf Optimizer. Journal of Electrical Systems, 11, 174–188.
14. Chaman-Motlagh, A. (2015). Superdefect Photonic Crystal Filter Optimization Using Grey Wolf Optimizer. IEEE Photonics Technology Letters, 27 (22), 2355–2358. https://doi.org/10.1109/lpt.2015.2464332

15. Nuaekaew, K., Artrit, P., Pholdee, N., Bureerat, S. (2017). Optimal reactive power dispatch problem using a two-archive multi-objective grey wolf optimizer. Expert Systems with Applications, 87, 79–89. https://doi.org/10.1016/j.eswa.2017.06.009
16. Koval, M., Sova, O., Orlov, O., Shyshatskyi, A., Artabaiev, Y., Shknai, O. et al. (2022). Improvement of complex resource management of special-purpose communication systems. Eastern-European Journal of Enterprise Technologies, 5 (9 (119)), 34–44. https://doi.org/10.15587/1729-4061.2022.266009
17. Ali, M., El-Hameed, M. A., Farahat, M. A. (2017). Effective parameters' identification for polymer electrolyte membrane fuel cell models using grey wolf optimizer. Renewable Energy, 111, 455–462. https://doi.org/10.1016/j.renene.2017.04.036
18. Zhang, S., Zhou, Y. (2017). Template matching using grey wolf optimizer with lateral inhibition. Optik, 130, 1229–1243. https://doi.org/10.1016/j.ijleo.2016.11.173
19. Khouni, S. E., Menacer, T. (2023). Nizar optimization algorithm: a novel metaheuristic algorithm for global optimization and engineering applications. The Journal of Supercomputing, 80 (3), 3229–3281. https://doi.org/10.1007/s11227-023-05579-4
20. Saremi, S., Mirjalili, S., Lewis, A. (2017). Grasshopper Optimisation Algorithm: Theory and application. Advances in Engineering Software, 105, 30–47. https://doi.org/10.1016/j.advengsoft.2017.01.004
21. Braik, M. S. (2021). Chameleon Swarm Algorithm: A bio-inspired optimizer for solving engineering design problems. Expert Systems with Applications, 174, 114685. https://doi.org/10.1016/j.eswa.2021.114685
22. Thamer, K. A., Sova, O., Shaposhnikova, O., Yashchenok, V., Stanovska, I., Shostak, S. et al. (2024). Development of a solution search method using a combined bio-inspired algorithm. Eastern-European Journal of Enterprise Technologies, 1 (4 (127)), 6–13. https://doi.org/10.15587/1729-4061.2024.298205
23. Yapici, H., Cetinkaya, N. (2019). A new meta-heuristic optimizer: Pathfinder algorithm. Applied Soft Computing, 78, 545–568. https://doi.org/10.1016/j.asoc.2019.03.012
24. Duan, H., Qiao, P. (2014). Pigeon-inspired optimization: a new swarm intelligence optimizer for air robot path planning. International Journal of Intelligent Computing and Cybernetics, 7 (1), 24–37. https://doi.org/10.1108/ijicc-02-2014-0005
25. Shyshatskyi, A., Romanov, O., Shknai, O., Babenko, V., Koshlan, O., Pluhina, T. et al. (2023). Development of a solution search method using the improved emperor penguin algorithm. Eastern-European Journal of Enterprise Technologies, 6 (4 (126)), 6–13. https://doi.org/10.15587/1729-4061.2023.291008
26. Yang, X. S (2012). Flower pollination algorithm for global optimization. Unconventional computing and natural computation. Berlin, Heidelberg: Springer, 240–249. https://doi.org/10.1007/978-3-642-32894-7_27
27. Gomes, G. F., da Cunha, S. S., Ancelotti, A. C. (2018). A sunflower optimization (SFO) algorithm applied to damage identification on laminated composite plates. Engineering with Computers, 35 (2), 619–626. https://doi.org/10.1007/s00366-018-0620-8
28. Mehrabian, A. R., Lucas, C. (2006). A novel numerical optimization algorithm inspired from weed colonization. Ecological Informatics, 1 (4), 355–366. https://doi.org/10.1016/j.ecoinf.2006.07.003
29. Qi, X., Zhu, Y., Chen, H., Zhang, D., Niu, B. (2013). An idea based on plant root growth for numerical optimization. Intelligent Computing Theories and Technology. Berlin, Heidelberg: Springer, 571–578. https://doi.org/10.1007/978-3-642-39482-9_66

30. Bezuhlyi, V., Oliynyk, V., Romanenko, I., Zhuk, O., Kuzavkov, V., Borysov, O., Korobchenko, S., Ostapchuk, E., Davydenko, T., Shyshatskyi, A. (2021). Development of object state estimation method in intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 5 (3 (113)), 54–64. https://doi.org/10.15587/1729-4061.2021.239854
31. Mahdi, Q. A., Shyshatskyi, A., Prokopenko, Y., Ivakhnenko, T., Kupriyenko, D., Golian, V. et al. (2021). Development of estimation and forecasting method in intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 3 (9 (111)), 51–62. https://doi.org/10.15587/1729-4061.2021.232718
32. Sova, O., Radzivilov, H., Shyshatskyi, A., Shevchenko, D., Molodetskyi, B., Stryhun, V. et al. (2022). Development of the method of increasing the efficiency of information transfer in the special purpose networks. Eastern-European Journal of Enterprise Technologies, 3 (4 (117)), 6–14. https://doi.org/10.15587/1729-4061.2022.259727
33. Zhang, H., Zhu, Y., Chen, H. (2013). Root growth model: a novel approach to numerical function optimization and simulation of plant root system. Soft Computing, 18 (3), 521–537. https://doi.org/10.1007/s00500-013-1073-z
34. Labbi, Y., Attous, D. B., Gabbar, H. A., Mahdad, B., Zidan, A. (2016). A new rooted tree optimization algorithm for economic dispatch with valve-point effect. International Journal of Electrical Power & Energy Systems, 79, 298–311. https://doi.org/10.1016/j.ijepes.2016.01.028
35. Murase, H. (2000). Finite element inverse analysis using a photosynthetic algorithm. Computers and Electronics in Agriculture, 29 (1-2), 115–123. https://doi.org/10.1016/s0168-1699(00)00139-3
36. Zhao, S., Zhang, T., Ma, S., Chen, M. (2022). Dandelion Optimizer: A nature-inspired metaheuristic algorithm for engineering applications. Engineering Applications of Artificial Intelligence, 114, 105075. https://doi.org/10.1016/j.engappai.2022.105075
37. Paliwal, N., Srivastava, L., Pandit, M. (2020). Application of grey wolf optimization algorithm for load frequency control in multi-source single area power system. Evolutionary Intelligence, 15 (1), 563–584. https://doi.org/10.1007/s12065-020-00530-5
38. Dorigo, M., Blum, C. (2005). Ant colony optimization theory: A survey. Theoretical Computer Science, 344 (2-3), 243–278. https://doi.org/10.1016/j.tcs.2005.05.020
39. Poli, R., Kennedy, J., Blackwell, T. (2007). Particle swarm optimization. Swarm Intelligence, 1 (1), 33–57. https://doi.org/10.1007/s11721-007-0002-0
40. Bansal, J. C., Sharma, H., Jadon, S. S., Clerc, M. (2014). Spider Monkey Optimization algorithm for numerical optimization. Memetic Computing, 6 (1), 31–47. https://doi.org/10.1007/s12293-013-0128-0
41. Yeromina, N., Kurban, V., Mykus, S., Peredrii, O., Voloshchenko, O., Kosenko, V. et al. (2021). The Creation of the Database for Mobile Robots Navigation under the Conditions of Flexible Change of Flight Assignment. International Journal of Emerging Technology and Advanced Engineering, 11 (5), 37–44. https://doi.org/10.46338/ijetae0521_05
42. Maccarone, A. D., Brzorad, J. N., Stone H. M. (2008). Characteristics And Energetics Of Great Egret And Snowy Egret Foraging Flights. Waterbirds, 31 (4), 541–549. https://doi.org/10.1675/1524-4695-31.4.541
43. Ramaji, I. J., Memari, A. M. (2018). Interpretation of structural analytical models from the coordination view in building information models. Automation in Construction, 90, 117–133. https://doi.org/10.1016/j.autcon.2018.02.025

44. Pérez-González, C. J., Colebrook, M., Roda-García, J. L., Rosa-Remedios, C. B. (2019). Developing a data analytics platform to support decision making in emergency and security management. Expert Systems with Applications, 120, 167–184. https://doi.org/10.1016/j.eswa.2018.11.023
45. Chen, H. (2018). Evaluation of Personalized Service Level for Library Information Management Based on Fuzzy Analytic Hierarchy Process. Procedia Computer Science, 131, 952–958. https://doi.org/10.1016/j.procs.2018.04.233
46. Chan, H. K., Sun, X., Chung, S.-H. (2019). When should fuzzy analytic hierarchy process be used instead of analytic hierarchy process? Decision Support Systems, 125, 113114. https://doi.org/10.1016/j.dss.2019.113114
47. Osman, A. M. S. (2019). A novel big data analytics framework for smart cities. Future Generation Computer Systems, 91, 620–633. https://doi.org/10.1016/j.future.2018.06.046
48. Nechyporuk, O., Sova, O., Shyshatskyi, A., Kravchenko, S., Nalapko, O., Shknai, O. et al. (2023). Development of a method of complex analysis and multidimensional forecasting of the state of intelligence objects. Eastern-European Journal of Enterprise Technologies, 2 (4 (122)), 31–41. https://doi.org/10.15587/1729-4061.2023.276168
49. Merrikh-Bayat, F. (2015). The runner-root algorithm: A metaheuristic for solving unimodal and multimodal optimization problems inspired by runners and roots of plants in nature. Applied Soft Computing, 33, 292–303. https://doi.org/10.1016/j.asoc.2015.04.048
50. Poliarush, O., Krepych, S., Spivak, I. (2023). hybrid approach for data filtering and machine learning inside content management system. Advanced Information Systems, 7 (4), 70–74. https://doi.org/10.20998/2522-9052.2023.4.09
51. Balochian, S., Baloochian, H. (2019). Social mimic optimization algorithm and engineering applications. Expert Systems with Applications, 134, 178–191. https://doi.org/10.1016/j.eswa.2019.05.035
52. Lenord Melvix, J. S. M. (2014). Greedy Politics Optimization: Metaheuristic inspired by political strategies adopted during state assembly elections. 2014 IEEE International Advance Computing Conference (IACC), 1157–1162. https://doi.org/10.1109/iadcc.2014.6779490
53. Moosavian, N., Roodsari, B. K. (2014). Soccer League Competition Algorithm, a New Method for Solving Systems of Nonlinear Equations. International Journal of Intelligence Science, 4 (1), 7–16. https://doi.org/10.4236/ijis.2014.41002
54. Hayyolalam, V., Pourhaji Kazem, A. A. (2020). Black Widow Optimization Algorithm: A novel meta-heuristic approach for solving engineering optimization problems. Engineering Applications of Artificial Intelligence, 87, 103249. https://doi.org/10.1016/j.engappai.2019.103249
55. Abualigah, L., Yousri, D., Abd Elaziz, M., Ewees, A. A., Al-qaness, M. A. A., Gandomi, A. H. (2021). Aquila Optimizer: A novel meta-heuristic optimization algorithm. Computers & Industrial Engineering, 157, 107250. https://doi.org/10.1016/j.cie.2021.107250
56. Hodlevskyi, M., Burlakov, G. (2023). Information technology of quality improvement planning of process subsets of the spice model. Advanced Information Systems, 7 (4), 52–59. https://doi.org/10.20998/2522-9052.2023.4.06
57. Askari, Q., Younas, I., Saeed, M. (2020). Political Optimizer: A novel socio-inspired meta-heuristic for global optimization. Knowledge-Based Systems, 195, 105709. https://doi.org/10.1016/j.knosys.2020.105709

58. Mohamed, A. W., Hadi, A. A., Mohamed, A. K. (2019). Gaining-sharing knowledge based algorithm for solving optimization problems: a novel nature-inspired algorithm. International Journal of Machine Learning and Cybernetics, 11 (7), 1501–1529. https://doi.org/10.1007/s13042-019-01053-x
59. Gödri, I., Kardos, C., Pfeiffer, A., Váncza, J. (2019). Data analytics-based decision support workflow for high-mix low-volume production systems. CIRP Annals, 68 (1), 471–474. https://doi.org/10.1016/j.cirp.2019.04.001
60. Harding, J. L. (2013). Data quality in the integration and analysis of data from multiple sources: some research challenges. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XL-2/W1, 59–63. https://doi.org/10.5194/isprsarchives-xl-2-w1-59-2013
61. Orouskhani, M., Orouskhani, Y., Mansouri, M., Teshnehlab, M. (2013). A Novel Cat Swarm Optimization Algorithm for Unconstrained Optimization Problems. International Journal of Information Technology and Computer Science, 5 (11), 32–41. https://doi.org/10.5815/ijitcs.2013.11.04
62. Karaboga, D., Basturk, B. (2007). A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. Journal of Global Optimization, 39 (3), 459–471. https://doi.org/10.1007/s10898-007-9149-x
63. Fister, I., Fister, I., Yang, X.-S., Brest, J. (2013). A comprehensive review of firefly algorithms. Swarm and Evolutionary Computation, 13, 34–46. https://doi.org/10.1016/j.swevo.2013.06.001
64. Sova, O., Radzivilov, H., Shyshatskyi, A., Shvets, P., Tkachenko, V., Nevhad, S. et al. (2022). Development of a method to improve the reliability of assessing the condition of the monitoring object in special-purpose information systems. Eastern-European Journal of Enterprise Technologies, 2 (3 (116)), 6–14. https://doi.org/10.15587/1729-4061.2022.254122
65. Khudov, H., Khizhnyak, I., Glukhov, S., Shamrai, N., Pavlii, V. (2024). The method for objects detection on satellite imagery based on the firefly algorithm. Advanced Information Systems, 8 (1), 5–11. https://doi.org/10.20998/2522-9052.2024.1.01
66. Owaid, S. R., Zhuravskyi, Y., Lytvynenko, O., Veretnov, A., Sokolovskyi, D., Plekhova, G. et al. (2024). Development of a method of increasing the efficiency of decision-making in organizational and technical systems. Eastern-European Journal of Enterprise Technologies, 1 (4 (127)), 14–22. https://doi.org/10.15587/1729-4061.2024.298568
67. Tyurin, V., Bieliakov, R., Odarushchenko, E., Yashchenok, V., Shaposhnikova, O., Lyashenko, A. et al. (2023). Development of a solution search method using an improved locust swarm algorithm. Eastern-European Journal of Enterprise Technologies, 5 (4 (125)), 25–33. https://doi.org/10.15587/1729-4061.2023.287316
68. Yakymiak, S., Vdovytskyi, Y., Artabaiev, Y., Degtyareva, L., Vakulenko, Y., Nevhad, S. et al. (2023). Development of the solution search method using the population algorithm of global search optimization. Eastern-European Journal of Enterprise Technologies, 3 (4 (123)), 39–46. https://doi.org/10.15587/1729-4061.2023.281007
69. Mohammed, B. A., Zhuk, O., Vozniak, R., Borysov, I., Petrozhalko, V., Davydov, I. et al. (2023). Improvement of the solution search method based on the cuckoo algorithm. Eastern-European Journal of Enterprise Technologies, 2 (4 (122)), 23–30. https://doi.org/10.15587/1729-4061.2023.277608

70. Raskin, L., Sukhomlyn, L., Sokolov, D., Vlasenko, V. (2023). Multi-criteria evaluation of the multifactor stochastic systems effectiveness. Advanced Information Systems, 7 (2), 63–67. https://doi.org/10.20998/2522-9052.2023.2.09
71. Arora, S., Singh, S. (2018). Butterfly optimization algorithm: a novel approach for global optimization. Soft Computing, 23 (3), 715–734. https://doi.org/10.1007/s00500-018-3102-4

DOI: 10.15587/978-617-8360-23-8.CH2

2 SET OF MATHEMATICAL MODELS FOR INFORMATION SYSTEMS RESOURCE MANAGEMENT

Yurii Zhuravskyi, Serhii Neronov, Ganna Plekhova, Olena Feoktystova, Igor Shostak, Anastasiia Voznytsia

ABSTRACT

This section of the research proposes a set of mathematical models for the functioning of information systems. The study is based on artificial intelligence theory, fuzzy set theory, and linguistic models.

The originality of the research lies in:

– the comprehensive description of the functioning process of information systems of various types, which allows improving the accuracy of modeling for subsequent managerial decision-making;

– the description of both static and dynamic processes occurring within information systems;

– the ability to model either an individual process within an information system or to conduct complex modeling of interrelated processes taking place within it;

– the dynamic description of the process of managing the state transitions of information systems during their functioning, which enables forecasting the system's evolution N steps ahead;

– the description of the process of managing computational operations during the functioning of information systems, which allows for planning rational workloads on the hardware infrastructure;

– modeling the dependency between the availability of system resources and the level of its security;

– modeling the dynamics of resource management in the course of system functioning, thereby enabling forecasting of resource utilization;

– describing possible structural states of information systems during their operation, which makes it possible to perform not only parametric but also structural management.

The proposed set of mathematical models is advisable for solving complex information system management tasks characterized by a high degree of complexity.

KEYWORDS

Information systems, destabilizing factors, levels of functioning, integrated modeling, efficiency, reliability.

Information systems are an integral component of all spheres of human activity and are applied to solve a wide range of tasks – from entertainment to highly specialized domains [1–3].

The main tasks addressed by information systems include [3–5]:

– processing heterogeneous data in the interests of a wide range of users, regardless of their application domain;

– storing heterogeneous data for user needs;

– transmitting data between individual users (or groups of users);

– supporting decision-making by authorized individuals;

– providing prerequisites for automated (intelligent) decision-making.

The development trends of modern information systems are aimed at addressing the following conceptual challenges [4–8]:

– improving the efficiency of heterogeneous data processing;

– enhancing the reliability of heterogeneous data processing;

– ensuring fault tolerance and resilience of information systems;

– increasing the accuracy of modeling information system functioning;

– maintaining a balance between efficiency and reliability in the processing of heterogeneous data, among others.

At the same time, existing scientific approaches to the synthesis and operation of information systems demonstrate insufficient accuracy and convergence. This is primarily due to the following reasons [1–9]:

– the significant influence of the human factor in the initial configuration of information systems;

– the large number of heterogeneous information sources that must be analyzed and further processed during the functioning of information systems;

– the operation of information systems under conditions of uncertainty, which causes delays in processing;

– the presence of numerous destabilizing factors affecting the functioning of information systems, among others.

These challenges stimulate the introduction of various strategies to improve the efficiency of information systems in processing heterogeneous data. One promising approach is the enhancement of existing mathematical models (or the development of new ones) for modeling the functioning of information systems.

The analysis of works [9–71] has shown that the common shortcomings of the above-mentioned studies are as follows:

– modeling of each approach is carried out only at a separate level of information system functioning;

– within a comprehensive approach, typically only two components of information system functioning are considered, which does not allow for a full assessment of the impact of managerial decisions on further functioning;

– the listed models, which are components of the aforementioned approaches, demonstrate weak integration with one another, preventing their unification into a cohesive framework;

– the models presented employ diverse mathematical apparatuses, requiring additional mathematical transformations, which in turn increase computational complexity and reduce modeling accuracy.

The aim of this research is the development of a polymodel complex for managing information system resources. This will allow modeling the functioning of information systems at different levels of their operation to support subsequent managerial decision-making. Such an approach enables the design (or improvement) of software for modern and next-generation information systems through the integration of these models.

To achieve this aim, the following objectives have been defined:

– to develop a polymodel complex for information system resource management;

– to identify the advantages and limitations of the proposed models and outline directions for their further improvement.

The object of the study is information systems. The problem addressed in the study is improving the accuracy of modeling the functioning of information systems. The subject of the study is the functioning processes of information systems using analytical-simulation and logical-dynamic models.

The hypothesis of the study is the potential to enhance both the efficiency and accuracy of information system functioning through the integration of multiple models of information system operation.

The proposed method was modeled in the Microsoft Visual Studio 2022 software environment (USA). The hardware platform used in the research process was based on an AMD Ryzen 5 processor.

2.1 DEVELOPMENT OF A POLYMODEL COMPLEX FOR INFORMATION SYSTEM RESOURCE MANAGEMENT

2.1.1 DYNAMIC MODEL OF INFORMATION SYSTEM MOTION CONTROL

Interaction operations between objects of information systems – either with one another or with service objects – can only occur when these objects enter specific interaction zones. These zones are defined by a matrix-based time-dependent function $E(t)=\left\|\varepsilon_{ij}(t)\right\|,\ i,j\in\left\{\overline{M}\cup\overline{M}\right\}$, referred to as the contact potential, where $\overline{M}$ – represent the mathematical models of the functioning information systems.

The elements of this matrix take a value of 1 if objects B_i and B_j fall within each other's interaction zones, and 0 otherwise. The geometric dimensions and shapes of these zones are determined by several factors, including:

– the type of interaction (e.g., energetic, frequency-based, informational),

– the technical characteristics of the hardware and software tools supporting the interaction,

– and the spatial positions of the objects involved.

Assume the motion state of object B_i at any time moment t is defined by two vectors: $r_i^{(d)}(t)$ and $\dot{r}_i^{(d)}(t), i\in\tilde{M}=\left\{M\cup\overline{M}\right\}$. $r_i^{(d)}(t)$ a 3D radius vector that characterizes the position of object B_i in space, $\dot{r}_i^{(d)}(t)$ – a vector characterizing the velocity of object. Introduce the motion state vector $x_i^{(d)}=\left\|r_i^{(d)T}\dot{r}_i^{(d)T}\right\|^T$.

Thus, the motion state of object B_i at any time $t\in(T_0,T_f]$ is defined by $x_i^{(d)}$. Under these conditions, the model for trajectory control of information system objects (*Model*Md) includes the following key elements.

Model of object motion process M_d

$$\dot{x}_i^{(d)}=f_i^{(d)}\left(x_i^{(d)},u_i^{(d)},t\right). \tag{2.1}$$

Constraints

$$q^{(d)}\left(x^{(d)},u^{(d)},t\right)\le 0. \tag{2.2}$$

Boundary conditions

$$h_0^{(d)}\left(x^{(d)}(T_0)\right)\le 0, h_1^{(d)}\left(x^{(d)}(T_f)\right)\le 0. \tag{2.3}$$

Quality indicators of programmed control:

$$J_1^{(d)} = \varphi^{(d)}\left(x^{(d)}(T_f)\right). \tag{2.4}$$

$$J_2^{(d)} = \int_{T_0}^{T_f} f_0^{(d)}\left(x^{(d)}(\tau), u^{(d)}(\tau), \tau\right) d\tau, \tag{2.5}$$

where $x^{(d)} = \left\| x_1^{(d)T}, x_2^{(d)T}, \ldots x_{m+\bar{m}}^{(d)T} \right\|^T$ — the state vector describing the movement of the information system and its service objects, M=1,…,m, $\bar{M} = 1, \ldots, \bar{m}$; $u^{(d)} = \left\| u_{vj}^{(d)T}(t), v^{(d)T}\left(x(t), t\right) \right\|^T$, $u^{(d)} = \left\| u_1^{(d)T}, \ldots, u_{m+\bar{m}}^{(d)T} \right\|^T$ — the components of the control input vector.

All functions in (2.1)–(2.5) are assumed to be known, given in analytical form, and continuously differentiable throughout the domain of the variables. The components of the control vector $u^{(d)}(t)$ are assumed to be Lebesgue-measurable functions defined on the interval $(T_0, T_f]$.

In this case, the contact potential of the object pair <B_i, B_j> can be calculated using the formula

$$\varepsilon_{ij}(t) = \gamma_+\left\{ R_j^{(d)} - \left| r_i^{(d)}(t) - r_j^{(d)}(t) \right| \right\}, \tag{2.6}$$

where $i, j \in \tilde{M}, \gamma_+(\tilde{\alpha}) = 1$, if $\tilde{\alpha} \geq 0, \gamma_+(\tilde{\alpha}) = 0$, if $\tilde{\alpha} < 0; R_j^{(d)}(t)$ — the specified interaction zone radius for object B_j, which, in the general case, is a closed spherical region.

From the analysis of equations (2.1)–(2.6), it follows that stationary (immobile) elements and service objects within information systems can be treated as a particular case of moving objects, for which the velocity vector $r_i(t) = r_i(t_0) = r_{i0}, \forall t \in (T_0, T_f], r_{i0}$ and the position vector define the fixed location of the object.

Equations (2.1)–(2.6) are written in general form, as their specific implementation is only possible when a particular system of forces acting on the objects during motion is defined, along with the selected reference frame, etc. These aspects are determined by the specific movement characteristics of each information system object or service object. The specific form of Model M_d will be established later during the development of a prototype software system that simulates the structural dynamics of the information system.

2.1.2 DYNAMIC MODEL OF OPERATION CONTROL IN INFORMATION SYSTEMS

The development of this and subsequent models is based on a dynamic interpretation of events occurring within an information system. The operation management model for tasks executed by information systems includes the following key components:

Model of the operation management process M_o:

$$\dot{x}_v^{(0,1)} = \sum_{j=1}^{m} u_{vj}^{(0,1)}; \dot{x}_{iæ}^{(0,2,v)} = \sum_{j=1}^{m} \sum_{\lambda=1}^{l_j} \varepsilon_{ij}(t) \Theta_{iæj\lambda}(t) u_{iæj\lambda}^{(0,2,v)}; \dot{x}_{vj}^{(0,3)} = u_{vj}^{(0,3)}; \tag{2.7}$$

$v=1,\dots,n;\ j=,\dots,m;\ i=1,\dots,m;\ æ=1,\dots,s_i.$

Constraints:

$$\sum_{j=1}^{m} u_{vj}^{(0,1)}\left[\sum_{\alpha\in\Gamma_{v1}}\left(a_{\alpha}^{(0,1)}-x_{\alpha}^{(0,1)}(t)\right)+\prod_{\beta\in\Gamma_{v2}}\left(a_{\beta}^{(0,1)}-x_{\beta}^{(0,1)}(t)\right)\right]=0. \quad (2.8)$$

$$\sum_{\lambda=1}^{l_j} u_{iæj\lambda}^{(0,2,v)}\left[\sum_{\tilde{\alpha}\in\Gamma_{v1}}\left(a_{i\tilde{\alpha}}^{(0,2,v)}-x_{i\tilde{\alpha}}^{(0,2,v)}(t)\right)+\prod_{\tilde{\beta}\in\Gamma_{iæ2}}\left(a_{i\tilde{\beta}}^{(0,2,v)}-x_{i\tilde{\beta}}^{(0,2,v)}(t)\right)\right]=0. \quad (2.9)$$

$$\sum_{v=1}^{u} u_{vj}^{(0,1)}(t)\le 1, \forall j; \sum_{j=1}^{m} u_{vj}^{(0,1)}(t)\le 1, \forall j; u_{vj}^{(0,1)}(t)\in\{0,1\}. \quad (2.10)$$

$$u_{iæj\lambda}^{(0,2,v)}(t)\in\left\{0,u_{vj}^{(0,1)}\right\}; u_{vj}^{(0,3)}(t)\in\{0,1\}; u_{vj}^{(0,3)}\left(a_{js_i}^{(0,2,v)}-x_{js_j}^{(0,2,v)}(t)\right)=0. \quad (2.11)$$

Boundary conditions

$$h_0^{(o)}\left(x^{(o)}(T_0)\right)\le 0;\ h_1^{(o)}\left(x^{(o)}(T_f)\right)\le 0. \quad (2.12)$$

Quality indicators of programmed operation management:

$$J_1^{(o)}=\sum_{v=1}^{n}\sum_{j=1}^{m}u_{vj}^{(0,3)}(T_f); J_{<2,\alpha,v>}^{(o)}=\sum_{i=1}^{m}\sum_{j=1}^{m}\left(x_{\alpha i}^{(0,3)}(T_f)-x_{vj}^{(0,3)}(T_f)\right); J_3^{(o)}=T_f-\sum_{j=1}x_{nj}^{(0,1)}(T_f). \quad (2.13)$$

$$J_{<4,i,v>}^{(o)}=\sum_{v,j,\lambda,æ}\int_{T_0}^{T_f}\varepsilon_{ij}(\tau)\Theta_{iæj\lambda}(\tau)u_{iæj\lambda}^{(0,2,v)}(\tau)d\tau, j=1,\dots,m; \lambda=1,\dots,l_j; æ=1,\dots,s_i. \quad (2.14)$$

$$J_{<5,i>}^{(o)}=\int_{T_0}^{T_f}\max_j \varepsilon_{ij}(\tau)d\tau, j\ne i. \quad (2.15)$$

$$J_{<6,i>}^{(o)}=\sum_{v,j,æ}\int_{T_0}^{T_f}\left[\varepsilon_{ij}(\tau)-\varepsilon_{ij}(\tau)u_{iæj\lambda}^{(0,2,v)}(\tau)\right]d\tau; \quad (2.16)$$

$$J_7^{(o)}=\sum_{i=1}^{m}\sum_{æ=1}^{s_i}\left(a_{iæ}^{(0,2,v)}-x_{iæ}^{(0,2,v)}(T_f)\right)^2; \quad (2.17)$$

$$J_8^{(o)}=\sum_{v=1}^{n}\sum_{i=1}^{m}\sum_{æ=1}^{s_i}\sum_{j=1}^{m}\sum_{\lambda=1}^{l_j}\int_{T_0}^{T_f}\tilde{\tilde{\alpha}}_{iæj\lambda}^{(v)}(\tau)u_{iæj\lambda}^{(0,2,v)}(\tau)d\tau; \quad (2.18)$$

$$J_9^{(o)} = \sum_{v=1}^{n}\sum_{i=1}^{m}\sum_{æ=1}^{s_i}\sum_{j=1}^{m}\sum_{\lambda=1}^{l_j}\int_{T_0}^{T_f}\tilde{\tilde{\beta}}_{iæ}^{(v)}(\tau)u_{iæj\lambda}^{(0,2,v)}(\tau)d\tau, \qquad (2.19)$$

where $x_v^{(0,1)}(t)$ – variable representing the duration of task A_v execution on object B_j ($j = 1,...,m$) at time t; $x_{iæ}^{(0,2,v)}(t)$ – variable characterizing the status of operation execution $D_æ^{(i)}$ (or $D_æ^{(i,j)}$) when solving the problem A_v; $x_v^{(0,3)}(t)$ – variable, numerically equal to the duration of the time interval from the moment of completion of the task A at object B_j until the moment $t = T_f$; $t = T_f$; $a_\alpha^{(0,1)}, a_\beta^{(0,1)}, a_{i\tilde{\alpha}}^{(0,2,v)}, a_{i\tilde{\beta}}^{(0,2,v)}, a_{is_i}^{(0,2,v)}, a_{iæ}^{(0,1,v)}$ – specified values (boundary conditions) which values must (or may) be accepted by the corresponding variables $x_\alpha^{(0,1)}, x_\beta^{(0,1)}(t)$, $x_{i\tilde{\alpha}}^{(0,2,v)}(t), x_{i\tilde{\beta}}^{(0,2,v)}(t), x_{is_i}^{(0,2,v)}(t), x_{iæ}^{(0,1,v)}(t)$ at the end of the information systems management interval at a given point in time $t = T_f$; $u_{vj}^{(0,1)}(t), u_{iæj\lambda}^{(0,2,v)}(t), u_{vj}^{(0,3)}(t)$ – controlling influences, where $u_{vj}^{(0,1)}(t) = 1$ if task A_v is solved on object B_j; $u_{vj}^{(0,1)}(t) = 0$ – otherwise $u_{iæj\lambda}^{(0,2,v)}(t) = 1$, if the operation $D_æ^{(i)}$ (or $D_æ^{(i,j)}$) is performed when solving problem A_v using the corresponding channel, $u_{iæj\lambda}^{(0,2,v)}(t) = 0$ – otherwise; $u_{vj}^{(0,3)}(t) = 1$ at the moment corresponding to the completion of task A_v on object B_j and at all subsequent moments until $t = T_f$, $u_{vj}^{(0,3)}(t) = 0$ – in opposite situations; Γ_{v1}, Γ_{v2} – a set of task numbers A_v, directly preceding and technologically related to task A_v using logical operations "AND", "OR" (or alternative OR), respectively; $\Gamma_{iæ1}, (\Gamma_{iæ2})$ – a set of interaction operation numbers performed on object B_i, immediately preceding and technologically related to operation $D_æ^{(i)}$ (or $D_æ^{(i,j)}$) using logical operations "AND", "OR", or alternative OR, respectively; $h_0^{(o)}, h_1^{(o)}$ – known differentiable functions, which are used to set the boundary conditions imposed on the vector $x^{(o)} = \left\| x_1^{(0,1)}, ..., x_n^{(0,1)}, x_{11}^{(0,2)}, ..., x_{ms_m}^{(0,2)}, x_{11}^{(0,3)}, ..., x_{nm}^{(0,3)} \right\|^{т}$ at times $t = T_0$ i $t = T_f$.

In **Table 2.1**, several examples of feasible combinations of boundary conditions for the tasks under consideration are presented.

In the following analysis, particular attention is given to the following boundary condition variants:

– variant K1: (<1,1>, <3,6>); variant K3: (<1,3>, <3,4>);

– variant K2: (<1,1>, <3,4>); variant K3: (<1,3>, <3,6>),

In each variant, the first tuple denotes the number corresponding to the selected variant for the initial time $t = T_0$, and the variant for defining the initial state $x(T_0)$, the second tuple similarly denotes $t = T_f$ i $x(T_f)$.

Table 2.1 Examples of possible combinations of boundary conditions

Time moments *t*	State vector *x*		Initial state vector *x*(t_0)			Final state vector *x*(*tf*)		
			Fixed	Unfixed		Fixed	Unfixed	
				Free	Partially Free		Free	Partially free
			1	2	3	4	5	6
Initial Time Moment t_0	Fixed	1	<1,1>	<1,2>	<1,3>	–	–	
	Unfixed	2	<2,1>	<2,2>	<2,3>	–	–	
Final Time Moment *tf*	Fixed	3	–	–	–	<3,4>	<3,5>	
	Unfixed	4	–	–	–	<4,4>	<4,5>	

Thus, constraints (2.8) and (2.9) define possible (alternative) task execution sequences A_v and their corresponding operations. In accordance with constraint (2.10), at any given time, each task A_v may be executed on only one object B_j ($v = 1,\ldots n$; $j = 1,\ldots,m$). Conversely, only one task may be executed on each object B_j only one task can be solved at any given time A_v (these restrictions correspond to the restrictions of classical assignment problems).

Table 2.2 provides examples of the constraints imposed on control inputs $u^{(0,2,v)}_{i æ j\lambda}(t)$ or various operational scenarios in servicing information systems.

Table 2.2 Possible variants of constraints

Variant number	Constraint representation	Variant number	Constraint representation
1	$\sum_{i=1}^{m} u^{(o,2)}_{i æ j\lambda} \le c^{(o,1)}_{æ j\lambda}$	8	$\sum_{æ=1}^{s_i}\sum_{\lambda=1}^{l_j}\sum_{j=1}^{m} u^{(o,2)}_{i æ j\lambda} \le c^{(o,8)}_{j}$
2	$\sum_{j=1}^{m} u^{(o,2)}_{i æ j\lambda} \le c^{(o,2)}_{i æ\lambda}$	9	$\sum_{i=1}^{m}\sum_{j=1}^{m} u^{(o,2)}_{i æ j\lambda} \le c^{(o,9)}_{æ\lambda}$
3	$\sum_{æ=1}^{s_i} u^{(o,2)}_{i æ j\lambda} \le c^{(o,3)}_{ij\lambda}$	10	$\sum_{\lambda=1}^{l_j}\sum_{i=1}^{m} u^{(o,2)}_{i æ j\lambda} \le c^{(o,10)}_{jæ}$
4	$\sum_{\lambda=1}^{l_j} u^{(o,2)}_{i æ j\lambda} \le c^{(o,4)}_{i æ j}$	11	$\sum_{i=1}^{m}\sum_{\lambda=1}^{l_j} u^{(o,2)}_{i æ j\lambda} \le c^{(o,11)}_{j\lambda}$
5	$\sum_{\lambda=1}^{l_j}\sum_{i=1}^{m}\sum_{j=1}^{m} u^{(o,2)}_{i æ j\lambda} \le c^{(o,5)}_{æ}$	12	$\sum_{\lambda=1}^{l_j}\sum_{j=1}^{m} u^{(o,2)}_{i æ j\lambda} \le c^{(o,12)}_{iæ}$
6	No constraints in this case	13	$\sum_{j=1}^{m}\sum_{æ=1}^{s_i} u^{(o,2)}_{i æ j\lambda} \le c^{(o,13)}_{i\lambda}$
7	No constraints in this case	14	$\sum_{æ=1}^{s_i}\sum_{\lambda=1}^{l_j} u^{(o,2)}_{i æ j\lambda} \le c^{(o,14)}_{ij}$

In **Tables 2.1** and **2.2**, as well as in the following formulas, let's assume for simplicity that the index of task number A_v, executed within the information systems, is fixed, and assigned to a specific object B_j. Therefore, this index will be omitted in subsequent notation. The constraints defined by equation (2.11) specify the conditions under which sets of operations can be executed, as well as the triggering of auxiliary control input $u^{(0,3)}_{vj}(t)$; $J^{(o)}_1 \div J^{(o)}_9$. The indicators J_k represents the quality metrics for managing the operations performed by the information system. In particular $J^{(o)}_1$ characterizes the total number of tasks successfully completed in the information system by time $t = T_f$; $J^{(o)}_{<2,\alpha,v>}$ — reflects the duration of the time interval during which task A_v was executed.

$J_3^{(o)}$ – denotes the total time interval required for completing all necessary tasks A_v, $v = 1,\ldots,n$; $J^{(o)}_{<4,i,v>}$ – corresponds to the duration of the time interval over which the service operation complex was performed on object B_i while solving task A_v; $J^{(o)}_{<5,i>}$ – equals the total time duration during which object B_i remained within the interaction zone (IZ) of object B_j.

Indicator (2.16) numerically corresponds to the duration of the time interval during which an object awaits service.

Indicator (2.17) is introduced in cases where it is necessary to evaluate the accuracy of boundary condition fulfillment or to minimize losses caused by the failure to execute interaction operations.

By using functions (2.18) and (2.19), it becomes possible to indirectly assess the quality of operation execution (OE) and the accuracy in meeting the directive timeframes for completing those operations.

Where:

$\tilde{\tilde{\alpha}}^{(v)}_{iæj\lambda}(\tau)$ – predefined smooth time-dependent weighting functions used to evaluate the quality of operations;

$\tilde{\tilde{\beta}}^{(v)}_{iæ}(\tau)$ – monotonically increasing (or decreasing) time functions, selected based on the directive start/end deadlines for operation execution.

2.1.3 DYNAMIC MODEL OF CHANNEL MANAGEMENT IN INFORMATION SYSTEMS

The state of a communication channel $C^{(i)}_{\lambda}$ on object B_i will be characterized by the readiness level of the channel to perform a given operation $D^{(i,j)}_{æ}$. To simplify the notation in the following formulas – as previously done – it is assumed that the index of task A_v, executed within the information system, is fixed, and assigned to object B_j. Therefore, this index will be omitted in subsequent expressions. In this case, the dynamic model describing the processes of channel reconfiguration takes the following form.

Channel management process model:

$$\dot{x}^{(k,1)}_{iæj\lambda} = \sum_{j'=1}^{m}\sum_{æ'=1}^{s_i}\Theta_{i'æ'jæ}u^{(k,1)}_{i'æ'j\lambda}\frac{b^{(j,\lambda)}_{i'æ'iæ} - x^{(k,1)}_{iæj\lambda}}{x^{(k,1)}_{i'æ'j\lambda}}, \tag{2.20}$$

$$\dot{x}^{(k,2)}_{j\lambda} = \sum_{i=1}^{m}\sum_{æ=1}^{s_i}(u^{(0,2)}_{iæj\lambda} + u^{(k,1)}_{iæj\lambda}), \tag{2.21}$$

Constraints:

$$u^{(0,2)}_{iæj\lambda}x^{(k,1)}_{iæj\lambda} = 0; \quad x^{(k,1)}_{iæj\lambda}(t) \in \{0;1\}, \tag{2.22}$$

$$\sum_{i=1}^{n}\sum_{æ=1}^{s_i}u^{(k,1)}_{iæj\lambda}(t) \le 1, \quad \forall j, \forall \lambda. \tag{2.23}$$

Boundary conditions:

$$h_0^{(k)}\left(x^{(k)}(T_0)\right)\leq 0;$$

$$h_1^{(k)}\left(x^{(k)}(T_f)\right)\leq 0. \tag{2.24}$$

Quality indicators of programmed channel management:

$$J_1^{(k)}=\sum_{\Delta_1=1}^{m-1}\sum_{\Delta_2=\Delta_1+1}^{m}\sum_{\lambda=1}^{l}\sum_{\zeta=1}^{l}\int_{T_0}^{T_f}\left(x_{\Delta_1\lambda}^{(k,2)}(\tau)-x_{\Delta_2\zeta}^{(k,2)}(\tau)\right)d\tau; \tag{2.25}$$

$$J_2^{(\hat{e})}=\sum_{\Delta_1=1}^{m-1}\sum_{\Delta_2=\Delta_1+1}^{m}\sum_{\lambda=1}^{l}\sum_{\zeta=1}^{l}\left(x_{\Delta_1\lambda}^{(\hat{e}2)}(T_f)-x_{\Delta_2\zeta}^{(\,k2)}(T_f)\right), \tag{2.26}$$

where $x_{i\text{æ}j\lambda}^{(k,1)}(t)$ – the state of channel $C_\lambda^{(i)}$ on object B_j using the reconfiguration from a readiness state for executing operation $D_{\text{æ}'}^{(i,j)}$ to a readiness state for operation $D_{\text{æ}}^{(i,j)}$; $b_{i'\text{æ}'i\text{æ}}^{(j,\lambda)}$ – a predefined value equal to the duration of the reconfiguration process between the respective channel states; $u_{i\text{æ}j\lambda}^{(k,1)}(t)$ – the control input, where $u_{i\text{æ}j\lambda}^{(k,1)}(t)=1$, if $C_\lambda^{(i)}$ if the channel is undergoing reconfiguration, and $u_{i\text{æ}j\lambda}^{(k,1)}(t)=0$ – otherwise. Constraints (2.22), (2.23) define the sequence of channel reconfiguration $C_\lambda^{(i)}$ and the conditions under which it can be initiated $C_\lambda^{(i)}$. The variable $x_{j\lambda}^{(k,2)}(t)$ represents the time interval during which the channel is actively engaged. As in the previous model $h_0^{(k)}$, $h_1^{(k)}$ – known differentiable functions that define boundary conditions for the state vector $x^{(k)}=\left\|u_{1111}^{(k,1)},\ldots,u_{msml}^{(k,1)},u_{11}^{(k,2)},\ldots,u_{ml}^{(k,2)}\right\|^{\mathrm{T}}$.

Indicators (2.25) and (2.26) are intended to evaluate the uniformity of channel utilization $t\in(T_0,T_f]$ throughout the control interval and at its completion.

2.1.4 DYNAMIC MODEL OF INFORMATION SYSTEM RESOURCE MANAGEMENT

Resource management process model:

$$\dot{x}_{j\lambda\pi}^{(p,1)}=-\sum_{i=1}^{m}\sum_{\text{æ}=1}^{s_i}d_{i\text{æ}j\lambda}^{(\pi)}\left(u_{i\text{æ}j\lambda}^{(o,2)}+u_{i\text{æ}j\lambda}^{(k,1)}\right), \tag{2.27}$$

$$\dot{x}_{j\lambda\mu}^{(p,2)}=-\sum_{i=1}^{m}\sum_{\text{æ}=1}^{s_i}g_{i\text{æ}j\lambda}^{(\mu)}\left(u_{i\text{æ}j\lambda}^{(o,2)}+u_{i\text{æ}j\lambda}^{(k,1)}\right), \tag{2.28}$$

$$\dot{x}_{j\lambda\pi\eta}^{(p,1)}=-\sum_{i=1}^{m}\sum_{\text{æ}=1}^{s_i}d_{i\text{æ}j\lambda}^{(\pi)}\left(u_{i\text{æ}j\lambda}^{(o,2)}+u_{i\text{æ}j\lambda}^{(k,1)}\right)+u_{j\lambda\pi(\eta-1)}^{(p,1)}, \tag{2.29}$$

$$\dot{x}^{(p,2)}_{j\lambda\pi\eta'} = -\sum_{i=1}^{m}\sum_{æ=1}^{s_i} g^{(\mu)}_{iæj\lambda}\left(u^{(o,2)}_{iæj\lambda} + u^{(k,1)}_{iæj\lambda}\right) + u^{(p,2)}_{j\lambda\mu(\eta'-1)}, \tag{2.30}$$

$$\dot{x}^{(p,3)}_{j\lambda\pi\eta} = u^{(p,1)}_{j\lambda\pi\eta};\ \dot{x}^{(p,4)}_{j\lambda\mu\eta'} = u^{(p,2)}_{j\lambda\mu\eta}. \tag{2.31}$$

Constraints:

$$\sum_{i,æ,\lambda} d^{(\pi)}_{iæj\lambda}\left(u^{(o,2)}_{iæj\lambda} + u^{(k,1)}_{iæj\lambda}\right) \le \tilde{H}^{(\pi)}_{j}(t), \tag{2.32}$$

$$\sum_{i,æ,\lambda}\int_{T_0}^{T_f} g^{(\mu)}_{iæj\lambda}\left(u^{(o,2)}_{iæj\lambda}(\tau) + u^{(k,1)}_{iæj\lambda}(\tau)\right)d\tau \le \int_{T_0}^{T_f}\tilde{\tilde{H}}^{(\mu)}_{j}(\tau)d\tau, \tag{2.33}$$

$$u^{(p,1)}_{j\lambda\pi\eta}\left(a^{(p,3)}_{j\lambda\pi(\eta-1)} - x^{(p,3)}_{j\lambda\pi(\eta-1)}\right) = 0,\ u^{(p,1)}_{j\lambda\pi\eta}\, x^{(p,1)}_{j\lambda\pi\eta} = 0, \tag{2.34}$$

$$u^{(p,2)}_{j\lambda\mu\eta}\left(d^{(p,4)}_{j\lambda\pi(\eta'-1)} - x^{(p,4)}_{j\lambda\pi(\eta'-1)}\right) = 0,\ u^{(p,2)}_{j\lambda\mu\eta'}\, x^{(p,2)}_{j\lambda\mu\eta'} = 0, \tag{2.35}$$

$$u^{(p,1)}_{j\lambda\pi\eta}(t)u^{(p,2)}_{j\lambda\mu\eta}(t) \in \{0,1\},\ \eta = 1,\ldots,\tilde{\rho}_\lambda;\ \eta' = 1,\ldots,\tilde{\tilde{\rho}}_\lambda. \tag{2.36}$$

Boundary conditions

$$h^{(p)}_{0}\left(x^{(p)}(T_0)\right) \le 0;\ h^{(p)}_{1}\left(x^{(p)}(T_f)\right) \le 0. \tag{2.37}$$

Quality indicators of programmed resource management:

$$J^{(p)}_{1j\pi} = \sum_{\lambda=1}^{l_j}\sum_{\eta=1}^{\tilde{\rho}_\lambda} x^{(p,3)}_{j\lambda\pi\eta}; \tag{2.38}$$

$$J^{(p)}_{2j\mu} = \sum_{\lambda=1}^{l_j}\sum_{\eta'=1}^{\tilde{\tilde{\rho}}_\lambda} x^{(p,4)}_{j\lambda\mu\eta'}, \tag{2.39}$$

where $x^{(p,1)}_{j\lambda\pi}(t), x^{(p,2)}_{j\lambda\mu}(t), x^{(p,1)}_{j\lambda\pi\eta}(t), x^{(p,2)}_{j\lambda\mu\eta'}(t)$ – the corresponding variables characterize the current volume of non-renewable resources $\Phi S^{(j)}_{\pi}$, renewable resources $\Phi N^{(j)}_{\mu}$, non-renewable replenishable, and renewable replenishable resources (at stages η and η›), used during the operation of the channel $C^{(j)}_{\lambda}$; $d^{(\pi)}_{iæj\lambda}$, $g^{(\mu)}_{iæj\lambda}$ – the prescribed consumption rates of non-renewable $\Phi S^{(j)}_{\pi}$ and renewable resources $\Phi N^{(j)}_{\mu}$ during the execution of operational activities (OA) $D^{(i,j)}_{æ}$ and the reconfiguration of the channel at unit intensity $C^{(i)}_{\lambda}$; $\tilde{H}^{(\pi)}_{j}(t)$, $\tilde{\tilde{H}}^{(\mu)}_{j}(t)$ – the replenishment (inflow) intensities of resources $\Phi S^{(j)}_{\pi}$ and $\Phi N^{(j)}_{\mu}$ accordingly. If the specified types of resources are non-replenishable, then the right-hand sides of expressions (2.32) and (2.33) will include fixed values $\tilde{H}^{(\pi)}_{j}$, $\tilde{\tilde{H}}^{(\mu)}_{j}$, which are interpreted as the maximum possible consumption intensities of the corresponding resources at each point in time.

If it is possible to organize a replenishment (regeneration) process for resources at object B_j then equations of the form (2.29)–(2.31) are introduced, where $u^{(p,1)}_{j\lambda\pi\eta}, u^{(p,2)}_{j\lambda\mu\eta}$ — represent control actions that regulate the course of replenishment (regeneration) of non-renewable and renewable resources; $a^{(p,3)}_{j\lambda\pi(\eta-1)}, a^{(p,4)}_{j\lambda\mu(\eta'-1)}$ — the specified volume of the replenishment (regeneration) $\Phi S^{(j)}_{\pi}$ operation for the non-renewable resource ($\Phi N^{(j)}_{\mu}$ — for the renewable resource) in the (η-1)-th cycle (on (η›-1)-th cycle) replenishment (regeneration) cycle; $\tilde{\rho}_{\lambda}, \tilde{\tilde{\rho}}_{\lambda}$ — represents the total allowable number of regeneration cycles for the corresponding resources.

Constraints (2.34)–(2.36) and auxiliary variables $x^{(p,3)}_{j\lambda\pi\eta}(t), x^{(p,4)}_{j\lambda\mu\eta'}(t)$ are introduced to define the class of control actions, as well as the sequence of resource regeneration (replenishment) cycles, and to determine the moments in time when these cycles are completed.

The vector functions $h^{(p)}_0, h^{(p)}_1$ assumed to be given and differentiable. Indicators of the form (2.38) and (2.39) characterize the time intervals during which the regeneration of non-renewable $\Phi S^{(j)}_{\pi}$ and renewable resources $\Phi N^{(j)}_{\mu}$, respectively, was carried out at object B_j. Additional indicators may be proposed to evaluate the uniformity (or irregularity) of the consumption (replenishment) of the respective resources.

2.1.5 DYNAMIC MODEL OF FLOW MANAGEMENT IN INFORMATION SYSTEMS

The model of the flow management process in information systems is defined as follows:

$$\dot{x}^{(p,1)}_{i æ j\lambda\rho} = u^{(p,1)}_{i æ j\lambda\rho};\ \dot{x}^{(p,2)}_{i æ j\lambda\rho} = u^{(p,2)}_{i æ j\lambda\rho}; \tag{2.40}$$

$$0 \le u^{(p,1)}_{i æ j\lambda\rho} \le c^{(p,1)}_{i æ j\lambda\rho} u^{(o,2)}_{i æ j\lambda}; \tag{2.41}$$

$$u^{(p,2)}_{i æ j\lambda\rho}\left(a^{(p,1)}_{i æ\rho} - x^{(p,1)}_{i æ j\lambda\rho}\right) = 0;\ u^{(p,2)}_{i æ j\lambda\rho} x^{(p,2)}_{i\tilde{æ}} = 0;\ u^{(p,2)}_{i æ j\lambda\rho}(t) \in \{0,1\}; \tag{2.42}$$

$$\sum_{i=1}^{m}\sum_{\lambda=1}^{l_i}\sum_{æ=1}^{s_i}\sum_{\rho=1}^{k_i} x^{(p,1)}_{i æ j\lambda\rho}\left(u^{(o,2)}_{i æ j\lambda\rho} + {}^{(o,2)}_{i æ j\lambda\rho}\right) \le \tilde{P}^{(1)}_j; \tag{2.43}$$

$$\sum_{i=1}^{m}\sum_{\lambda=1}^{l_i}\sum_{æ=1}^{s_i} u^{(p,1)}_{i æ j\lambda\rho} \le \tilde{P}^{(2)}_{j\rho}; \tag{2.44}$$

$$\sum_{\lambda=1}^{l_i}\sum_{æ=1}^{s_i}\sum_{\rho=1}^{k_i} u^{(p,1)}_{i æ j\lambda\rho} \le \tilde{P}^{(3)}_{ij}. \tag{2.45}$$

Boundary conditions:

$$h^{(p)}_0\left(x^{(p)}(T_0)\right) \le 0;$$

$$h^{(p)}_1\left(x^{(p)}(T_f)\right) \le 0. \tag{2.46}$$

Performance indicators of software-based flow management in information systems:

$$J_1^{(p)} = \sum_{i=1}^{m}\sum_{æ=1}^{s_i}\sum_{\substack{j=1\\ i\neq j}}^{m}\sum_{\lambda=1}^{l_i}\sum_{\rho=1}^{k_i}\left(a_{iæ\rho}^{(p,1)} - x_{iæj\lambda\rho}^{(p,1)}\right)x_{iæj\lambda}^{(p,1)}\Bigg|_{t=T_f}, \tag{2.47}$$

$$J_2^{(p)} = \sum_{i=1}^{m}\sum_{æ=1}^{s_i}\sum_{\substack{j=1\\ i\neq j}}^{m}\sum_{\lambda=1}^{l_i}\sum_{\rho=1}^{k_i}\int_{T_0}^{T_f} x_{iæj\lambda}^{(p,2)}(\tau)d\tau, \tag{2.48}$$

where $x_{iæj\lambda\rho}^{(p,1)}(t)$ — a variable that characterizes the current volume of information of type "ρ" received by object B_j from object B_i during the execution of the operational activity (OA) $D_{æ}^{(i,j)}$ (or the volume of information processed at object B_j, $i = j$); $x_{iæj\lambda\rho}^{(p,2)}(t)$ — an auxiliary variable that characterizes the total duration (time) of the presence of information of type ρ at object B, received (or processed) during the interaction between objects B_i and B_j in the course of executing the operational activity (OA) $D^{(i,j)}$ via channels $C_{\lambda}^{(i)}, C_{\lambda}^{(j)}$; $c_{iæj\lambda\rho}^{(p,1)}$ — a given constant that defines the maximum allowable value of $u_{iæj\lambda\rho}^{(p,1)}$; $u_{iæj\lambda\rho}^{(p,1)}$ — the intensity of information transmission from object B_i to object B_j (or the intensity of information processing at object B_j under the condition $i = j$); $u_{iæj\lambda\rho}^{(p,2)}(t)$ — auxiliary control action that takes the value $u_{iæj\lambda\rho}^{(p,2)}(t)=1$, if the reception (or processing) of information at object Bj, $u_{iæj\lambda\rho}^{(p,2)}(t)=0$ — otherwise, or in the case when, after the completion of operation $D_{æ}^{(i,j)}$(or $D_{æ}^{(i)}$, if $i = j$) the execution of operation $D_{\tilde{æ}}^{(i,j)}$, (or $D_{\tilde{æ}}^{(i)}$ if $i = j$), begins, which directly follows in the technological control cycle of object B_i after operation $D_{æ}^{(i,j)}$(or $D_{æ}^{(i)}$); $\tilde{P}_j^{(1)}, \tilde{P}_{j\rho}^{(2)}, \tilde{P}_{ij}^{(3)}$ — given values that respectively characterize: the maximum possible volume of information that can be stored at object B_j, the throughput capacity of object B_j with respect to the information flow of type p; and the throughput capacity of the channels connecting objects B_i and B_j; $a_{iæ\rho}^{(p,1)}$ — the specified volume of information of type p that can be transmitted from object B_i (or processed at object B_i) during the execution of the corresponding operation.

The functions $h_0^{(p)}, h_1^{(p)}$ are assumed to be known and differentiable. Objective functions of the form (2.47) are introduced in cases where it is necessary to evaluate the total losses caused by the absence (or loss) of specific types of information during the operation of information systems. The auxiliary variable $x_{iæj\lambda}^{(p,1)}$ takes non-zero values in cases when information exchange occurs between B_i and B_j (or information processing takes place at object B_i, if $i = j$).

The indicator of the form (2.48) is used to assess the total time losses caused by delays in the transmission, processing, and storage of information during the operation of the information system (i.e., the overall loss in the efficiency of transmitting, processing, and storing information circulating within the information network).

2.1.6 DYNAMIC MODEL OF PARAMETER CONTROL OF OPERATIONS CONDUCTED IN THE INFORMATION SYSTEM

When constructing an operations control model (model M_0), the specific characteristics of how these operations are carried out (executed) are considered. However, the execution process of both target and technological operations in information systems is accompanied by changes in a range of parameters (physical, technical, technological, etc.) that characterize each of these operations. Therefore, the operations control models (model M_0) must be supplemented each time with models for controlling the parameters of operations (model M_e). As an example, consider a model for controlling the parameters of operations related to performing measurements and evaluating the components of the state vector of the motion of object B_i using a channel $C_\lambda^{(j)}$ located on object B_j. In this case, one of the most critical parameters characterizing the measurement operations is the accuracy of determining the state vector of the motion of object B_i.

Let the linearized models of the motion of object B_i as well as the model of the measurement instruments (observation channels for tracking the trajectory of object B_i) be given in the following form:

$$\dot{x}_i^{(d)} = F(t)x_i^{(d)}, \tag{2.49}$$

$$y_{j\lambda}^{(i)}(t) = d_{j\lambda}^T(t)x_i^{(d)} + \xi_{j\lambda}, \tag{2.50}$$

where $x_i^{(d)} = \left\| r_i^{(d)T} \dot{r}_i^{(d)T} \right\|^T$ – the state vector of the motion of object B_i; $F_i(t)$ – given matrix; $\xi_{j\lambda}$ – uncorrelated measurement errors of channel $C_\lambda^{(j)}$, which follow a normal distribution with zero mean and variance equal to $\sigma_{j\lambda}^2$.

$d_j(t)$ – a given vector that relates the vector of estimated parameters $x_i^{(d)}$ to the measurable parameters $y_{j\lambda}^{(i)}(t)$. In this case, the model for controlling the parameters of operations takes the following form

$$\dot{Z} = -Z_i F_i - F_i^T Z_i - \sum_{j=1}^{n} \sum_{\text{æ} \in \Gamma_{i\text{æ}}} \sum_{\lambda=1}^{l_j} u_{i\text{æ} j\lambda}^{(e,1)} \frac{d_{i\lambda} d_{j\lambda}^T}{\sigma_{j\lambda}^2}, Z_i = \tilde{\tilde{K}}_i^{-1}. \tag{2.51}$$

Constraints

$$0 \le u_{i\text{æ} j\lambda}^{(e,1)} \le c_{j\lambda}^{(e)} u_{i\text{æ} j\lambda}^{(o,2)}. \tag{2.52}$$

Boundary conditions:

– option “a”:

$$t = T_0,\ \tilde{\tilde{K}}_i(T_0) = \tilde{\tilde{K}}_{io}, \tag{2.53}$$

$$t = T_f,\ b_\gamma^{\grave{o}} \tilde{\tilde{K}}_i b_\gamma \le \sigma_{\gamma i}^2; \tag{2.54}$$

– option “b”:

$$t = T_0,\ \tilde{\tilde{K}}_i(T_0) = \tilde{\tilde{K}}_{io}, \tag{2.55}$$

$$t = T_f, \int_{T_0}^{T_f} \sum_{i,\tilde{æ},j,\lambda} u_{i\tilde{æ}j\lambda}^{(e,1)}(\tau)d\tau \le \tilde{J}_1^{(e)}, \tag{2.56}$$

Performance indicators for operational parameter management in information systems:

– option "a"

$$J_1^{(e)} = \int_{T_0}^{T_f} \sum_{i,\tilde{æ},j,\lambda} u_{i\tilde{æ}j\lambda}^{(e,1)}(\tau)d\tau; \tag{2.57}$$

– option "b"

$$J_{2\gamma i}^{(e)} = b_\gamma^T \tilde{\tilde{K}}_i b_\gamma, \tag{2.58}$$

where Z_i – the matrix inverse of the correlation matrix $\tilde{\tilde{K}}_i$ of the estimation errors for the state vector of object B_i; $u_{i\tilde{æ}j\lambda}^{(e,1)}$ – the intensity of measurements of the motion parameters of object B_i; $\Gamma_{i\tilde{æ}}$ – the set of indices of measurement operations performed on object B_i; $c_{j\lambda}^{(e)}$ – given constants characterizing the technical capabilities of the channel $C_\lambda^{(j)}$ in performing measurement operations; $b_\gamma = \|0,0,\dots,0,1,0,\dots,0\|^T$ – an auxiliary vector used to extract the required $\tilde{\tilde{K}}_i$ element from the matrix γ; $\sigma_{j\lambda}^2$ – the given accuracy of determining the γ-th component of the state vector of object B_i; $\tilde{\tilde{K}}_{i0}$ – the given matrix characterizing the estimation errors of the state vector of object B_i at time $t = T_0$; $\tilde{J}_1^{(e)}$ – a given value representing the total resource consumption of object B_j when executing the entire set of measurement operations.

Indicator (2.56) allows for a quantitative assessment of the total resource expenditure by information systems during the execution of measurement operations.

The objective function (2.57) characterizes the accuracy of determining the γ-th element of the state vector of object B_i.

2.1.7 DYNAMIC MODELS OF STRUCTURAL DYNAMICS MANAGEMENT OF INFORMATION SYSTEMS

In constructing dynamic models for managing the structural dynamics of information systems (model M_C), a dynamic interpretation of the processes involved in executing service operation complexes is employed, as before.

To formalize these processes, it is possible to utilize the previously developed dynamic models for managing operations within information networks (model M_O) and communication channels (model M_K).

2.1.7.1 MODEL OF POLYSTRUCTURAL STATE MANAGEMENT OF INFORMATION SYSTEMS

The model describing the process of managing polystructural states (model $M_{\tilde{n}}^{(1)}$):

$$\dot{x}_{\delta\eta_1}^{(c,1)} = u_{\delta\eta_1}^{(c,1)};\ \dot{\tilde{x}}_{\delta}^{(c,1)} = \sum_{\delta'=1}^{K_\Delta} \frac{\tilde{h}_{\delta'\delta}^{(c,1)} - \tilde{x}_{\delta}^{(c,1)}}{\tilde{x}_{\delta'}^{(c,1)}} \tilde{u}_{\delta'}^{(c,1)};\ \dot{\tilde{\tilde{x}}}_{\delta\eta_1}^{(c,1)} = \tilde{\tilde{u}}_{\delta\eta_1}^{(c,1)}; \tag{2.59}$$

$$\delta = 1,\dots,K_\Delta;\ \eta_1 = 1,\dots,Œ_1.$$

Constraints:

$$\sum_{\delta=1}^{K_\Delta}\left(u_{\delta\eta_1}^{(c,1)}(t) + \tilde{u}_{\delta}^{(c,2)}\right) \le 1, \forall\eta_1;\ \ u_{\delta\eta_1}^{(c,1)}(t) \in \{0,1\};\ \ \tilde{u}_{\delta}^{(c,1)}(t), \tilde{\tilde{u}}_{\delta\eta_1}^{(c,1)}(t) \in \{0,1\}; \tag{2.60}$$

$$\sum_{\eta_1=1}^{?_1} u_{\delta\eta_1}^{(c,1)} \cdot \tilde{x}_{\delta}^{(c,1)} = 0,\ \ u_{\delta\eta_1}^{(c,1)}\left(a_{\delta(\eta_1-1)}^{(c,1)} - x_{\delta(\eta_1-1)}^{(c,1)}(t)\right) = 0; \tag{2.61}$$

$$\tilde{u}_{\delta}^{(c,1)}\left[\sum_{\chi'\in\Gamma_{\delta 1}^{(2)}}\ \sum_{\omega'\in\Gamma_{\delta 2}^{(2)}} \tilde{x}_{\chi'\omega'}^{(c,2)} + \prod_{\chi''\in\Gamma_{\delta 3}^{(2)}}\ \prod_{\omega''\in\Gamma_{\delta 4}^{(2)}} \tilde{x}_{\chi''\omega''}^{(c,2)}\right] = 0; \tag{2.62}$$

$$\tilde{\tilde{u}}_{\delta\eta_1}^{(c,1)}\left(a_{\delta\eta_1}^{(c,1)} - x_{\delta\eta_1}^{(c,1)}(t)\right) = 0. \tag{2.63}$$

Boundary conditions:

$$t = T_0 : x_{\delta\eta_1}^{(c,1)}(T_0) = \tilde{\tilde{x}}_{\delta\eta_1}^{(c,1)}(T_0) = 0;\ \ \tilde{x}_{\delta\eta_1}^{(c,1)}(T_0) \in R^1; \tag{2.64}$$

$$t = T_f : x_{\delta\eta_1}^{(c,1)}(T_f) \in R^1;\ \ \tilde{x}_{\delta\eta_1}^{(c,1)}(T_f) \in R^1;\ \ \tilde{\tilde{x}}_{\delta\eta_1}^{(c,1)}(T_f) \in R^1. \tag{2.65}$$

Performance indicators for managing polystructural macrostates of information systems:

$$J_{1\delta}^{(c,1)} = \sum_{\eta_1=1}^{Œ_1} x_{\delta\eta_1}^{(c,1)}(T_f); \tag{2.66}$$

$$J_{2}^{(c,1)} = \sum_{\eta_1=1}^{Œ_1}\sum_{\delta=1}^{K_\Delta}\left(a_{\delta}^{(c,1)} - x_{\delta\eta_1}^{(c,1)}(T_f)\right)^2; \tag{2.67}$$

$$J_{3\delta}^{(c,1)} = \sum_{\delta=1}^{K_\Delta}\int_{T_0}^{T_f} \tilde{u}_{\delta}^{(c,1)}(\tau)d\tau; \tag{2.68}$$

$$J_{2\eta_1\delta(\eta_1-1)}^{(c,1)} = \left[\tilde{\tilde{x}}_{\delta\eta_1}^{(c,1)} - \left(\tilde{\tilde{a}}_{\delta(\eta_1+1)}^{(c,1)} + \tilde{\tilde{x}}_{\delta(\eta_1+1)}^{(c,1)}\right)\right]\Bigg|_{t=T_f}. \tag{2.69}$$

The following notations are used: $x^{(c,1)}_{\delta\eta_1}(t)$ — a variable characterizing the degree of completion of macrooperation $D^{(c,1)}_{\delta\eta_1}$, which describes the functioning of the information system in the polystructural state S_δ during the η_1-th control cycle of the given system; $\tilde{x}^{(c,1)}_{\delta}(t)$ — a variable characterizing the degree of completion of the macrooperation $\tilde{D}^{(c,1)}_{\delta}$, which is associated with the transition of the information system from the current polystructural state $S_{\delta'}$ to the desired microstate S_δ (in the special case $\delta' = \delta$); $\tilde{\tilde{x}}^{(c,1)}_{\delta\eta_1}(t)$ — an auxiliary variable which value numerically corresponds to the duration of the time interval that has passed since the completion of microoperation $D^{(c,1)}_{\delta\eta_1}$; $\tilde{h}^{(c,1)}_{\delta'\delta}(t)$ — a given value numerically equal to the duration of the transition of the information system from polystructural state $S_{\delta'}$ to state S_δ; $u^{(c,1)}_{\delta\eta_1}(t)$ — a control input that takes the value 1 if macrooperation $D^{(c,1)}_{\delta\eta_1}$ must be executed, and 0 otherwise; $\tilde{\tilde{u}}^{(c,1)}_{\delta\eta_1}$ — an auxiliary control input that takes the value 1 at the time corresponding to the completion of microoperation $D^{(c,1)}_{\delta\eta_1}$, and 0 otherwise; $\tilde{u}^{(c,1)}_{\delta}(t)$ — a control input that takes the value 1 if the information system must transition from the current polystructural microstate $S_{\delta'}$ to the required state S_δ, and 0 otherwise.

Constraints of type (2.60)–(2.63) define the order and sequence of activation (or deactivation) of the above-mentioned control inputs. In expression (2.62) $\Gamma^{(2)}_{\delta 1},\Gamma^{(2)}_{\delta 3},\Gamma^{(2)}_{\delta 2},\Gamma^{(2)}_{\delta 4}$, it corresponds to the set of indices of structure types and structural states in which those structures may reside.

Indicator (2.66) makes it possible to evaluate the total duration of the information system's presence in microstate S_δ.

The Mayer-type functional (2.67) enables the assessment of total losses resulting from the failure to meet the directive-specified durations for which the information network must remain in the required macrostates. In expression (2.67) $a^{(c,1)}_{\delta}$, — denotes the directive-specified duration of the information network's presence in the polystructural state S_δ.

Indicator (2.68) provides a quantitative estimate of the total time during which the information network operates in a transitional mode.

Functional (2.69) allows for the evaluation of the time interval between two successive entries of the information network into the polystructural state S_δ (during control cycles η1 and (η1+1)).

2.1.7.2 MODEL OF DYNAMICS MANAGEMENT OF STRUCTURES OF A GIVEN TYPE OF INFORMATION SYSTEMS

The model describing the process of managing the structural dynamics of information systems (model $M^{(2)}_C$):

$$\dot{x}^{(c,2)}_{\chi\omega\eta_2} = u^{(c,2)}_{\delta\omega\eta_2};\ \dot{\tilde{x}}^{(c,2)}_{\chi\omega} = \sum_{\omega'=1}^{K_\Omega} \frac{\tilde{h}^{(c,2)}_{\omega'\omega\chi} - \tilde{x}^{(c,2)}_{\chi\omega}}{\tilde{x}^{(c,2)}_{\chi\omega'}} \tilde{u}^{(c,2)}_{\chi\omega'};\ \dot{\tilde{\tilde{x}}}^{(c,2)}_{\chi\omega\eta_2} = \tilde{\tilde{u}}^{(c,2)}_{\chi\omega\eta_2}; \tag{2.70}$$

$$\chi = 1,\dots,K_C;\ \omega = 1,\dots,K_\Omega;\ \eta_2 = 1,\dots,Œ_2.$$

Constraints:

$$\sum_{\omega=1}^{K_\Omega}\left(u^{(c,2)}_{\chi\omega\eta_2}(t) + \tilde{u}^{(c,2)}_{\chi\omega'}\right) \le 1, \forall\chi, \forall\eta_2;\ u^{(c,2)}_{\chi\omega\eta_2}(t) \in \{0,1\};\ \tilde{u}^{(c,2)}_{\chi\omega}(t), \tilde{\tilde{u}}^{(c,2)}_{\chi\omega\eta_2}(t) \in \{0,1\}; \tag{2.71}$$

$$\sum_{\eta_2=1}^{Œ_2} u_{\chi\omega\eta_2}^{(c,2)} \cdot \tilde{x}_{\chi\omega}^{(c,2)} = 0, \quad u_{\chi\omega\eta_2}^{(c,2)} \left(a_{\chi\omega(\eta_2-1)}^{(c,2)} - x_{\chi\omega(\eta_2-1)}^{(c,2)}(t) \right) = 0; \tag{2.72}$$

$$\tilde{u}_{\chi\omega}^{(c,2)} \left[\sum_{i' \in \Gamma_{\chi\omega 1}^{(3)}} \sum_{w' \in \Gamma_{\chi\omega 2}^{(3)}} \sum_{f' \in \Gamma_{\chi\omega 3}^{(3)}} \tilde{x}_{i'w'f'}^{(c,3)} + \prod_{i'' \in \Gamma_{\chi\omega 4}^{(3)}} \prod_{w'' \in \Gamma_{\chi\omega 5}^{(3)}} \prod_{f'' \in \Gamma_{\chi\omega 6}^{(3)}} \tilde{x}_{i''w''f''}^{(c,3)} \right] = 0; \tag{2.73}$$

$$\tilde{\tilde{u}}_{\chi\omega\eta_2}^{(c,2)} \left(a_{\chi\omega\eta_2}^{(c,2)} - x_{\chi\omega\eta_2}^{(c,2)} \right) = 0. \tag{2.74}$$

Boundary conditions:

$$t = T_0 : x_{\chi\omega\eta_2}^{(c,2)}(T_0) = \tilde{\tilde{x}}_{\chi\omega\eta_2}^{(c,2)}(T_0) = 0; \quad \tilde{x}_{\chi\omega}^{(c,2)}(T_0) \in R^1; \tag{2.75}$$

$$t = T_f : x_{\chi\omega\eta_2}^{(c,2)}(T_f) \in R^1; \ \tilde{x}_{\chi\omega\eta_2}^{(c,2)}(T_f) \in R^1; \ \tilde{\tilde{x}}_{\chi\omega\eta_2}^{(c,2)}(T_f) \in R^1; \tag{2.76}$$

Performance indicators for managing the structural dynamics of the specified type:

$$J_{1\chi\omega}^{(c,2)} = \sum_{\eta_2=1}^{Œ_2} x_{\chi\omega\eta_2}^{(c,2)}(T_f); \tag{2.77}$$

$$J_{2\chi\omega}^{(c,2)} = \sum_{\eta_2=1}^{Œ_2} \tilde{\tilde{u}}_{\chi\omega\eta_2}^{(c,2)}; \tag{2.78}$$

$$J_{3\chi}^{(c,2)} = \int_{T_0}^{T_f} \sum_{\omega=1}^{K_\Omega} \tilde{u}_{\chi\omega}^{(c,2)}(\tau) d\tau; \tag{2.79}$$

$$J_{4\omega\eta_2}^{(c,2)} = \sum_{\chi=1}^{K_C} \tilde{\tilde{u}}_{\chi\omega\eta_2}^{(c,2)}(t); \tag{2.80}$$

$$J_{5\omega\eta_2}^{(c,2)} = \tilde{\tilde{u}}_{\chi\omega\eta_2}^{(c,2)}(t). \tag{2.81}$$

The following notations are used: $x_{\chi\omega\eta_2}^{(c,2)}(t)$ – a variable characterizing the degree of completion of macrooperation $D_{\chi\omega\eta_2}^{(c,2)}$, which describes the process of structure G_χ being in structural state $S_{\chi\omega}$ during the η_2-th control cycle; $\tilde{x}_{\chi\omega}^{(c,2)}(t)$ – a variable characterizing the degree of completion of the macrooperation describing the transition of structure G_χ from the current structural state $S_{\chi\omega}$ to the required structural state $S_{\chi\omega}$; $\tilde{\tilde{x}}_{\chi\omega\eta_2}^{(c,2)}(t)$ – an auxiliary variable which value numerically equals the time interval that has elapsed since the completion of microoperation $D_{\chi\omega\eta_2}^{(c,2)}$; $\tilde{h}_{\omega'\omega\chi}^{(c,2)}$ – a given value numerically equal to the duration of the transition of structure G_χ from structural state $S_{\chi\omega}$ to structural state $S_{\chi\omega}$; $u_{\chi\omega\eta_2}^{(c,2)}(t)$ – a control input that takes the value 1 if macrooperation $D_{\chi\omega}^{(c,2)}$, 0 is to be performed, and 0 otherwise; $\tilde{\tilde{u}}_{\chi\omega\eta_2}^{(c,2)}(t)$ – an auxiliary control input that takes the value 1 at the moment corresponding to the completion of macrooperation $D_{\chi\omega\eta_2}^{(c,2)}$, and 0 otherwise; $\tilde{u}^{(c,2)}_{\chi\omega}(t)$ – a control input that takes the value 1 if the transition of the structure G_χ from the current state $S_{\chi\omega}$ to the required structural state $S_{\chi\omega}$ 0 is to be performed, and 0 otherwise.

The constraints of type (2.71)–(2.74) define the order and sequence of activation (or deactivation) of the above-mentioned control inputs.

In expression (2.73) $\Gamma^{(3)}_{\chi\omega1},\Gamma^{(3)}_{\chi\omega4};\Gamma^{(3)}_{\chi\omega2},\Gamma^{(3)}_{\chi\omega5};\Gamma^{(3)}_{\chi\omega3},\Gamma^{(3)}_{\chi\omega6}$, the set corresponds to the set of indices of objects that are part of the structure of the information system, the set of macrostate indices of these objects, and the set of indices of locations within the macrostates of information system objects.

Indicator (2.77) provides a quantitative measure of the total duration during which a structure of type G_χ remains in structural state $S_{\chi\omega}$; Functional of type (2.78) determines the number of times the structure G_χ has entered structural state $S_{\chi\omega}$. Indicator (2.79) allows for a quantitative assessment of the total time the structure G_χ remains in a transitional state. Indicator (2.80) allows for the assessment of the total number of heterogeneous structures $G_{\chi'}$, that are in structural state S_ω ($\delta = \omega$) where η_2 – during the control cycle.

Functional (2.81) evaluates the presence $J^{(c,2)}_{\delta\chi\omega\eta_2}=1$ (or absence $J^{(c,2)}_{\delta\chi\omega\eta_2}=0$) of structure G_χ *y* in structural state $S_{\chi\omega}$.

2.1.7.3 MODEL OF MACROSTATE MANAGEMENT OF OBJECTS WITHIN THE INFORMATION SYSTEM

The model describing the process of managing the structural dynamics of an information system (model $M^{(3)}_C$):

$$\dot{x}^{(c,3)}_{iwf\eta_3}=u^{(c,3)}_{iwf\eta_3};\ \dot{\tilde{x}}^{(c,3)}_{iwf}=\sum_{w'=1}^{K_W}\sum_{f'=1}^{K_F}\frac{\tilde{h}^{(c,3)}_{w'f'wf}-\tilde{x}^{(c,3)}_{iwf}}{\tilde{x}^{(c,3)}_{iw'f'}}\tilde{u}^{(c,3)}_{iw'f'};\qquad(2.82)$$

$$\dot{\tilde{\tilde{x}}}^{(c,3)}_{iwf\eta_3}=\tilde{\tilde{u}}^{(c,3)}_{iwf\eta_3};\ i=1,\dots,m;w=1,\dots,K_W;f=1,\dots,K_F;\eta_3=1,\dots,Œ_3.\qquad(2.83)$$

Constraints:

$$\sum_{w=1}^{K_W}\sum_{f=1}^{K_F}\left(u^{(c,3)}_{iwf\eta_3}(t)+\tilde{u}^{(c,3)}_{iwf}\right)\le 1,\forall i;\forall\eta_3;\qquad(2.84)$$

$$\sum_{i=1}^{m}\sum_{w=1}^{K_W}u^{(c,3)}_{iwf\eta_3}(t)\le 1,\forall f;\forall\eta_3;\qquad(2.85)$$

$$u^{(c,3)}_{iwf\eta_3}(t)\in\{0,1\};\ \tilde{u}^{(c,3)}_{iwf}(t),\tilde{\tilde{u}}^{(c,3)}_{iwf\eta_3}(t)\in\{0,1\};\qquad(2.86)$$

$$\sum_{\eta_3=1}^{Œ_3}u^{(c,3)}_{iwf\eta_3}\cdot\tilde{x}^{(c,3)}_{iwf\eta_3}=0,\ u^{(c,3)}_{iwf\eta_3}\left(a^{(c,3)}_{iwf(\eta_3-1)}-x^{(c,3)}_{iwf(\eta_3-1)}\right)=0;\qquad(2.87)$$

$$\tilde{u}^{(c,3)}_{iwf}\left[\sum_{\alpha'\in\Gamma^{(4)}_{iwf1}}\left(a^{(o,2)}_{i\alpha'}-\tilde{x}^{(o,2)}_{i\alpha'}(t)\right)+\prod_{\beta'\in\Gamma^{(4)}_{iwf2}}\left(a^{(o,2)}_{i\beta'}-\tilde{x}^{(o,2)}_{i\beta'}(t)\right)\right]=0;\qquad(2.88)$$

$$\tilde{\tilde{u}}_{iwf\eta_3}^{(c,3)}\left(\tilde{\tilde{a}}_{iwf(\eta_3-1)}^{(c,3)}-\tilde{\tilde{x}}_{iwf(\eta_3-1)}^{(c,3)}\right)=0. \tag{2.89}$$

Boundary conditions:

$$t=T_0: x_{iwf\eta_3}^{(c,3)}(T_0)=\tilde{\tilde{x}}_{iwf\eta_3}^{(c,3)}(T_0)=0;\ \tilde{x}_{iwf}^{(c,3)}(T_0)\in R^1; \tag{2.90}$$

$$t=T_f: x_{iwf\eta_3}^{(c,3)}(T_f)\in R^1;\ \tilde{x}_{iwf}^{(c,3)}(T_f)\in R^1;\ \tilde{\tilde{x}}_{iwf\eta_3}^{(c,3)}(T_f)\in R^1; \tag{2.91}$$

Performance indicators for managing the structural dynamics of the specified type:

$$J_{1wf\eta_3}^{(c,3)}=\sum_{i=1}^{m}\tilde{\tilde{u}}_{iwf\eta_3}^{(c,3)}(T_f); \tag{2.92}$$

$$J_{2i}^{(c,3)}=\int_{T_0}^{T_f}\sum_{w=1}^{K_W}\sum_{f=1}^{K_F}\tilde{u}_{ifw}^{(c,3)}(\tau)d\tau; \tag{2.93}$$

$$J_{3iwf}^{(c,3)}=\sum_{\eta_3=1}^{Œ_3}x_{iwf\eta_3}^{(c,3)}(T_f); \tag{2.94}$$

$$J_{4iwf}^{(c,3)}=\sum_{\eta_3=1}^{K_3}\left(a_{iwf}^{(c,3)}-x_{iwf}^{(c,3)}(T_f)\right)^2; \tag{2.95}$$

$$J_{5i\eta_3(\eta_3+1)}^{(c,3)}=\left[\tilde{\tilde{x}}_{iwf\eta_3}^{(c,3)}-\left(\tilde{\tilde{a}}_{iwf}^{(c,3)}+\tilde{\tilde{x}}_{iwf(\eta_3+1)}^{(c,3)}\right)\right]\Bigg|_{t=T_f}. \tag{2.96}$$

The following notation is adopted: $x_{iwf\eta_3}^{(c,3)}(t)$ – a variable characterizing the degree of completion of microoperation $D_{iwf\eta_3}^{(c,3)}$, which describes the functioning process of object B_i in microstate S_{iwf} during the η_3-rd control cycle; $\tilde{x}_{iwf}^{(c,3)}$ – a variable characterizing the degree of completion of the microoperation $\tilde{D}_{iwf}^{(c,3)}$, that describes the transition process of object B_i from the current macrostate $S_{iw'f'}$ to the required microstate S_{iwf}; $\tilde{\tilde{x}}_{iwf\eta_3}^{(c,3)}(t)$ – an auxiliary variable which value numerically equals the time interval that has elapsed since the completion of macrooperation $D_{iwf\eta_3}^{(c,3)}$; $\tilde{h}_{w'f'wf}^{(c,3)}$ – a given value numerically equal to the duration of the transition of object B_i from macrostate $S_{iw'f'}$ (w', w – the indices of the macrostates of object B_i, f', f – are the indices of the respective positions within those macrostates; $u_{iwf\eta_3}^{(c,3)}(t)$ – a control input that takes the value 1 if microoperation $D_{iwf\eta_3}^{(c,3)}$, 0 – to be executed, and 0 otherwise; $\tilde{u}_{iwf}^{(c,3)}(t)$ – a control input that takes the value 1 if a transition of object B_i from the current microstate $S_{iw'f'}$ to the required microstate S_{iwf}; $\tilde{\tilde{u}}_{iwf\eta_3}^{(c,3)}(t)$ – an auxiliary control input that takes the value 1 now corresponding to the completion of macrooperation $D_{iwf\eta_3}^{(c,3)}$, 0 – otherwise.

Constraints (2.84)–(2.89) define the order and sequence of activation (or deactivation) of the aforementioned control inputs.

In expression (2.88), $\Gamma_{iwf1}^{(4)}, \Gamma_{iwf2}^{(4)}$ – denotes the set of operation indices executed on object B_i (during interaction with object B_j), that directly precede macrooperation $D_{iwf}^{(c,3)}$ and are logically linked to it by "AND", "OR", or exclusive "OR" operators. Constraint (2.88) establishes the connection between model M_0 and model M_C. In turn, the interrelation of models $M_C^{(3)}, M_C^{(2)}, M_C^{(2)}, M_C^{(1)}$ is implemented through mixed-type constraints (2.73) and (2.62), respectively.

Quality indicator (2.92) characterizes the number of objects B_i, that were in microstate η_3 - during the S_{iwf}. The function of type (2.93) provides a quantitative assessment of the total duration during which object B_i remained in transitional macrostates. Indicator (2.94) determines the total time object B_i spends in microstate S_{iwf}; the Mayer-type functional (2.95) evaluates the total losses incurred due to failure to meet the directive-specified duration of the object's B_i presence in microstate S_{iwf}. In expression (2.95) $a_{iwf}^{(c,3)}$, – denotes the directive-specified duration of object B_i's presence in macrostate S_{iwf}. Functional (2.96) enables the evaluation of the time interval between two successive entries of object B_i into microstate S_{iwf} (during control cycles η_3 and (η_3 + 1)). It should be emphasized that the list of quality indicators for managing the structural dynamics of information systems (within the framework of models $M_C^{(1)}, M_C^{(2)}, M_C^{(3)}$) can be significantly extended – for example, by utilizing functionals like those proposed in models M_0, M_k, M_p, M_e, M_d, M_f). However, such extensions are determined by the specific applied problems for which the discussed models are employed. In models $M_C^{(2)}, M_C^{(3)}$ the patterns of change in variables are of the same nature as the corresponding variables in the model $M_C^{(1)}$.

Using the dynamic model for managing auxiliary operations (model M_v), let's incorporate into the previously discussed models M_0, M_k, and M_p the constraints related to the continuity of the processes involved in channel reconfiguration and the execution of operations within information systems. The necessity of accounting for these constraints arises from the specific nature of applying the above-mentioned dynamic models. During the numerical search for optimal control programs for managing the structural dynamics of information systems, interruptions may occur at certain time points within the interval (T_0, T_f], both during channel reconfiguration and operation execution.

In practice, modern technical means of information systems in some cases allow interruptions of ongoing operations (e.g., in multiprogramming or multiprocessing modes). In other cases, strict prohibition of operation interruption is enforced (e.g., when transmitting highly sensitive information or when an object exists in an abnormal state). Under such conditions, abstract mathematical models (e.g., Models M_0, M_k, M_p) must incorporate possible formalized variants for the optimal resolution of conflict situations related to interruptions.

There are several approaches to formalizing the constraints on the continuity of operations, all of which share a common feature: accounting for continuity constraints on operations and channel reconfiguration leads to an expansion of the dimensionality of the phase space in the corresponding mathematical models.

To address this, auxiliary variables are introduced, which must satisfy the following differential equations:

$$\dot{x}_{i æ j\lambda}^{(v,1)} = u_{i æ j\lambda}^{(o,2)};\ \ \dot{x}_{i æ j\lambda}^{(v,2)} = x_{i æ j\lambda}^{(v,1)};\ \ \dot{x}_{i æ j\lambda}^{(v,3)} = u_{i æ j\lambda}^{(o,1)}; \tag{2.97}$$

$$\dot{x}_{i æ j\lambda}^{(v,4)} = u_{i æ j\lambda}^{(k,1)};\ \ \dot{x}_{i æ j\lambda}^{(v,5)} = u_{i æ j\lambda}^{(v,2)};\ \ \dot{x}_{i æ j\lambda}^{(v,6)} = u_{i æ j\lambda}^{(v,3)} - u_{i æ j\lambda}^{(v,2)}, \tag{2.98}$$

where $x_{i æ j\lambda}^{(v,\zeta)}, \zeta = 1...6$ – auxiliary variables, and $u_{i æ j\lambda}^{(v,1)}, u_{i æ j\lambda}^{(v,2)}, u_{i æ j\lambda}^{(v,3)}$ – auxiliary control actions, which must satisfy the following constraints:

$$u_{iæj\lambda}^{(v,1)}\left(a_{iæ}^{(o,2)}-\sum_{i=1}^{m}\sum_{\lambda=1}^{l}x_{iæj\lambda}^{(v,1)}\right)=0, \tag{2.99}$$

$$u_{iæj\lambda}^{(v,2)}x_{iæj\lambda}^{(v,4)}=0,\ \ u_{iæj\lambda}^{(v,3)}x_{iæj\lambda}^{(v,1)}=0; \tag{2.100}$$

$$u_{iæj\lambda}^{(v,1)}(t)\in\left\{0,1\right\},\ \ u_{iæj\lambda}^{(v,2)}(t)\in\left\{0,1\right\},\ \ u_{iæj\lambda}^{(v,3)}(t)\in\left\{0,1\right\}. \tag{2.101}$$

Constraints (2.99) and (2.100) define a possible variant of "activating" the auxiliary control actions $u_{iæj\lambda}^{(v,1)}(t), u_{iæj\lambda}^{(v,2)}(t), u_{iæj\lambda}^{(v,3)}(t)$.

Taking the above into account, the first approach to formalizing continuity conditions reduces to the formulation of isoperimetric conditions of the following form:

$$\int_{T_0}^{T_f}\left(1-u_{iæj\lambda}^{(k,1)}\right)x_{iæj\lambda}^{(v,4)}x_{iæj\lambda}^{(k,1)}\left(a_{iæ}^{(o,2)}-x_{iæ}^{(o,2)}\right)d\tau=0, \tag{2.102}$$

$$\int_{T_0}^{T_f}\left(1-u_{iæj\lambda}^{(o,1)}\right)x_{iæj\lambda}^{(v,1)}x_{iæj\lambda}^{(v,1)}\left(a_{iæ}^{(o,2)}-x_{iæ}^{(o,2)}\right)d\tau=0, \tag{2.103}$$

where $x_{iæj\lambda}^{(v,1)}(t_0)=x_{iæj\lambda}^{(v,4)}(t_0)=0,\ x_{iæj\lambda}^{(v,1)}(T_f)\in R^1, x_{iæj\lambda}^{(v,4)}(T_f)\in R^1$ – the set of real numbers. Relations (2.102) and (2.103) define, respectively, the constraints on the continuity of executing the channel reconfiguration C_λ^j operation $D_æ^{(i,j)}$. It should be noted that the constraints on the continuity of operations related to the replenishment (regeneration) of stored and non-renewable resources are formulated in the same way as for interaction operations.

An alternative approach to formalizing the continuity constraints of operations and channel reconfiguration in information systems may also be proposed. These constraints, when expressed as additional boundary conditions, are written as follows:

$$\left.\left\{\left[x_{iæj\lambda}^{(v,3)}x_{iæj\lambda}^{(v,1)}+\frac{\left(a_{iæ}^{(o,2)}\right)^2}{2}-x_{iæj\lambda}^{(v,2)}\right]^2 x_{iæj\lambda}^{(v,1)}\right\}\right|_{t=T_f}=0, \tag{2.104}$$

$$\left.\left(x_{iæj\lambda}^{(v,6)}-x_{iæj\lambda}^{(v,4)}\right)^2\right|_{t=T_f}=0.$$

Subject to the condition, $x_{iæj\lambda}^{(v,\zeta)}(T_0)=0, x_{iæj\lambda}^{(v,\zeta)}(T_f)\in R^1$.

In expression (2.104), the value of the product $x_{iæj\lambda}^{(v,3)}x_{iæj\lambda}^{(v,1)}$ is numerically equal to the area under the integrand curve corresponding to the solution of the first equation in formula (2.97) over the time interval $(t'_{iæj\lambda}, T_f]$ where $t'_{iæj\lambda}$, t – the moment when the operation $D_æ^{(i,j)}$, performed by the channel, is completed C_λ^j.

The value $\frac{\left(a_{iæ}^{(o,2)}\right)^2}{2}$ is numerically equal to the area under the integrand curve corresponding to the solution of the first equation in formula (2.97) over the time interval $(T_0, t'_{iæj\lambda}]$, assuming that the interaction operation (OA) was executed without interruption using the resources of a single channel C_λ^j. From the analysis of expression (2.104), it follows that in the case where the OA $D_æ^{(i,j)}$ was executed by channel C_λ^j without interruptions, the difference between the values inside the square brackets equals zero. Otherwise (if OA $D_æ^{(i,}$ was interrupted), this difference is non-zero. To account for cases in which the channel C_λ^j is not scheduled to perform OA $D_æ^{(i,j)}$ within the interval $(T_0, T_f]$, an additional multiplier is introduced into expression (2.104) $x_{iæj\lambda}^{(v,1)}$, which takes the value zero at time $t = T_0$.

2.1.8 MODEL OF INFORMATION SYSTEM SECURITY MANAGEMENT UNDER CENTRALIZED CONTROL

The purpose of developing a model for managing the security of information networks is as follows:

— to model the allocation of the required number of resources for each element of the information network in response to a specific type of cyberattack, within a limited time interval;

— to model the number of engaged resources in each element of the information system, as well as to model the number of available (free) resources in the system.

The need for additional resource allocation is assumed to be deterministic and time-dependent. Such resource allocation planning accounts for constraints on resource levels (preventing shortages or overutilization), as well as the minimization of total costs, including redistribution between the elements of the information system.

To model the security management process of information systems, the following notations are introduced:

$N = i|i = 1,\ldots,n$: the set of n service requests within each element of the information system;

$P = 0p|0p = 01,\ldots, 0m$: the set of m total information system resources;

$Np = i|i = 0p, 1\ldots n$: the set representing n service requests, where node $0p$ represents a particular information system element p that supplies resources;

$R = r|r = 1,\ldots u$: the set of u types of information system resources used for protection against a specific type of cyberattack;

$V = \upsilon|\upsilon = 1,\ldots k$: the set of k homogeneous information channels with capacity Q and their respective bandwidth.

Resource reserves and consumption in the information system:

$H_p^{L,r}, h_i^{L,r}, H_p^{E,r}, h_i^{E,r}$: the cost of maintaining the readiness of information system resources in element p for the benefit of element i;

$I_{p0}^{L,r}, L_{i0}^{L,r}, I_{p0}^{E,r}, L_{i0}^{E,r}$: the initial level of information system resources of type r in element p intended for element i;

$C_p^L, c_i^L, C_p^E, c_i^E$: the maximum volume of information system resources in element p allocated for element i;

D^r_{pit} : the required amount of resources to be delivered from element *p* to satisfy the needs of element $i \in N$ during the period $t \in T$.

Costs associated with the transfer of computing resources between information system elements:

The distance between information system elements $i \in N_p$, $j \in N_p$, d^p_{ij}.

w^r_L and w^r_E: weighting coefficients of utilized and unused resources of type *r* within the information system;

e*r*: the cost of allocating one unit of information system resources;

s*r*: the cost of utilizing resources of type *r* from another element of the information system.

Information system resource maintenance costs:

– g*r*: the cost of maintaining a single unit of information system resource of type *r*.

The model for managing the security of information systems within a closed information system – comprising multiple elements that supply available resources and multiple elements that utilize them – is described as follows

$$\min \sum_{i\in N}\sum_{t\in T}\sum_{r\in R}(h_i^{Lr}L_{it}^{Lr}+h_i^{Er}L_{it}^{Er})+\sum_{p\in P}\sum_{t\in T}\sum_{r\in R}(H_p^{L,r}I_{pt}^{Lr}+H_p^{Er}I_{pt}^{Er})+$$
$$+\sum_{p\in P}\sum_{t\in T}\sum_{r\in R}e_r n_t^{p,r}+\sum_{p\in P}\sum_{t\in T}\sum_{p'\in P}\sum_{r\in R}g_r F_{pt}^{p'r}+\sum_{i\in N}\sum_{p\in P}\sum_{p'\in P}\sum_{t\in T}\sum_{r\in R}s_r W_{ip't}^{pr}+$$
$$+\sum_{p\in P}\sum_{t\in T}\sum_{i\in N_p}\sum_{j\in N_p}(a\sum_{v\in V}x_{ijvt}^{p}+\sum_{r\in R}b(w_L^r X_{ijt}^{pr}+w_E^r E_{ijt}^{pr})d_{ij}^{p}), \quad (2.105)$$

subject to the following conditions:

$$L_{pit}^{Lr}=L_{pit-1}^{Lr}+\sum_{p'\in P}Q_{pit}^{p'r}-D_{pit}^{r} \quad \forall i\in N, t\in T, p\in P, r\in R, \quad (2.106)$$

$$I_{pt}^{Lr}=I_{pit-1}^{Lr}-\sum_{i\in N}\sum_{p'\in P}Q_{pit}^{p'r}+\sum_{p'\in P}F_{pt}^{p'r}, \quad \forall t\in T, p\in P, r\in R, \quad (2.107)$$

$$L_{it}^{Er}=L_{it-1}^{Er}-\sum_{p\in P}Z_{it}^{pr}+\sum_{p\in P}D_{pit}^{r}-\sum_{p\in P}\sum_{p'\in P}W_{ip't}^{pr}, \quad \forall i\in N, t\in T, r\in R, \quad (2.108)$$

$$I_{pt}^{Er}=I_{pt-1}^{Er}+\sum_{i\in N}Z_{it}^{pr}-\sum_{p'\in P}F_{pt}^{p'r}+n_t^{pr}+\sum_{p'\in P}W_{ip't}^{pr} \quad \forall p\in P, t\in T, r\in R, \quad (2.109)$$

$$\sum_{i\in N_p, i\neq j}(X_{ijt}^{pr}-X_{jit}^{pr})=\sum_{p'\in P}Q_{pjt}^{p'r}, \quad \forall j\in N, p\in P, t\in T, r\in R, \quad (2.110)$$

$$\sum_{i\in N_p, i\neq j}(E_{jit}^{pr}-E_{ijt}^{pr})=Z_{jt}^{pr}+\sum_{p'\in P}W_{jp't}^{pr} \quad \forall j\in N, p\in P, t\in T, r\in R, \quad (2.111)$$

$$0\le\sum_{p\in P}\sum_{r\in R}L_{pit}^{Lr}\le c_i^L \quad \forall i\in N, t\in T, \quad (2.112)$$

$$0\le\sum_{r\in R}I_{pt}^{Lr}\le C_p^L, \quad \forall p\in P, t\in T, \quad (2.113)$$

$$0 \le \sum_{p\in P}\sum_{r\in R} L_{pit}^{Er} \le c_i^E, \ \ \forall i \in N, t \in T, \tag{2.114}$$

$$0 \le \sum_{r\in R} I_{pt}^{Er} \le C_p^E, \ \ \forall p \in P, t \in T, \tag{2.115}$$

$$\sum_{p\in P}\sum_{r\in R}(X_{ijt}^{pr} + E_{ijt}^{pr}) \le Q\sum_{p\in P}\sum_{v\in V} x_{ijvt}^{p}, \ \ \forall i, j \in N_p, t \in T, \tag{2.116}$$

$$\sum_{i\in N_p}\sum_{v\in V} x_{ijvt}^{p} \le 1, \ \ \forall j \in N, p \in P, t \in T, \tag{2.117}$$

$$\sum_{i\in N_p i\ne j} x_{ijvt}^{p} = \sum_{i\in N_p i\ne j} x_{jivt}^{p}, \ \ \forall vs. \in V, j \in N_p, p \in P, t \in T, \tag{2.118}$$

$$\sum_{j\in N} x_{0_p jvt}^{p} \le 1, \ \ \forall vs. \in V, p \in P, t \in T. \tag{2.119}$$

The analytical expressions presented above form the foundation for managing the security of information systems under centralized control.

2.1.8.1 MATHEMATICAL MODEL OF INFORMATION SYSTEM SECURITY MANAGEMENT IN SELF-ORGANIZATION MODE

In the self-organization mode model, there is no pooling of shared resources among the elements of the information system $W_{ip't}^{pr} = F_{pt}^{p'r} = 0$ for $p' \ne p, \forall p, p' \in P, i \in N, t \in T, r \in R$. Each element of the information system independently manages its own resources with respect to the elements acting as resource requesters. Thus, the mathematical model is solved independently for each IS element that supplies resources, and the costs to be minimized include expenses associated with maintaining an adequate level of IS resources and transportation costs for their delivery. The model is described as follows

$$\begin{aligned} \min \sum_{i\in N}\sum_{t\in T}\sum_{r\in R}(h_i^{Lr}L_{it}^{Lr} + h_i^{Er}L_{it}^{Er}) + \sum_{t\in T}\sum_{r\in R}(H_p^{L,r}I_{pt}^{Lr} + H_p^{Er}I_{pt}^{Er}) + \\ + \sum_{t\in T}\sum_{r\in R} e_r n_t^{p,r} + \sum_{p\in P}\sum_{t\in T}\sum_{p'\in P}\sum_{r\in R} g_r F_{pt}^{p'r} + \\ + \sum_{t\in T}\sum_{i\in N_p}\sum_{j\in N_p}(a\sum_{v\in V} x_{ijvt}^{p} + \sum_{r\in R} b(w_L^r X_{ijt}^{pr} + w_E^r E_{ijt}^{pr}) d_{ij}^{p}). \end{aligned} \tag{2.120}$$

The objective function aims to minimize the total cost incurred by the IS element *p* in maintaining the necessary resources in readiness for delivery to each requester. This includes the cost of keeping resources available for use, as well as the fixed and variable transportation costs required for their delivery.

These costs depend on:

$$L_{pit}^{Lr} = L_{pit-1}^{Lr} + Q_{pit}^{r} - D_{pit}^{r}, \quad \forall i \in N, t \in T, r \in R, \tag{2.121}$$

$$I_{pt}^{Lr} = I_{pt-1}^{Lr} - \sum_{i \in N} Q_{pit}^{r} + F_{pt}^{r}, \quad \forall t \in T, r \in R, \tag{2.122}$$

$$L_{it}^{Er} = L_{it-1}^{Er} - \sum_{p \in P} Z_{it}^{pr} + \sum_{p \in P} D_{pit}^{r}, \quad \forall i \in N, t \in T, r \in R, \tag{2.123}$$

$$I_{pt}^{Er} = I_{pt-1}^{Er} + \sum_{i \in N} Z_{it}^{pr} - F_{pt}^{r} + n_{t}^{pr}, \quad \forall t \in T, r \in R, \tag{2.124}$$

$$\sum_{i \in N_p, i \neq j} (X_{ijt}^{pr} - X_{jit}^{pr}) = Q_{pjt}^{r}, \quad \forall j \in N, t \in T, r \in R, \tag{2.125}$$

$$\sum_{i \in N_p, i \neq j} (E_{jit}^{pr} - E_{ijt}^{pr}) = Z_{jt}^{pr}, \quad \forall j \in N, t \in T, r \in R, \tag{2.126}$$

$$0 \le \sum_{p \in P} \sum_{r \in R} L_{pit}^{Lr} \le c_{i}^{L}, \ \forall i \in N, t \in T, \tag{2.127}$$

$$0 \le \sum_{r \in R} I_{pt}^{Lr} \le C_{p}^{L}, \quad \forall t \in T, \tag{2.128}$$

$$0 \le \sum_{p \in P} \sum_{r \in R} L_{pit}^{Er} \le c_{i}^{E}, \ \forall i \in N, t \in T, \tag{2.129}$$

$$0 \le \sum_{r \in R} I_{pt}^{Er} \le C_{p}^{E}, \ \forall t \in T, \tag{2.130}$$

$$\sum_{p \in P} \sum_{r \in R} (X_{ijt}^{pr} + E_{ijt}^{pr}) \le Q \sum_{p \in P} \sum_{v \in V} x_{ijvt}^{p}, \ \forall i, j \in N_p, t \in T, \tag{2.131}$$

$$\sum_{i \in N_p} \sum_{v \in V} x_{ijvt}^{p} \le 1, \quad \forall j \in N, t \in T, \tag{2.132}$$

$$\sum_{i \in N_p i \neq j} x_{ijvt}^{p} = \sum_{i \in N_p i \neq j} x_{jivt}^{p}, \ \forall vs. \in V, j \in N_p, t \in T, \tag{2.133}$$

$$\sum_{j \in N} x_{0_p\, jvt}^{p} \le 1, \quad \forall vs. \in V, t \in T, \tag{2.134}$$

Q_{pit}^{r}: the quantity of information system resources of type *r*, owned by supplier *p*, that were delivered to client *i* during period *t*;

F_{pt}^{r}: the quantity of available information system resources of type *r*, owned by supplier *p*, that were replenished with products at their level during period *t*.

2.1.9 GENERALIZED DETERMINISTIC LOGICAL–DYNAMIC MODEL OF STRUCTURAL DYNAMICS MANAGEMENT OF INFORMATION NETWORKS

An analysis of previously developed models for managing structural dynamics shows that, in general, the generalized deterministic dynamic model for managing the structural dynamics of an information system (Model *M*) can be represented in the following form:

$$\dot{x} = f(x, u, t), \tag{2.135}$$

$$h_0(x(T_0)) \le 0, \ \ h_1(x(T_f)) \le 0, \tag{2.136}$$

$$q^{(1)}(x, u) = 0, \ \ q^{(2)}(x, u) = 0, \tag{2.137}$$

$$J_i = J_i\left(x(t), u(t), t\right) = \varphi_i\left(x(t_f)\right) + \int_{T_0}^{T_f} f_{0i}\left(x(\tau), u(\tau), \tau\right) d\tau, i = 1, \ldots l_M, \tag{2.138}$$

where x — the generalized state vector of the multistructural configuration of the information system; u — the generalized control input vector; h_0, h_1 — known vector functions used to define the initial data for the control problem of the structural dynamics of the information system at time $t = T_0$ and the terminal (desired) values of the system state vector at the end of the control interval ($t = T_f$).

It should be noted that the boundary conditions in the previously formulated structural dynamics control problems of the information system may be either fixed at both ends of the phase trajectory $x(t)$ or variable.

The vector functions $q^{(1)}$, $q^{(2)}$ define the fundamental spatiotemporal, technical, and technological constraints imposed on the functioning process of the information system.

The conducted analysis shows that all sets of indicators used to assess the quality of structural dynamics management in information systems can be categorized into the following groups:

J_1 — indicators assessing the operational efficiency of the information system;

J_2— indicators assessing the throughput capacity of the information system;

J_3 — indicators assessing the quality of operations (tasks) performed as part of the technological control cycles;

J_4 — indicators assessing resource consumption during the functioning of the information system;

J_5 — indicators assessing the flexibility of structural configurations of the information system (structural and functional self-organization indicators);

J_6 — indicators assessing the resilience (survivability) of the information system;

J_7 — indicators assessing the interference resistance of the information system;

J_8 — indicators assessing the reliability of the information system during its target deployment;

J_9 — indicators assessing the *security* (protection level) of the information system.

CONCLUSIONS

As a result of the conducted research, a polymodel complex was developed, comprising analytica-simulation and logical-dynamic models for managing the motion, channels, resources, complexes, and parameters of target functions, as well as supporting operations, flows, and structures of information systems. In generalized form, this polymodel complex was represented as a multilevel alternative dynamic system graph with a reconfigurable structure.

The first advantage of the proposed generalized description lies in its ability to ensure, at the conceptual, model-algorithmic, informational, and software levels of detail, correct alignment (according to the criteria of homomorphisms and dynamomorphisms of relations) of the mathematical (analytical-simulation) models of structural dynamics management of information systems (both real and virtual-software) with their logical-algebraic and logical-linguistic analogues (models) constructed on the basis of intelligent information technologies. Unlike existing scenario-based behavioral models of information system functioning, which are built on finite-state automata and simulation descriptions, the proposed logical-dynamic approach makes it possible, at a constructive level, to address both the synthesis of technologies for information system functioning and the tasks of integrated planning of information processes occurring within them, thereby ensuring the effective functioning of the Industrial Internet of Things.

The second advantage of the proposed polymodel complex is its ability to uniformly (using the same mathematical structures) provide a formal description of both the tasks of integrated modeling of information system management processes and the tasks of planning their operations, plan correction (re-planning), as well as the tasks of real-time control and monitoring of their state. This ensures correct inter-model coordination with a unified language for describing the analyzed processes.

Overall, the dynamic interpretation of processes for managing the elements and subsystems of information systems proposed by the authors makes it possible to significantly reduce the dimensionality of these software control tasks (through recurrent model descriptions), to decompose and parallelize the initial planning and management tasks of information systems, and to improve the efficiency of solving such tasks when using modern multiprocessor and multicore computers. It also enhances the stability of the computational process associated with solving the tasks of planning and managing information systems.

Consideration of uncertainty factors within the developed polymodel complex is proposed using combined approaches, based on the authors' technologies of integrated modeling. These are oriented both toward analytical-simulation modeling of possible scenarios of proactive information system management programs at the execution stage — with subsequent correction and implementation of the required level of various types of redundancy (functional, technical, temporal, etc.) — and toward the construction and analysis of approximated reachability regions in the space of criterial functions and interval-defined perturbations.

Such approaches make it possible to identify the most robust proactive management programs for information systems.

CONFLICT OF INTEREST

The authors declare that they have no conflict of interest in relation to this research, whether financial, personal, authorship or otherwise, that could affect the research and its results presented in this paper.

USE OF ARTIFICIAL INTELLIGENCE

The authors confirm that they did not use artificial intelligence technologies in creating the submitted work.

REFERENCES

1. Dudnyk, V., Sinenko, Y., Matsyk, M., Demchenko, Y., Zhyvotovskyi, R., Repilo, I. et al. (2020). Development of a method for training artificial neural networks for intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 3 (2 (105)), 37–47. https://doi.org/10.15587/1729-4061.2020.203301
2. Sova, O., Shyshatskyi, A., Salnikova, O., Zhuk, O., Trotsko, O., Hrokholskyi, Y. (2021). Development of a method for assessment and forecasting of the radio electronic environment. EUREKA: Physics and Engineering, 4, 30–40. https://doi.org/10.21303/2461-4262.2021.001940
3. Pievtsov, H., Turinskyi, O., Zhyvotovskyi, R., Sova, O., Zvieriev, O., Lanetskii, B., Shyshatskyi, A. (2020). Development of an advanced method of finding solutions for neuro-fuzzy expert systems of analysis of the radioelectronic situation. EUREKA: Physics and Engineering, 4, 78–89. https://doi.org/10.21303/2461-4262.2020.001353
4. Zuiev, P., Zhyvotovskyi, R., Zvieriev, O., Hatsenko, S., Kuprii, V., Nakonechnyi, O. et al. (2020). Development of complex methodology of processing heterogeneous data in intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 4 (9 (106)), 14–23. https://doi.org/10.15587/1729-4061.2020.208554
5. Kuchuk, N., Mohammed, A. S., Shyshatskyi, A., Nalapko, O. (2019). The Method of Improving the Efficiency of Routes Selection in Networks of Connection with the Possibility of Self-Organization. International Journal of Advanced Trends in Computer Science and Engineering, 8 (1.2), 1–6. https://doi.org/10.30534/ijatcse/2019/0181.22019
6. Shyshatskyi, A., Zvieriev, O., Salnikova, O., Demchenko, Ye., Trotsko, O., Neroznak, Ye. (2020). Complex Methods of Processing Different Data in Intellectual Systems for Decision Support System. International Journal of Advanced Trends in Computer Science and Engineering, 9 (4), 5583–5590. https://doi.org/10.30534/ijatcse/2020/206942020
7. Pozna, C., Precup, R.-E., Horvath, E., Petriu, E. M. (2022). Hybrid Particle Filter–Particle Swarm Optimization Algorithm and Application to Fuzzy Controlled Servo Systems. IEEE Transactions on Fuzzy Systems, 30 (10), 4286–4297. https://doi.org/10.1109/tfuzz.2022.3146986

8. Yang, X.-S., Deb, S. (2013). Cuckoo search: recent advances and applications. Neural Computing and Applications, 24 (1), 169–174. https://doi.org/10.1007/s00521-013-1367-1
9. Mirjalili, S. (2015). The Ant Lion Optimizer. Advances in Engineering Software, 83, 80–98. https://doi.org/10.1016/j.advengsoft.2015.01.010
10. Yu, J. J. Q., Li, V. O. K. (2015). A social spider algorithm for global optimization. Applied Soft Computing, 30, 614–627. https://doi.org/10.1016/j.asoc.2015.02.014
11. Mirjalili, S., Mirjalili, S. M., Lewis, A. (2014). Grey Wolf Optimizer. Advances in Engineering Software, 69, 46–61. https://doi.org/10.1016/j.advengsoft.2013.12.007
12. Koval, V., Nechyporuk, O., Shyshatskyi, A., Nalapko, O., Shknai, O., Zhyvylo, Y. et al. (2023). Improvement of the optimization method based on the cat pack algorithm. Eastern-European Journal of Enterprise Technologies, 1 (9 (121)), 41–48. https://doi.org/10.15587/1729-4061.2023.273786
13. Gupta, E., Saxena, A. (2015). Robust generation control strategy based on Grey Wolf Optimizer. Journal of Electrical Systems, 11, 174–188.
14. Chaman-Motlagh, A. (2015). Superdefect Photonic Crystal Filter Optimization Using Grey Wolf Optimizer. IEEE Photonics Technology Letters, 27 (22), 2355–2358. https://doi.org/10.1109/lpt.2015.2464332
15. Nuaekaew, K., Artrit, P., Pholdee, N., Bureerat, S. (2017). Optimal reactive power dispatch problem using a two-archive multi-objective grey wolf optimizer. Expert Systems with Applications, 87, 79–89. https://doi.org/10.1016/j.eswa.2017.06.009
16. Koval, M., Sova, O., Orlov, O., Shyshatskyi, A., Artabaiev, Y., Shknai, O. et al. (2022). Improvement of complex resource management of special-purpose communication systems. Eastern-European Journal of Enterprise Technologies, 5 (9 (119)), 34–44. https://doi.org/10.15587/1729-4061.2022.266009
17. Ali, M., El-Hameed, M. A., Farahat, M. A. (2017). Effective parameters' identification for polymer electrolyte membrane fuel cell models using grey wolf optimizer. Renewable Energy, 111, 455–462. https://doi.org/10.1016/j.renene.2017.04.036
18. Zhang, S., Zhou, Y. (2017). Template matching using grey wolf optimizer with lateral inhibition. Optik, 130, 1229–1243. https://doi.org/10.1016/j.ijleo.2016.11.173
19. Khouni, S. E., Menacer, T. (2023). Nizar optimization algorithm: a novel metaheuristic algorithm for global optimization and engineering applications. The Journal of Supercomputing, 80 (3), 3229–3281. https://doi.org/10.1007/s11227-023-05579-4
20. Saremi, S., Mirjalili, S., Lewis, A. (2017). Grasshopper Optimisation Algorithm: Theory and application. Advances in Engineering Software, 105, 30–47. https://doi.org/10.1016/j.advengsoft.2017.01.004
21. Braik, M. S. (2021). Chameleon Swarm Algorithm: A bio-inspired optimizer for solving engineering design problems. Expert Systems with Applications, 174, 114685. https://doi.org/10.1016/j.eswa.2021.114685
22. Thamer, K. A., Sova, O., Shaposhnikova, O., Yashchenok, V., Stanovska, I., Shostak, S. et al. (2024). Development of a solution search method using a combined bio-inspired algorithm. Eastern-European Journal of Enterprise Technologies, 1 (4 (127)), 6–13. https://doi.org/10.15587/1729-4061.2024.298205
23. Yapici, H., Cetinkaya, N. (2019). A new meta-heuristic optimizer: Pathfinder algorithm. Applied Soft Computing, 78, 545–568. https://doi.org/10.1016/j.asoc.2019.03.012

24. Duan, H., Qiao, P. (2014). Pigeon-inspired optimization: a new swarm intelligence optimizer for air robot path planning. International Journal of Intelligent Computing and Cybernetics, 7 (1), 24–37. https://doi.org/10.1108/ijicc-02-2014-0005
25. Shyshatskyi, A., Romanov, O., Shknai, O., Babenko, V., Koshlan, O., Pluhina, T. et al. (2023). Development of a solution search method using the improved emperor penguin algorithm. Eastern-European Journal of Enterprise Technologies, 6 (4 (126)), 6–13. https://doi.org/10.15587/1729-4061.2023.291008
26. Yang, X. S (2012). Flower pollination algorithm for global optimization. Unconventional computing and natural computation. Berlin, Heidelberg: Springer, 240–249. https://doi.org/10.1007/978-3-642-32894-7_27
27. Gomes, G. F., da Cunha, S. S., Ancelotti, A. C. (2018). A sunflower optimization (SFO) algorithm applied to damage identification on laminated composite plates. Engineering with Computers, 35 (2), 619–626. https://doi.org/10.1007/s00366-018-0620-8
28. Mehrabian, A. R., Lucas, C. (2006). A novel numerical optimization algorithm inspired from weed colonization. Ecological Informatics, 1 (4), 355–366. https://doi.org/10.1016/j.ecoinf.2006.07.003
29. Qi, X., Zhu, Y., Chen, H., Zhang, D., Niu, B. (2013). An idea based on plant root growth for numerical optimization. Intelligent Computing Theories and Technology. Berlin, Heidelberg: Springer, 571–578. https://doi.org/10.1007/978-3-642-39482-9_66
30. Bezuhlyi, V., Oliynyk, V., Romanenko, I., Zhuk, O., Kuzavkov, V., Borysov, O., Korobchenko, S., Ostapchuk, E., Davydenko, T., Shyshatskyi, A. (2021). Development of object state estimation method in intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 5 (3 (113)), 54–64. https://doi.org/10.15587/1729-4061.2021.239854
31. Mahdi, Q. A., Shyshatskyi, A., Prokopenko, Y., Ivakhnenko, T., Kupriyenko, D., Golian, V. et al. (2021). Development of estimation and forecasting method in intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 3 (9 (111)), 51–62. https://doi.org/10.15587/1729-4061.2021.232718
32. Sova, O., Radzivilov, H., Shyshatskyi, A., Shevchenko, D., Molodetskyi, B., Stryhun, V. et al. (2022). Development of the method of increasing the efficiency of information transfer in the special purpose networks. Eastern-European Journal of Enterprise Technologies, 3 (4 (117)), 6–14. https://doi.org/10.15587/1729-4061.2022.259727
33. Zhang, H., Zhu, Y., Chen, H. (2013). Root growth model: a novel approach to numerical function optimization and simulation of plant root system. Soft Computing, 18 (3), 521–537. https://doi.org/10.1007/s00500-013-1073-z
34. Labbi, Y., Attous, D. B., Gabbar, H. A., Mahdad, B., Zidan, A. (2016). A new rooted tree optimization algorithm for economic dispatch with valve-point effect. International Journal of Electrical Power & Energy Systems, 79, 298–311. https://doi.org/10.1016/j.ijepes.2016.01.028
35. Murase, H. (2000). Finite element inverse analysis using a photosynthetic algorithm. Computers and Electronics in Agriculture, 29 (1-2), 115–123. https://doi.org/10.1016/s0168-1699(00)00139-3
36. Zhao, S., Zhang, T., Ma, S., Chen, M. (2022). Dandelion Optimizer: A nature-inspired metaheuristic algorithm for engineering applications. Engineering Applications of Artificial Intelligence, 114, 105075. https://doi.org/10.1016/j.engappai.2022.105075

37. Paliwal, N., Srivastava, L., Pandit, M. (2020). Application of grey wolf optimization algorithm for load frequency control in multi-source single area power system. Evolutionary Intelligence, 15 (1), 563–584. https://doi.org/10.1007/s12065-020-00530-5
38. Dorigo, M., Blum, C. (2005). Ant colony optimization theory: A survey. Theoretical Computer Science, 344 (2-3), 243–278. https://doi.org/10.1016/j.tcs.2005.05.020
39. Poli, R., Kennedy, J., Blackwell, T. (2007). Particle swarm optimization. Swarm Intelligence, 1 (1), 33–57. https://doi.org/10.1007/s11721-007-0002-0
40. Bansal, J. C., Sharma, H., Jadon, S. S., Clerc, M. (2014). Spider Monkey Optimization algorithm for numerical optimization. Memetic Computing, 6 (1), 31–47. https://doi.org/10.1007/s12293-013-0128-0
41. Yeromina, N., Kurban, V., Mykus, S., Peredrii, O., Voloshchenko, O., Kosenko, V. et al. (2021). The Creation of the Database for Mobile Robots Navigation under the Conditions of Flexible Change of Flight Assignment. International Journal of Emerging Technology and Advanced Engineering, 11 (5), 37–44. https://doi.org/10.46338/ijetae0521_05
42. Maccarone, A. D., Brzorad, J. N., Stone H. M. (2008). Characteristics And Energetics Of Great Egret And Snowy Egret Foraging Flights. Waterbirds, 31 (4), 541–549. https://doi.org/10.1675/1524-4695-31.4.541
43. Ramaji, I. J., Memari, A. M. (2018). Interpretation of structural analytical models from the coordination view in building information models. Automation in Construction, 90, 117–133. https://doi.org/10.1016/j.autcon.2018.02.025
44. Pérez-González, C. J., Colebrook, M., Roda-García, J. L., Rosa-Remedios, C. B. (2019). Developing a data analytics platform to support decision making in emergency and security management. Expert Systems with Applications, 120, 167–184. https://doi.org/10.1016/j.eswa.2018.11.023
45. Chen, H. (2018). Evaluation of Personalized Service Level for Library Information Management Based on Fuzzy Analytic Hierarchy Process. Procedia Computer Science, 131, 952–958. https://doi.org/10.1016/j.procs.2018.04.233
46. Chan, H. K., Sun, X., Chung, S.-H. (2019). When should fuzzy analytic hierarchy process be used instead of analytic hierarchy process? Decision Support Systems, 125, 113114. https://doi.org/10.1016/j.dss.2019.113114
47. Osman, A. M. S. (2019). A novel big data analytics framework for smart cities. Future Generation Computer Systems, 91, 620–633. https://doi.org/10.1016/j.future.2018.06.046
48. Nechyporuk, O., Sova, O., Shyshatskyi, A., Kravchenko, S., Nalapko, O., Shknai, O. et al. (2023). Development of a method of complex analysis and multidimensional forecasting of the state of intelligence objects. Eastern-European Journal of Enterprise Technologies, 2 (4 (122)), 31–41. https://doi.org/10.15587/1729-4061.2023.276168
49. Merrikh-Bayat, F. (2015). The runner-root algorithm: A metaheuristic for solving unimodal and multimodal optimization problems inspired by runners and roots of plants in nature. Applied Soft Computing, 33, 292–303. https://doi.org/10.1016/j.asoc.2015.04.048
50. Poliarush, O., Krepych, S., Spivak, I. (2023). hybrid approach for data filtering and machine learning inside content management system. Advanced Information Systems, 7 (4), 70–74. https://doi.org/10.20998/2522-9052.2023.4.09

51. Balochian, S., Baloochian, H. (2019). Social mimic optimization algorithm and engineering applications. Expert Systems with Applications, 134, 178–191. https://doi.org/10.1016/j.eswa.2019.05.035
52. Lenord Melvix, J. S. M. (2014). Greedy Politics Optimization: Metaheuristic inspired by political strategies adopted during state assembly elections. 2014 IEEE International Advance Computing Conference (IACC), 1157–1162. https://doi.org/10.1109/iadcc.2014.6779490
53. Moosavian, N., Roodsari, B. K. (2014). Soccer League Competition Algorithm, a New Method for Solving Systems of Nonlinear Equations. International Journal of Intelligence Science, 4 (1), 7–16. https://doi.org/10.4236/ijis.2014.41002
54. Hayyolalam, V., Pourhaji Kazem, A. A. (2020). Black Widow Optimization Algorithm: A novel meta-heuristic approach for solving engineering optimization problems. Engineering Applications of Artificial Intelligence, 87, 103249. https://doi.org/10.1016/j.engappai.2019.103249
55. Abualigah, L., Yousri, D., Abd Elaziz, M., Ewees, A. A., Al-qaness, M. A. A., Gandomi, A. H. (2021). Aquila Optimizer: A novel meta-heuristic optimization algorithm. Computers & Industrial Engineering, 157, 107250. https://doi.org/10.1016/j.cie.2021.107250
56. Hodlevskyi, M., Burlakov, G. (2023). Information technology of quality improvement planning of process subsets of the spice model. Advanced Information Systems, 7 (4), 52–59. https://doi.org/10.20998/2522-9052.2023.4.06
57. Askari, Q., Younas, I., Saeed, M. (2020). Political Optimizer: A novel socio-inspired meta-heuristic for global optimization. Knowledge-Based Systems, 195, 105709. https://doi.org/10.1016/j.knosys.2020.105709
58. Mohamed, A. W., Hadi, A. A., Mohamed, A. K. (2019). Gaining-sharing knowledge based algorithm for solving optimization problems: a novel nature-inspired algorithm. International Journal of Machine Learning and Cybernetics, 11 (7), 1501–1529. https://doi.org/10.1007/s13042-019-01053-x
59. Gödri, I., Kardos, C., Pfeiffer, A., Váncza, J. (2019). Data analytics-based decision support workflow for high-mix low-volume production systems. CIRP Annals, 68 (1), 471–474. https://doi.org/10.1016/j.cirp.2019.04.001
60. Harding, J. L. (2013). Data quality in the integration and analysis of data from multiple sources: some research challenges. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XL-2/W1, 59–63. https://doi.org/10.5194/isprsarchives-xl-2-w1-59-2013
61. Orouskhani, M., Orouskhani, Y., Mansouri, M., Teshnehlab, M. (2013). A Novel Cat Swarm Optimization Algorithm for Unconstrained Optimization Problems. International Journal of Information Technology and Computer Science, 5 (11), 32–41. https://doi.org/10.5815/ijitcs.2013.11.04
62. Karaboga, D., Basturk, B. (2007). A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. Journal of Global Optimization, 39 (3), 459–471. https://doi.org/10.1007/s10898-007-9149-x
63. Fister, I., Fister, I., Yang, X.-S., Brest, J. (2013). A comprehensive review of firefly algorithms. Swarm and Evolutionary Computation, 13, 34–46. https://doi.org/10.1016/j.swevo.2013.06.001
64. Sova, O., Radzivilov, H., Shyshatskyi, A., Shvets, P., Tkachenko, V., Nevhad, S. et al. (2022). Development of a method to improve the reliability of assessing the condition of the monitoring object in spe-

cial-purpose information systems. Eastern-European Journal of Enterprise Technologies, 2 (3 (116)), 6–14. https://doi.org/10.15587/1729-4061.2022.254122

65. Khudov, H., Khizhnyak, I., Glukhov, S., Shamrai, N., Pavlii, V. (2024). The method for objects detection on satellite imagery based on the firefly algorithm. Advanced Information Systems, 8 (1), 5–11. https://doi.org/10.20998/2522-9052.2024.1.01
66. Owaid, S. R., Zhuravskyi, Y., Lytvynenko, O., Veretnov, A., Sokolovskyi, D., Plekhova, G. et al. (2024). Development of a method of increasing the efficiency of decision-making in organizational and technical systems. Eastern-European Journal of Enterprise Technologies, 1 (4 (127)), 14–22. https://doi.org/10.15587/1729-4061.2024.298568
67. Tyurin, V., Bieliakov, R., Odarushchenko, E., Yashchenok, V., Shaposhnikova, O., Lyashenko, A. et al. (2023). Development of a solution search method using an improved locust swarm algorithm. Eastern-European Journal of Enterprise Technologies, 5 (4 (125)), 25–33. https://doi.org/10.15587/1729-4061.2023.287316
68. Yakymiak, S., Vdovytskyi, Y., Artabaiev, Y., Degtyareva, L., Vakulenko, Y., Nevhad, S. et al. (2023). Development of the solution search method using the population algorithm of global search optimization. Eastern-European Journal of Enterprise Technologies, 3 (4 (123)), 39–46. https://doi.org/10.15587/1729-4061.2023.281007
69. Mohammed, B. A., Zhuk, O., Vozniak, R., Borysov, I., Petrozhalko, V., Davydov, I. et al. (2023). Improvement of the solution search method based on the cuckoo algorithm. Eastern-European Journal of Enterprise Technologies, 2 (4 (122)), 23–30. https://doi.org/10.15587/1729-4061.2023.277608
70. Raskin, L., Sukhomlyn, L., Sokolov, D., Vlasenko, V. (2023). Multi-criteria evaluation of the multifactor stochastic systems effectiveness. Advanced Information Systems, 7 (2), 63–67. https://doi.org/10.20998/2522-9052.2023.2.09
71. Arora, S., Singh, S. (2018). Butterfly optimization algorithm: a novel approach for global optimization. Soft Computing, 23 (3), 715–734. https://doi.org/10.1007/s00500-018-3102-4

DOI: 10.15587/978-617-8360-23-8.CH3

3 DEVELOPMENT OF THE SCIENTIFIC AND METHODOLOGICAL FRAMEWORK FOR THE INTELLECTUAL ASSESSMENT OF PARAMETERS IN DECISION SUPPORT SYSTEMS

Andrii Shyshatskyi, Ganna Plekhova, Danylo Pliekhov, Oleksii Nalapko, Yuliia Vakulenko, Andrii Lebedynskyi

ABSTRACT

The object of the research is decision support systems. The subject of the research is the process of evaluating parameters in decision support systems. The study proposes: a methodology for the intellectual assessment of parameters in decision support systems and a method for multi-criteria evaluation of hierarchical systems. The originality of the research lies in the use of additional advanced procedures that allow:

– verification of the topology and parameters of decision support systems, taking into account the degree of uncertainty in the input data regarding the known information, achieved through the use of an enhanced penguin swarm algorithm. This reduces the time required for the initial configuration of the evaluation methodology during its setup;

– initial selection of individuals for configuring the evolving artificial neural network, carried out using an enhanced genetic algorithm, which reduces solution search time and increases the reliability of the obtained results;

– exploration of solution spaces for the problem of parameter evaluation in decision support systems, which are described by atypical functions, using the enhanced penguin swarm algorithm;

– configuration of the weights of the evolving artificial neural network improves the accuracy of parameter evaluation in decision support systems;

– utilization of additional mechanisms for correcting the parameters of the evolving artificial neural network is applied by a procedure for modifying the membership function;

– reliability of parameter evaluation in decision support systems is increased by parallel evaluation through multiple assessment methods;

– use of hybrid parameter evaluation for decision support systems allows for correct operation in the absence of conditions of stationarity, homogeneity, normality, and independence.

An example of the application of the proposed scientific and methodological framework for evaluating parameters in decision support systems showed a 25% increase in evaluation reliability, achieved by utilizing additional procedures while maintaining the required operational efficiency.

KEYWORDS

Artificial neural networks, enhanced genetic algorithm, destabilizing factors, metaheuristic algorithm.

The problem of improving the reliability of parameter evaluation in decision support systems is becoming increasingly urgent in modern information systems with various functional purposes [1]. The experience from recent conflicts, involving the use of modern information systems, shows that existing approaches to evaluating parameters in decision support systems do not allow for reliable assessments with the required operational speed [2].

This issue is linked to the following reasons:

– the significant role of the human factor in the evaluation process of decision support system parameters [3];

– the large number of diverse components in decision support systems [3];

– parameter evaluation in decision support systems occurs under conditions of uncertainty, which causes delays in their processing [4];

– the presence of many destabilizing factors that affect the reliability of the parameter evaluation in decision support systems;

– the presence of both structured and unstructured data in decision support systems that need to be processed, among other factors.

Given the diversity, numerous destabilizing factors, and the various dimensions of the indicators describing them, the need for evaluating parameters in decision support systems prompts the search for new approaches. One such approach is the use of metaheuristic algorithms [5–8].

The use of metaheuristic algorithms in their canonical form can improve the operational speed of parameter evaluation in decision support systems. However, further increasing the operational speed of parameter evaluation leads to a deterioration in the reliability of parameter assessments.

This motivates the introduction of various strategies to improve the convergence speed and accuracy of basic metaheuristic algorithms when evaluating parameters in decision support systems. One approach to improving the reliability of parameter evaluation is its further enhancement by combining, comparing, and developing new procedures for their joint use.

An analysis of the works [9–71] shows that common shortcomings in the aforementioned research include:

– the lack of a hierarchical system of indicators for comprehensive evaluation of decision support systems;

– the failure to account for the computational resources of the system managing the evaluation process of decision support system parameters;

– the absence of mechanisms for adjusting the indicator system managing the evaluation process of decision support system parameters;

– the lack of selective engagement of artificial neural network training methods;

– high computational complexity;

– the failure to account for computational (hardware) resources available in the system;

– the absence of prioritized search in a specific direction.

The aim of this research is to develop a methodology for the intellectual assessment of parameters in decision support systems. This will improve the reliability of parameter evaluation in decision support systems with the required operational speed and the generation of subsequent management decisions based on intellectual evaluation.

This will enable the development (or improvement) of software for the operation of decision support systems.

To achieve this aim, the following tasks were set:

– define the algorithm for implementing the methodology;

– provide an example of applying the methodology for the intellectual evaluation of parameters in decision support systems;

– offer recommendations for integrating the proposed methodology into decision support systems.

The object of research is decision support systems. The problem addressed in the research is improving the reliability of parameter evaluation in decision support systems while ensuring the required operational speed, regardless of the volume of incoming data. The subject of research is the process of evaluating parameters in decision support systems.

Parameters in the evaluation system of decision support systems generally include various types of origin and units of measurement, as well as varying degrees of impact on the overall evaluation result. To address this, it is appropriate to use artificial intelligence theory, specifically:

– an enhanced genetic algorithm, which allows automating the evaluation process and conducting random, ordered changes to information and the rearrangement of individuals in the parameter evaluation plane of decision support systems. This enhanced genetic algorithm is also used in this research for the preliminary selection of individuals to improve the reliability of parameter evaluation in decision support systems. The enhanced genetic algorithm is also used for tuning the parameters of an evolving artificial convolutional neural network;

– an enhanced penguin swarm algorithm – for verifying the topology and parameters of decision support systems, as well as the topology and parameters of destabilizing influencing factors. This leads to an increase in the reliability of the obtained parameter evaluation in decision support systems;

– evolving artificial neural networks – enabling the generalized evaluation of decision support system parameters, which are of different origins and units of measurement, taking into account the number of input parameters to be assessed.

The hypothesis of the research is that the reliability of parameter evaluation in decision support systems can be improved with the required operational speed using the intellectual evaluation methodology.

The modeling of the proposed methodology was carried out in the Microsoft Visual Studio 2022 programming environment (USA). The task solved in the simulation process was the determination of the composition of a military grouping (forces). The hardware used in the research process is the AMD Ryzen 5.

Parameters for the operation of the enhanced algorithm:

– number of iterations – 25;

– number of individuals in the swarm algorithm – 25;

– range of feature space – [–100, 100].

The structure of the evolving artificial neural network is presented in the work [20].

3.1 ALGORITHM FOR IMPLEMENTING THE INTELLECTUAL PARAMETER EVALUATION METHODOLOGY IN DECISION SUPPORT SYSTEMS

The methodology for the intellectual evaluation of parameters in decision support systems consists of the following sequence of actions:

Action 1. Input of initial data.

At this stage, the available initial data on decision support systems and destabilizing influencing factors are input, specifically:

– the number and type of technical means included in the decision support systems;

– the number and type of destabilizing factors that affect the objectivity of the evaluation of the state of the decision support systems;

– technical characteristics of the means included in the decision support systems;

– technical characteristics of destabilizing factors that affect the objectivity of the evaluation of the state of decision support systems;

– topology of connections within the decision support systems;

– topology of connections of destabilizing factors;

– the type of data circulating within decision support systems;

– available computational resources of the decision support systems;

– information about the operational environment of the decision support systems, etc.

This procedure involves the processing of arrays at the initial observation window, exponential normalization of the data, and setting tasks for the learning, testing, and forecasting processes.

Action 2. Verification of parameters necessary for calculations.

At this stage, the initial data about the decision support system and destabilizing factors are clarified. This is done by taking into account the type of uncertainty about the state of decision support systems using the enhanced penguin swarm algorithm proposed by the authors in work [20].

Action 3. Formation of the topology of the evolving artificial neural network.

At this stage, the enhanced penguin swarm algorithm is used to form the topology of the evolving artificial neural network, proposed by the authors in work [20], based on the verified data.

Action 4. Preliminary selection of individuals for the genetic algorithm.

To improve the reliability of the obtained solutions, the preliminary selection of individuals is carried out using the enhanced genetic algorithm proposed by the authors in study [19]. The enhanced genetic algorithm is further used in *Action 5.3*.

Action 5. Parallel evaluation of the decision support system's state using multiple approaches.

Action 5.1. Evaluation of the decision support system's state based on the multiple regression algorithm.

The traditional technology for sequential evaluation and forecasting the state of decision support systems based on observations containing a stochastic component relies on the mathematical apparatus of multivariate regression [13].

In general, multivariate regression generalizes the one-dimensional linear regression algorithm to the situation of multiple interdependent variables *X*, which define the structure of the base model.

The multiple regression algorithm includes the description of the dependence of the predicted parameters on the values of the input parameters, i.e., the regressors, which are the control parameters of the decision support system.

For linear forecasting, the application of regression analysis is based on the possibility of sequentially evaluating the state parameters, control, and output parameters of decision support systems.

Let's assume that the average values of the predicted output characteristics of the decision support systems $Z_{k+\tau}=\left(z_1,\ldots,z_{M_y}\right)_{k+\tau}$, $k=1,\ldots,N$ are related to the state parameters, which include control parameters $X_k=\left(x_1,\ldots,x_{M_X}\right)_k$, $k=1,\ldots,N$ in a functional dependency of the form

$$Z_{k+\tau}=f\left(X_k\right)+V_k,\ k=1,\ldots,N. \tag{3.1}$$

It is assumed that the additive noise (in our case, intentional destructive influence) is centered $EV_k=0$, $k=1,\ldots,N$.

The task of regression evaluation is to establish the form of the relationship between dependent and independent variables over time. For the task of corrective control, the functional dependency (3.1) allows for linearization, which makes it possible to restrict the model to linear regression

$$Z_{k+\tau}=C_kX_k+V_k,\ k=1,\ldots,N. \tag{3.2}$$

The rapid aging of data, caused by the transient nature of military grouping (force) operations, formed by a non-stationary process, results in the use of a multidimensional sample on a sliding observation window of size L as the initial data. In this case, the output data arrays at each forecasting step are specified by matrices

$$X_{L:M_X}=\begin{bmatrix} x_{11}, & x_{12} & \ldots & x_{1M_x} \\ x_{21}, & x_{22} & \ldots & x_{2M_x} \\ \ldots & \ldots & \ldots & \ldots \\ x_{L1}, & x_{L2} & \ldots & x_{LM_x} \end{bmatrix};\ Z_{L:M_Y}=\begin{bmatrix} z_{11}, & z_{12} & \ldots & z_{1M_y} \\ z_{21}, & z_{22} & \ldots & z_{2M_y} \\ \ldots & \ldots & \ldots & \ldots \\ z_{L-\tau,1}, & z_{L-\tau,2} & \ldots & z_{L-\tau,M_y} \end{bmatrix}.$$

Then, based on the minimization of the quadratic functional

$$V^TV=\left(Z-XC\right)^T\left(Z-XC\right)=Z^TZ-2C^TX^TZ+C^TX^TXC,$$

i.e., using the least squares method, it is possible to obtain the well-known matrix expression for the predictive transfer coefficient of linear regression

$$C=\left(X^TX\right)^{-1}X^T. \tag{3.3}$$

In this case, the linear regression forecasting algorithm is described by the simplest matrix relationship of the form

$$Z_{k+\tau} = C_k X_k, \tag{3.4}$$

where $Z_{k+\tau} = \left(z_1, \dots, z_{M_Z}\right)^T_{k+\tau}$, the regressors use only those state parameters of the decision support system (decision support system) components that allow manipulation of values during the control process, i.e., control parameters $U_k = \left(u_1, \dots, u_{M_U}\right)^T_k$.

The traditional linear regression scheme includes important assumptions known as the Gauss-Markov conditions [13, 14]. This algorithm fits the requirements for adaptation, which is associated with the change in the coefficient of pairwise correlation.

The second feature of the developed algorithm is the application of a sliding observation window. It is important to note that within this window, the output array of forecasted parameter values for decision support systems $Z_{L:M_Y}$ should be shifted backwards by τ time steps relative to the array of regressors $X_{L:M_X}$.

The main forecasting cycle occurs over the sliding observation interval. At each step, the current mean values and covariance structures are corrected.

The forecast of the parameter values for decision support systems is performed by the previously described method of vector multiplication of the current centered monitoring data values and the matrix transfer coefficient of the least squares filter. The justification for the optimality of this approach directly follows from the well-known Gauss-Markov theorem [9].

The values of the parameters for the decision support system at the output of the predictor form the vector of the evolution of the output parameters. Typically, the quality indicators used are the root mean square deviation (RMSD) of the forecast or the average values of the obtained errors, which allows for a forecast that is quite close to the actual process trajectory (the average relative error does not exceed 9%).

Action 5.2. Evaluation of the decision support system's state based on the enhanced canonical correlation method.

The enhanced canonical correlation method is a generalization of multiple correlation for the case where there are two or more interrelated variables *X* and *Y* [9, 18]. From the perspective of forming a linear forecast, the application of canonical correlations means the ability to simultaneously evaluate a group of interrelated output parameters, considered as generalized linear combinations of interrelated parameters. Let's consider the mathematical apparatus of canonical correlations. Let's define possible linear combinations for *q* variables *Y* and *p* variables *X* in the general population

$$X^* = \sum_{i=1}^{p} \alpha_i X_i; \; Y^* = \sum_{j=1}^{q} \beta_j Y_j.$$

The tasks of canonical correlations include determining the coefficients α_i and β_i [10].

Let's consider the algorithm for multivariate analysis based on the enhanced canonical correlation method, where the output array is divided into observed and unobserved parts: $X \in N_p\{\mu_1, P_1\}$ and $Y \in N_q\{\mu_2, P_2\}$, respectively. In this case, the covariance matrix will take the following form

$$P = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix}. \tag{3.5}$$

Then, the correlation coefficient will be

$$\rho_c = \frac{\operatorname{cov}(X\alpha, Y\beta)}{\sqrt{\operatorname{var}(X\alpha)\cdot \operatorname{var}(Y\beta)}} = \frac{\alpha^T P_{12}\beta}{\sqrt{(\alpha^T P_{11}\alpha)\cdot(\beta^T P_{22}\beta)}}.$$

Now, let's assume that the state of the decision support system at time k is described (with sufficient accuracy) by m parameters, combined into a vector $X = (x_1, \ldots, x_m)$. Geometrically, this means that the state of the decision support system is a point in the m-dimensional phase space R^m. Let there be measurement results for these parameters at time moments $k = 1, \ldots, N$.

Let's combine the obtained measurement results of the decision support system parameters into a matrix X of size $n \times m$. The row of this matrix with index i corresponds to the result of the i-th vector measurement, $i = 1, \ldots, n$, while the column with index j represents the set of n measurement values of the j-th parameter of the decision support system in each measurement, $j = 1, \ldots, m$.

The task is to refine or create a mathematical model based on the available data that fits the tasks of both forecasting parameter values and control. In the case of non-stationary processes, the model is not universal and requires constant reconfiguration.

If the data is normalized, estimates of the covariance and correlation matrices are constructed. The second estimate (the correlation matrix) involves normalizing the standard deviation; if this operation has already been performed, the results of these estimates coincide. These calculations are carried out for known values of α and P, which are replaced by their estimates. The quality of the estimate will depend on the size and reliability of the initial data on the parameters of the decision support system.

Consider the following task: there are m parameters, of which p are observable, and the remaining q are unobservable. The task is to estimate the predicted parameters based on the available output data.

As an assumption, let's consider that the initial data is already normalized and centered. The covariance matrix will take the form (3.5), where P_{11} – the covariance matrix of the observed parameters, which values are obtained during monitoring, P_{22} – the covariance matrix of the predicted parameters, and P_{12} – the cross-covariance matrix between the observed and unobserved parameters.

The task is to find the matrix of weighting coefficients C under the condition of minimizing the average sum of squared residuals

$$trE\left[(x_2 - C\cdot x_1)^T (x_2 - C\cdot x_1)\right] \to \min.$$

Let's transform this expression into the form

$$trE\left[\left(x_2 - C\cdot x_1\right)^T\left(x_2 - C\cdot x_1\right)\right]=$$
$$=tr\left[E\left(x_2x_2^T\right)-C\cdot E\left(x_1x_2^T\right)-E\left(x_2x_1^T\right)\cdot C^T + C\cdot E\left(x_1x_1^T\right)\cdot C^T\right]=$$
$$=tr\left[P_{22}-C\cdot P_{12}-P_{21}^T\cdot C^T + C\cdot P_{11}\cdot C^T\right]\to \min.$$

By substituting the initial data, the formula for the optimal linear estimate of the vector X_2 based on the known vector X_1 will take the form

$$\hat{X}_2 = E\left(X_2\right)+P_{12}^T\cdot P_{11}^{-1}\cdot\left(X_1 - E\left(X_1\right)\right). \quad (3.6)$$

Now, using sample estimates

$$\hat{X}_2 = \overline{X}_2 + \hat{P}_{12}^T\cdot \hat{P}_{11}^{-1}\cdot\left(X_1 - \overline{X}_1\right).$$

In terms of the task of forecasting the output characteristics of the decision support system, expression (3.7) will take the form

$$\tilde{Z}_{t+\tau} = \overline{Z}_{t-L,t} + \hat{P}_{UZ}^T\cdot \hat{P}_{u}^{-1}\cdot\left(U_t - \overline{U}_{t-L,t}\right). \quad (3.7)$$

Substituting the found value of the matrix into the expression for the average sum of squared residuals, the covariance matrix of the estimation errors will take the form

$$P_v = P_{22} - 2C\cdot P_{12} + C\cdot P_{11}\cdot C^T = P_{22} - 2P_{12}^T\cdot P_{11}^{-1}\cdot P_{12} + P_{12}^T\cdot P_{11}^{-1}\cdot P_{11}\cdot P_{11}^{-1}\cdot P_{12} =$$
$$= P_{22} - P_{12}^T\cdot P_{11}^{-1}\cdot P_{12}. \quad (3.8)$$

The diagonal elements of this matrix represent the variance of the estimates of the corresponding components (in dimensionless units). Typically, confidence intervals calculated using this formula turn out to be overly pessimistic. The variances can be determined by calculating the sum of the squared errors in the prediction based on the available dataset, but they will only be relevant to this specific data. The connection of these variances with the corresponding theoretical characteristics depends on the size of the available sample of parameters for the decision support system.

At the same time, training is carried out based on data from a sliding observation window.

Action 5.3. Hybrid parameter estimation for the decision support system.

As mentioned earlier, the statistical algorithms proposed above for parameter estimation in decision support systems provide the best solution when a number of assumptions (stationarity, homogeneity, normality, independence, etc.) are met, which, in practice, are not always valid.

However, a complete rejection of statistical algorithms for forecasting the parameters of decision support systems is also irrational. The universality of the quadratic criterion allows obtaining good initial approximations to the averaged dynamics of the forecasted process. Hence, the study proposes the development of a hybrid algorithm that combines multivariate statistical analysis algorithms with a computational scheme that self-develops, based on evolutionary modeling methods. The core idea is to replace the optimization of a dynamic system with its evolutionary process. In fact, this refers to the stochastic self-organization of the applied mathematical model.

Let's assume that, based on the traditional statistical algorithm $A\{S(A),x\}$, characterized by a given structure $S(A)$ and a set of parameters x, the necessary output parameter of the decision support system $\hat{y}$ is estimated.

In this case, the effectiveness of the algorithm $Eff(A)$ is evaluated based on its application to the output data from the sliding window. The effectiveness indicator is typically represented by the general quality metrics described earlier or local accuracy measures, such as the total square of the prediction error.

At this stage, the second part of the enhanced genetic algorithm, proposed in work [19], is used. Let's introduce two operators: the variability operator $Var(A): A \Rightarrow \{A_1,\ldots,A_{N_a} : A_i \neq A_j \neq A, \forall i,j\}$ and the selection operator $Sel(A_1,\ldots,A_{N_g}) : \{A_1,\ldots,A_{N_g}\} \Rightarrow \{A_{<1>},\ldots,A_{<N_a>} : Eff(A_{<1>}) \geq \ldots \geq Eff(A_{<N_a>}) \geq Eff(A_j), \forall j > N_a\}$, where N_a – the number of "selected" algorithms that are used for further reproduction; N_g=N_a(1+ N_d) – the number of strategies for one generation that are subject to selection, N_d – the number of strategy offspring generated according to the specified rules at each iteration.

Let A_0=$A\{S_0(x), x_0\}$ – be a certain variant of the forecasting algorithm with given parameters and structure, accepted as the baseline "father" algorithm. Then, the technology of evolutionary modeling reduces to the repeated application of the sequence of operators

$$\begin{array}{c} A_o \Rightarrow Var(A_0) = \{A_a\} = \{A_1,\ldots,A_{N_g}\} \Rightarrow Var(A_d) = \{A_d\} = \{A_1,\ldots,A_{N_d}\} \\ \Uparrow \qquad\qquad\qquad\qquad \Downarrow \\ \psi(A_1,\ldots,A_{N_g}) = A_0 = \{A_{<1>},\ldots,A_{<N_a>}\} \Leftarrow \{A_g\} = \{A_a \cup A_d\}. \end{array} \quad (3.9)$$

The presented approach to evolutionary optimization, combined with the previously described algorithm based on the canonical correlation method, forms a unified hybrid algorithm. This algorithm retains all the advantages of statistical analysis and supplements them, allowing for the avoidance of the drawbacks associated with the lack of Gaussianity and stationarity in real observation series for the parameters of non-stationary complex technical objects.

Action 6. Formation of a generalized parameter evaluation for the decision support system.

Based on the evolving artificial neural network, a generalized evaluation of the state of the decision support system is formed. This is done through the convolution of each group of parameters for the system's state. The architecture of the evolving artificial neural network for evaluating the parameters of the decision support system is presented in work [20].

Action 7. Verification of the stop criterion for the combined algorithm.

The algorithm terminates if the maximum number of iterations has been reached. Otherwise, the generation of new positions and checking of conditions are repeated.

Action 8. Determining the number of required computational resources for evaluation.

To avoid the cyclic repetition of calculations in *Actions 1–8* of this method and to increase computational efficiency, the system's load is additionally determined. If the computational complexity exceeds the established threshold, the number of software-hardware resources that need to be added is determined using the method proposed in work [20].

Action 9. Training the knowledge bases of agents. At this stage, the training of the knowledge bases of agents from the list of bio-inspired algorithms used in this study is performed. The method of deep learning proposed in work [20] is used as the learning method.

End.

3.2 EXAMPLE OF APPLYING THE PROPOSED METHODOLOGY FOR PARAMETER EVALUATION IN DECISION SUPPORT SYSTEMS

The evaluation of the effectiveness of the proposed methodology for parameter evaluation in decision support systems, based on their own quality indicators, such as the root mean square deviation (3.7), the mean value of the sum of squared residuals (3.8), and other similar characteristics, allows comparing them by the degree of reliability of the evaluation and forecasting.

However, it does not provide answers to questions about the advisability of improving the values of these indicators. Like any mathematical or informational tool, the effectiveness of the proposed methodology for evaluation and forecasting can only be assessed through the quality indicators of the metasystem for which it was created and improved.

In this context, the metasystem is represented by the proactive decision support system. The performance indicators of such a system are the external or exogenous numerical characteristics that are hierarchically specified by the higher-level decision support system.

The suitability criterion for the forecasting algorithm (3.9) is the verification of the condition for the membership of the forecasted values of the parameters in the state vector of the decision support system within the constraint set $\left\{\left|x_i^* \pm \Delta_i\right| \wedge \Omega_{per}^i\right\}$, $\forall i = 1,\ldots,M$.

Here $x_i^* \pm \Delta_i$, $\forall i = 1,\ldots,M$ – the sets of constraints that correspond to the requirement for stabilizing the values of the decision support system's parameters around a reference value x_i^*, $\forall i = 1,\ldots,M$, which is determined by the regulation of the controlled parameter in the decision support system, Ω_{per} – the set of technical constraints imposed on the parameters of the decision support system.

As an example, let's build a proactive parameter evaluation system for the decision support system based on the algorithm for iterating through the possible values of control parameters.

The formation of the ε-neighborhood $\Delta = \Delta\left(U_0\left(t\right)\right)$ can be carried out using several methods:

1) $\Delta = \left[U_0 - R/2; U_0 + R/2\right]$, where $R = abs\left(U_{max} - U_{min}\right)$; U_{max}, U_{min} – the boundaries of the acceptable range of the control parameter changes in the decision support system;

2) $\Delta = \left[U_0 - s\left(U\right); U_0 + s\left(U\right)\right]$, where $s\left(U\right)$ – the standard deviation of the change in the decision support system's parameter;

3) $\Delta = \left[U_0 - t_0 * s / sqrt\left(N\right); U_0 + t_0 * s / sqrt\left(N\right) \right]$, where t_0 – the critical value of the t-statistic for the Student's distribution at the chosen confidence level α, N – the number of observations in the sliding window;

4) $\Delta = \left[U_0 - \%R * U_0; U_0 + \%R * U_0 \right]$, where $\%R$ – the half-interval used to search for the best solution (for example, $\%R$=0.05 for a 5%-th deviation).

Next, the number of steps for iterating within the parameter change range is established N_{step} (e.g., 10). The total number of possible variants of the parameter evaluations for the decision support system, formed as the number of permutations with repetition, is given by $\left(N_{step}\right)^{M_{man}}$, where M_{man} – the number of parameters in the decision support system that are used for manipulation.

It is important to note that the number of possible controls grows rapidly with an increase in N_{step} and M_{man}. Examples of the number of possible evaluation variants for different values of N_{step} and M_{man} are presented in **Table 3.1**.

Table 3.1 Number of iterations for parameter evaluation variants in the decision support system

M_{man}	N_{step}				
	5	10	15	20	25
2	25	100	225	400	625
3	125	1000	3375	8000	15625
4	625	10000	50625	160000	390625
5	3125	100000	759375	3200000	9765625

Considering that each step is associated with a considerable number of operations, including the calculation of inverse matrices, the growth of these parameters should be done while taking into account the computational capabilities of the hardware. According to the adopted algorithm, for each variant of parameter evaluation, a forecast is formed based on regression, neural network, or other technologies. Comparing the forecasted output parameter values against each other, considering the set of technological constraints imposed on the parameters of the decision support system, allows for the direct identification of the optimal value of the state parameter of the decision support system at a given time.

Comparing the obtained result with the traditional scheme of situational assessment, which is implemented during the management of the decision support system, allows for the evaluation of the terminal effectiveness of hybrid evaluation and forecasting through the quality indicators of the higher-level system for which it was created.

3.3 RECOMMENDATIONS FOR INTEGRATING THE PROPOSED METHODOLOGY INTO DECISION SUPPORT SYSTEMS

As an example of implementing hybrid evaluation and forecasting of decision support system parameters for non-stationary processes, let's consider the option of building it based on the back estimation (BE) procedure of the state parameters of the decision support system. The formation of improved state evaluation of the decision support system is carried out by sequential (step-by-step) modification of the chosen initial output parameter and the back-calculation of the output parameters (with improved outcomes) into manipulation parameters (control parameters used in the current situation).

Let's consider the formalized formulation of intellectual evaluation and forecasting of parameter values for the decision support system based on the development of an algorithm for back estimation of possible parameter values.

There is an initial value of the parameter U_0, obtained from the data about the decision support system under consideration. Then, using a predefined improvement step δY for the system's state indicator $Y = Y_0 + \delta Y$, the evaluation effectiveness according to the defined evaluation criterion is found to be higher, i.e., $Eff\left(Y\right) > Eff\left(Y_0\right)$.

The step size is selected taking into account the physical and technical characteristics of the specific decision support system. In the considered example, as already mentioned, it was chosen as 2–3% of the forecast, estimated based on the current state of the decision support system. Using data from a limited sliding window, as described earlier, the parameters of the intellectual parameter evaluation methodology are refined.

This methodology is based on generalized linear regression and the linking parameter that relates the state evaluation parameter to its output parameters and state parameters $\tilde{Y}_{k+1} = F\left(U_k\right),\ k = L+1,\ldots,N$. For a non-degenerate operator F it is possible to construct an inverse mapping $\widehat{U}_k^* = F^{-1}\left(\tilde{Y}_{k+1} + \delta Y_{k+1}\right)$, which allows obtaining the values of the control parameters $\widehat{U}_k^*$, that have increased efficiency compared to the reference control being compared. At the same time, it is necessary to additionally verify the condition of admissibility for the found control values $\widehat{U}_k^*$ and other state parameters of the decision support system $\widehat{X}_k^*$, i.e., the membership of the corresponding numerical values of the parameters in the set of admissible values (3.11).

The consideration of the variation of state parameters in the decision support system is carried out by using a sliding observation window. The size of the window is selected based on the dynamics of variation in the average values of the controlled parameters.

The core of neural network-based forecasting technologies is the iterative refinement of the weight coefficients of the multiplicative inputs of nonlinear nodes, unified by a single network structure [19, 20].

The process of correcting weight coefficients is carried out according to the feedback signal, formed by the difference between the network's output signals and the actual measured values, combined with the corresponding input signals into the training dataset. Let's consider an example of evaluating and forecasting the state of the decision support system.

Let $\left(x_1, x_2, \ldots, x_p\right)^T$ – be the input parameters, $w^1 = \left(w_{11}^1, w_{22}^1, \ldots, w_{p1}^1\right)^T$, $w^{12} = \left(w_{11}^2, w_{22}^2, \ldots, w_{p1}^2\right)^T$ – be the boosting coefficients of the first and second generations of the models. The artificial neural network, with an evolving structure, has a different number of neurons at each level: level A (input layer) – p neurons, level S (first layer) – l neurons, and level R (second layer) – k neurons.

Let N be the number of input and output points obtained during the experiment or through simulation, $X = \left(X_1, X_2, \ldots, X_p\right)$ – be the input vector, $D = \left(d_1, d_2, \ldots, d_k\right)$ – be the real or computed outputs [15, 18].

The objective function to be minimized is as follows

$$E(\varpi)=\frac{1}{2}\left[\sum_{i=1}^{N}\sum_{j=1}^{l}\left(y_{ij}^{1}-d_{ij}^{1}\right)+\sum_{i=1}^{N}\sum_{j=1}^{k}\left(y_{ij}^{2}-d_{ij}^{2}\right)\right]^{2}=\frac{1}{2}\sum_{i=1}^{N}\left(\sum_{j=1}^{l}\left(y_{ij}^{1}-d_{ij}^{1}\right)+\sum_{j=1}^{k}\left(y_{ij}^{2}-d_{ij}^{2}\right)\right)^{2}.$$

Minimization is achieved through gradient descent, meaning the adjustment of weight coefficients is formulated as

$$\Delta\varpi_{ij}^{(n)}=-\eta\frac{\partial E}{\partial\omega_{ij}},\quad n=1,2,$$

where $\varpi_{ij}^{(n)}$ – the weight characteristic of the connection between the *i*-th neuron of the *n*-1 level and the *j*-th neuron of the *n*-th level is *n*, $1<0<\eta$ – the learning rate coefficient. It is known that $\frac{\partial E}{\partial\omega_{ij}}=\frac{\partial E}{\partial y_j}\cdot\frac{\partial y_j}{\partial S_j}\cdot\frac{\partial S_j}{\partial\omega_{ij}}$, where y_j – the output of the neuron, $S_j=\sum_{i=1}^{N}\omega_{ij}x_{ij}$ – the weighted sum of its input signals (the argument of the activation function). Typical activation functions used are the sigmoid $A=\frac{1}{1+e^{-\sum\omega_i x_i}}$ or the hyperbolic tangent $A=\operatorname{th}x=\frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$, $\operatorname{th}'x=\frac{1}{ch^{2}(x)}$, where $ch(x)$ – the hyperbolic cosine, th *x* is the hyperbolic tangent, $\left(\operatorname{th}x\right)'=1-\left(\operatorname{th}x\right)^{2}$. The third factor represents the output of the previous layer neuron. The first factor can be expanded as $\frac{\partial E}{\partial y_i}=\sum_k\frac{\partial E}{\partial y_k}\cdot\frac{\partial y_k}{\partial S_k}\cdot\frac{\partial S_k}{\partial y_i}=\sum_k\frac{\partial E}{\partial y_k}\cdot\frac{\partial y_k}{\partial S_k}$.

The last sum is calculated over the neurons of the (*n-1*)-th layer. Let's introduce a new substitution: $\delta_j^{(n)}=\frac{\partial E}{\partial y_j}\cdot\frac{\partial y_j}{\partial S_j}$ which gives the recursive formula $\delta_j^{(n)}=\left[\sum_k\delta_k^{(n+1)}\cdot\omega_{jk}^{(n+1)}\right]\cdot\frac{\partial y_j}{\partial S_j}$, which allows, knowing $\delta_j^{(n+1)}$, to compute $\delta_j^{(n)}$. For the output layer $\delta_e^{(n)}=\left(y_e^{(n)}-d_e\right)\cdot\frac{dy_e}{dS_e}$. Then, the weight adjustment will have the form $\Delta\varpi_{ij}^{(n)}=-\eta\cdot\delta_j^{(n)}y_i^{(n-1)}$, $n=1,2$ [5, 8].

To give the weight correction process some inertia, in order to smooth out abrupt jumps when moving across the surface of the objective function, the last expression is supplemented with the values of the weight changes from the previous iteration

$$\Delta\varpi_{ij}^{(n)}=-\eta\cdot,\left(\mu\cdot\Delta\omega_j^{(n)}\cdot(t-1)+(1-\mu)\cdot\delta_j^{(n)}y_i^{(n-1)}\right),\ n=1,2,$$

where μ – the inertia coefficient, *t* – the current iteration number.

For the sigmoid function $\delta_e^{(n)}=\left(y_e^{(n)}-d_e\right)\cdot\left(1-S_e\right)\cdot S_e$, and for the hyperbolic tangent function $\delta_e^{(n)}=\left(y_e^{(n)}-d_e\right)\cdot\left(1-S_e^{2}\right)\cdot S_e$ [5, 8].

Let's evaluate the effectiveness of the proposed parameter evaluation methodology for the decision support system in comparison with the known approaches for evaluating such systems. The results of the evaluation based on the reliability of the decisions made are presented in **Table 3.2**.

Table 3.2 Evaluation of the effectiveness of the proposed methodology for parameter evaluation in decision support systems

Approach name	Evaluation completeness	Accuracy	Sensitivity	Average value
Densenet 201	0.6163	0.4243	0.4485	0.4335
Densenet 121	0.9523	0.8489	0.8590	0.8588
MobileNetV2	0.9289	0.9295	0.9289	0.9287
DenseNet-SEGR	0.9588	0.9514	0.9511	0.9512
Gradient boosting classifier	0.92021	0.91128	0.9003	0.91449
KNN	0.8736	0.8839	0.88529	0.9003
LSTM	0.7981	0.8005	0.8191	0.8217
RNN	0.8014	0.8122	0.8022	0.8101
CNN	0.9232	0.9104	0.9271	0.9301
Proposed method	0.9511	0.9611	0.9601	0.9612

As seen from **Table 3.2**, the reliability of the parameter evaluation in decision support systems is improved by 17–21% due to the use of additional procedures, while maintaining the required operational speed.

3.4 METHOD OF MULTI-CRITERIA EVALUATION OF HIERARCHICAL SYSTEMS

The method of multi-criteria evaluation of hierarchical systems consists of the following sequence of actions:

Action 1. Input of initial data.

At this stage, the available initial data for the beginning of the multi-criteria evaluation method are entered. The following information is introduced at this stage:

– the number of subsystems in the hierarchical system;

– characteristics of each subsystem in the hierarchical system (the number of elements in each subsystem, the number of connections between each element in the subsystem, the type of element in the subsystem (purpose, main technical characteristics), etc.);

– the number of connections between each subsystem (or a specific element) in the hierarchical system;

– the type and number of individual elements of the hierarchical system that are not part of any subsystem of the hierarchical system.

Action 2. Verification of the entered data and clarification of the relationships between the elements of the hierarchical system.

To reduce the subjectivity of the obtained evaluation, at this stage, the entered data is verified, and the relationships between the elements of the hierarchical system are clarified and described using the enhanced penguin swarm algorithm proposed in work [20].

Action 3. Description of external and internal factors affecting the hierarchical system being analyzed.

At this stage, the list of external factors affecting the functioning process of the hierarchical system is defined, along with their degree of influence on the functioning process of the hierarchical system. Internal factors present within the system are also introduced. This procedure is based on the evaluation method proposed in study [19], which uses the mathematical framework of fuzzy cognitive models.

Action 4. Verification and clarification of the established factors.

The procedure for verifying and clarifying the established factors consists of two stages. In the first stage, it involves using failure tree analysis and the interval-valued fuzzy Pythagorean hierarchical process to rank and select the most critical factors. In the second stage, it involves using the interval-valued fuzzy Pythagorean method for evaluating and visualizing the cause-and-effect relationships between the selected factors.

Action 4.1. Reduction of uncertainty using the interval-valued pythagorean fuzzy set.

In this study, a combination of fuzzy set theory (in this case, the Pythagorean fuzzy set) with multi-criteria evaluation methods is proposed to structure and solve complex decision-making tasks that involve broad and hierarchically organized criteria. This combination is widely used to overcome the inaccuracies that arise when relying on expert evaluation in multi-criteria evaluation methods.

The Pythagorean fuzzy set is defined as follows

$$P=\left\{\left\langle x, P\left(\mu_p(x), \nu_p(x)\right)\right\rangle; x\in X\right\}, \tag{3.10}$$

where X – a finite set, $\mu_p(x): X\mapsto\left[0,1\right]$ and $\nu_p(x): X\mapsto\left[0,1\right]$ – the degree of membership and degree of non-membership of an element $x\in X$ to the set P. The values of $\mu_p(x)$ and $\nu_p(x)$ must satisfy the following conditions

$$0\le\mu_p(x)^2+\nu_p(x)^2\le 1. x\in X. \tag{3.11}$$

The degree of uncertainty of the Pythagorean fuzzy set with respect to the set P can be calculated as follows

$$\pi_p(x)=\sqrt{1-\mu_p(x)^2-\nu_p(x)^2}. \tag{3.12}$$

For a more precise representation of variation and uncertainty, the interval-valued Pythagorean fuzzy set is used, in which intervals are employed to represent the degree of membership instead of point values. The set $\tilde{P}$ is defined as follows

$$\tilde{P}=\left\{\left\langle x, \tilde{P}\left(\left[\mu_{\tilde{P}_L}(x), \mu_{\tilde{P}_U}(x)\right], \left[\nu_{\tilde{P}_L}(x), \nu_{\tilde{P}_U}(x)\right]\right)\right\rangle; x\in X\right\}, \tag{3.13}$$

where $\left\langle\left[\mu_{\tilde{P}_L}(x), \mu_{\tilde{P}_U}(x)\right], \left[\nu_{\tilde{P}_L}(x), \nu_{\tilde{P}_U}(x)\right]\right\rangle$ – the interval-valued Pythagorean fuzzy number $0\le\mu_{\tilde{P}_L}(x)\le$ $\le\mu_{\tilde{P}_U}(x)\le\nu_{\tilde{P}_L}(x)\le\nu_{\tilde{P}_U}(x)$. $\mu_{\tilde{P}_L}(x)$ and $\nu_{\tilde{P}_L}(x)$ must satisfy the expression

$$0\le\mu_{\tilde{P}_L}(x)^2+\nu_{\tilde{P}_L}(x)^2\le 1. x\in X. \tag{3.14}$$

The value of uncertainty for the interval-valued Pythagorean fuzzy set with respect to $\tilde{P}$ can be calculated as follows

$$\pi_{\tilde{P}}(x)=\left[\pi_{\tilde{P}_L}(x),\pi_{\tilde{P}_U}(x)\right]=\left[\sqrt{1-\mu_{\tilde{P}_U}(x)^2-\nu_{\tilde{P}_U}(x)^2},\sqrt{1-\mu_{\tilde{P}_L}(x)^2-\nu_{\tilde{P}_L}(x)^2}\right]. \quad (3.15)$$

Action 4.2. Evaluation and visualization of cause-and-effect relationships between selected factors.

Given that one interval Pythagorean fuzzy set is provided to describe the cause-and-effect relationships between the selected factors, there is the following expression

$\tilde{P}=\left(\left[\mu_{\tilde{P}_L}(x),\mu_{\tilde{P}_U}(x)\right],\left[\nu_{\tilde{P}_L}(x),\nu_{\tilde{P}_U}(x)\right]\right),\left[\mu_{\tilde{P}_L}(x),\mu_{\tilde{P}_U}(x)\right]\subseteq[0,1],\left[\nu_{\tilde{P}_L}(x),\nu_{\tilde{P}_U}(x)\right]\subseteq[0,1],$ and

$0\le\mu_{\tilde{P}_L}(x)^2+\nu_{\tilde{P}_L}(x)^2\le 1$, with parameter $\lambda>0$. In this case, the following operation is performed:

$$\lambda\tilde{P}=\left(\left[\sqrt{1-\left(1-\mu_{\tilde{P}_L}(x)^2\right)^{\lambda}},\sqrt{1-\left(1-\mu_{\tilde{P}_U}(x)^2\right)^{\lambda}},\left[\nu_{\tilde{P}_L}(x)^{\lambda},\nu_{\tilde{P}_U}(x)^{\lambda}\right]\right]\right), \quad (3.16)$$

$$\lambda\tilde{P}=\left(\left[\mu_{\tilde{P}_L}(x)^{\lambda},\mu_{\tilde{P}_U}(x)^{\lambda}\right],\left[\sqrt{1-\left(1-\nu_{\tilde{P}_L}(x)^2\right)^{\lambda}},\sqrt{1-\left(1-\nu_{\tilde{P}_U}(x)^2\right)^{\lambda}}\right]\right), \quad (3.17)$$

$\tilde{P}=\left(\left[\mu_{\tilde{P}_L}(x),\mu_{\tilde{P}_U}(x)\right],\left[\nu_{\tilde{P}_L}(x),\nu_{\tilde{P}_U}(x)\right]\right),\left[\mu_{\tilde{P}_L}(x),\mu_{\tilde{P}_U}(x)\right]\subseteq[0,1],\left[\nu_{\tilde{P}_L}(x),\nu_{\tilde{P}_U}(x)\right]\subseteq[0,1],$ and
$0\le\mu_{\tilde{P}_L}(x)^2+\nu_{\tilde{P}_L}(x)^2\le 1$.

Given that two interval-valued Pythagorean fuzzy sets $\tilde{P}_1=\left(\left[a_1,b_1\right],\left[c_1,d_1\right]\right)$ and $\tilde{P}_2=\left(\left[a_2,b_2\right],\left[c_2,d_2\right]\right)$ are provided, the following operation is performed:

$$\tilde{P}_1\oplus\tilde{P}_2=\left(\left[\sqrt{a_1^2+a_2^2-a_1^2a_2^2,b_1^2+b_2^2-b_1^2b_2^2}\right],\left[c_1c_2,d_1d_2\right]\right), \quad (3.18)$$

$$\tilde{P}_1\oplus\tilde{P}_2=\left(\left[a_1a_2,b_1b_2\right],\left[\sqrt{c_1^2+c_2^2-c_1^2c_2^2,d_1^2+d_2^2-d_1^2d_2^2}\right]\right). \quad (3.19)$$

Action 5. Vulnerability analysis of the subsystem (individual element) of the hierarchical system.

Fault tree analysis is widely used to identify potential root causes, referred to as basic events, as well as to determine the probability of an unexpected event, known as the top event. The top event is placed at the top of the tree, while the basic events are at the bottom. Basic events (BE) within the fault tree are considered statistically independent and are combined using logical operators (AND/OR).

The fault tree analysis includes both qualitative and quantitative assessments. In the qualitative evaluation, the fault tree establishes and explains the theoretical relationships between the fault tree and basic events based on "AND" and "OR" logic. In the quantitative assessment, the basic events and their logical relationships are identified to construct the logical expression of the fault tree. The probability of the top event can be calculated quantitatively based on the probabilities of each risk factor. In this study, the fault tree is used to analyze cause-and-effect relationships between the identified factors, as well as to rank them by the probability of occurrence. The probability of the top event is evaluated using equations (3.20)–(3.22), which are derived from the principles of Boolean algebra:

$$P_{OR} = 1 - \prod_{i=1}^{n} (1 - P_i), \tag{3.20}$$

$$P_2 = \prod_{i=1}^{n} P_i, \tag{3.21}$$

$$P_{IE} = \prod_{j \in M} \left(1 - \prod_{BE_i \in Q_j}^{n} (1 - P_i) \right), \tag{3.22}$$

where P_i – the probability of occurrence of the basic event BE_i; Q_j – the set of basic events BE_i.

To assess the importance of each basic event, its contribution to the probability of the top event is determined. This information is highly valuable for decision-makers, as it allows identifying the most vulnerable points in the system. By doing so, decision support systems can effectively identify the factors most likely to lead to failures and require increased attention.

To identify and prioritize the most critical basic events leading to the top event, the Birnbaum importance measure is used. The Birnbaum importance measure is a key metric based on fault tree analysis and is used to assess the criticality of individual components or events in the system. It quantitatively evaluates the contribution of each basic event to the occurrence of the top event. Formally, the value of the Birnbaum importance measure for a specific basic event is defined as follows

$$IM_{BE_i}^{BIM} = P\left(IE \mid BR_i = 1\right) - P\left(IE \mid BR_i = 0\right), \tag{3.23}$$

where $IM_{BE_i}^{BIM}$ – the Birnbaum importance measure for the basic event BR_i.

Once the Birnbaum Importance Measure (BIM) values for all basic events are computed, they can be sorted according to their level of importance. A higher BIM value indicates a higher level of significance of the corresponding basic event with respect to the occurrence of the top event.

Action 6. Ranking of impact factors on the hierarchical system.

The use of fault tree analysis methods and interval-valued Pythagorean fuzzy sets ensures two types of weights: relative importance and corresponding ranking. To provide a balanced assessment that considers both the impact of the vulnerabilities and the likelihood of their occurrence, a corrective weight is introduced. This weight is used to reconcile both indicators, forming an updated ranking of the factors. Based on this new ranking, the most important factors are selected for further analysis. The updated ranking is calculated using the following formula

$$MR_i = w \cdot R_i^1 + (1 - w) \cdot R_i^2, \tag{3.24}$$

where MR_i – the combined ranking value for a factor i, R_i^1 – the ranking of factor i, obtained from the results of the fault tree analysis, R_i^2 – the rank of factor i, obtained from the results of the interval-valued

Pythagorean fuzzy sets, *w* – the corrective weight that defines the impact of each aspect. After this, the factors that cause failures can be re-sorted based on the combined ranking.

To determine the effectiveness of the proposed method, a simulation of its operation was conducted to solve the multi-criteria evaluation task of the state of the military grouping (forces) under the initial conditions specified in **Section 3.4.**

Separate parts of the computational experiment, using the proposed method, are presented in **Tables 3.3** and **3.4**. The overall computational experiment is detailed across more than 140 pages, with only a specific part of it presented in this section.

Table 3.3 Results of the calculation of membership functions for decisions based on rules

№	Results of the calculation of membership functions for decisions based on rules											
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
1	0.0007	0.045	0.048	0.04	0.066	0.032	0.007	0.005	0.009	0.049	0.063	0.044
2	0.061	0.039	0.116	0	0.126	0.158	0.147	0.018	0.072	0.137	0.162	0.163
3	0.065	0.041	0.05	0.027	0.011	0.058	0.033	0.04	0.045	0.056	0.067	0.046
4	0.095	0.074	0.153	0.068	0.004	0.1	0.0018	0.169	0.0052	0.053	0.046	0.163
5	0.174	0.0147	0.083	0.083	0.076	0.002	0.102	0.083	0.162	0.116	0.09	0.105
6	0.028	0.057	0.019	0.036	0.047	0.038	0.025	0.028	0.0029	0.005	0.036	0.063
7	0.061	0.067	0.056	0.045	0.012	0.014	0.0007	0.012	0.022	0.056	0.069	0.00216
8	0.197	0.219	0.211	0.232	0.197	0.203	0.057	0.07	0.119	0.13	0.138	0.0054
9	0	0.122	0.124	0.157	0.243	0.003	0.262	0.208	0	0.165	0.084	0.151
10	0.146	0.079	0.142	0.076	0.005	0.121	0.107	0.121	0.114	0.091	0.049	0.139
11	0.165	0.139	0.065	0.044	0.07	0.1	0.083	0.163	0.061	0.165	0.133	0.086
12	0.026	0.039	0.001	0.006	0.043	0.021	0.036	0.013	0.014	0.034	0.02	0.03
13	0.035	0.006	0.037	0.04	0.021	0.038	0.004	0.0005	0.033	0.017	0.021	0.017
14	0.0054	0.003	0.033	0.021	0.007	0.028	0.029	0.0076	0.05	0.033	0.017	0.038
15	0.049	0.009	0.012	0.021	0.033	0.03	0.044	0.023	0.024	0.034	0.018	0.041
16	0.03	0.042	0.027	0.019	0.014	0.047	0.029	0.011	0.036	0.023	0.05	0.033
17	0.021	0.0005	0.031	0.028	0.032	0.047	0.031	0.02	0.024	0.012	0.02	0.032
18	0.03	0.008	0.016	0.044	0.02	0.036	0.016	0.048	0.05	0.014	0.035	0.0086
19	0.026	0.039	0.038	0.014	0.003	0.002	0.031	0.011	0.031	0.0076	0.034	0.013
20	0.007	0.046	0.049	0.033	0.015	0.007	0.049	0.023	0.05	0.016	0.03	0.034
21	0.042	0.026	0.026	0.025	0.037	0.029	0.027	0.021	0.015	0.01	0.041	0.00758

Continuation of Table 3.3

	1	2	3	4	5	6	7	8	9	10	11	12
22	0.126	0.027	0.017	0.315	0.033	0.096	0.206	0.305	0.093	0.146	0.116	0.00332
23	0.391	0.462	0.616	0.443	0.077	0.231	0.0064	0.077	0.616	0.109	0.237	0.61
24	0.132	0.005	0.04	0.002	0.035	0.139	0.063	0.0088	0.112	0.118	0.109	0.037
25	0.14	0.125	0.044	0.139	0.13	0.074	0.107	0.125	0.1	0.054	0.021	0.158
26	0.041	0.047	0.02	0.026	0.008	0.016	0.025	0.019	0.043	0.031	0.04	0.049
27	0.022	0.014	0.041	0.037	0.034	0.046	0.013	0.027	0.022	0.011	0.042	0.012
28	0.038	0.008	0.015	0.011	0.018	0	0.017	0.033	0.018	0.042	0.043	0.023
29	0.037	0	0.039	0.015	0.035	0.004	0.021	0.017	0.039	0.031	0.004	0.05
30	0.007	0.028	0.011	0.031	0.012	0.048	0.021	0.026	0.032	0.036	0.033	0.026
31	0.032	0.011	0.007	0.018	0.033	0.036	0.04	0.011	0.038	0.024	0.018	0.045
32	0.041	0.02	0.05	0.027	0.008	0.017	0.05	0.024	0.031	0.045	0.034	0.022
33	0.022	0.019	0.039	0.049	0.043	0.000	0.045	0.029	0.0025	0.016	0.013	0.037
34	0.042	0.048	0.011	0.02	0.013	0.042	0.006	0.0035	0.014	0.0056	0.049	0.049
35	0.05	0.032	0.032	0.037	0.027	0.014	0.005	0.046	0.038	0.02	0.037	0.039
36	0.081	0.044	0.049	0.102	0.016	0.146	0.053	0.114	0.133	0.054	0.054	0.086
37	0.139	0.153	0.025	0.172	0.014	0.142	0.025	0.114	0.063	0.04	0.091	0.135
38	0.019	0.044	0.012	0.004	0.03	0.047	0.008	0.024	0.05	0.033	0.008	0.0015
39	0.023	0.034	0.041	0.003	0.015	0.015	0.05	0.048	0.018	0.036	0.035	0.027
40	0.034	0.063	0.056	0.023	0.085	0.045	0.025	0.0073	0.012	0.113	0.078	0.036
41	0.045	0.016	0.023	0.027	0.032	0.006	0.027	0.011	0.036	0.045	0.038	0.041
42	0.018	0.013	0.019	0.038	0.05	0.021	0.023	0.03	0.028	0.024	0.015	0.045
43	0.0005	0.031	0.033	0.028	0.047	0.023	0.0005	0.035	0.0066	0.034	0.044	0.031
Defense	0.174	0.147	0.153	0.083	0.126	0.158	0.147	0.169	0.162	0.137	0.162	0.163
Counteroffensive	0.391	0.462	0.616	0.443	0.243	0.231	0.262	0.305	0.616	0.165	0.237	0.61
Stabilization actions	0.139	0.153	0.056	0.172	0.14	0.142	0.05	0.114	0.063	0.113	0.091	0.135
Error	0.42	0.334	0.174	0.347	0.609	0.636	0.569	0.525	0.178	0.729	0.617	0.197

Table 3.4 Comparative results of the state evaluation process for the troop (force) grouping

	Using the method	Without using the method
Operational efficiency of the evaluation process		
Best case, sec.	49 – 303	56 – 507.1
Worst case, sec.	255.1 – 2501.5	382.8 – 3977
Reliability of the obtained decisions		
Best case, sec.	0.89 – 1.0	0.64 – 0.85
Worst case, sec.	0.77 – 1.0	0.617 – 0.75

From the analysis of **Table 3.4**, it can be concluded that the proposed method provides an average increase in accuracy and operational efficiency by 35%, while ensuring high convergence of the obtained results at a level of 93.17%.

CONCLUSIONS

The algorithm for implementing the methodology has been defined, thanks to additional and improved procedures, which allow:

– verification of the topology and parameters of decision support systems, taking into account the degree of uncertainty in the initial data regarding the known information, through the use of the enhanced penguin swarm algorithm. This allows reducing the time for initial setup during the first configuration of the evaluation methodology;

– preliminary selection of individuals for configuring the evolving artificial neural network, carried out using the enhanced genetic algorithm, which reduces solution search time and increases the reliability of the obtained solutions;

– exploration of solution spaces for parameter evaluation problems in decision support systems, described by atypical functions, using the enhanced penguin swarm algorithm;

– configuration of the weights of the evolving artificial neural network, leading to increased accuracy in evaluating decision support system parameters;

– use of additional mechanisms to adjust the parameters of the evolving artificial neural network through the procedure of modifying the membership function;

– increased reliability in evaluating decision support system parameters by parallel evaluation using multiple evaluation methods;

– use of hybrid parameter evaluation for decision support systems, enabling proper operation in the absence of conditions for stationarity, homogeneity, normality, and independence;

– simultaneous search for solutions in different directions;

– calculation of the required number of computational resources needed when existing resources are insufficient for calculations.

An example of using the proposed methodology for evaluating decision support system parameters showed an increase in the reliability of parameter evaluation by 17–21% through the use of additional procedures, while maintaining the required operational speed.

The algorithm for implementing the method has been determined, thanks to additional and improved procedures, which allow:

– verification of the input data and clarification of relationships between elements in the hierarchical system using the enhanced penguin swarm algorithm. This minimizes the error of entering incorrect data for the operational grouping of troops (forces);

– description of the external and internal factors affecting the hierarchical system, which is subject to multi-criteria evaluation using fuzzy cognitive models;

– adaptation to the hierarchical system type through multi-level adaptation of the indicator system and evaluation criteria;

– reduction of uncertainty using interval Pythagorean fuzzy sets, which improves the reliability of multi-criteria evaluation of the state of hierarchical systems;

– identification of the most vulnerable elements of the hierarchical system using a fault tree;

– adaptation of the membership function type based on the available computational resources, ensuring adaptation to the available computational resources.

An example of using the proposed method for multi-criteria evaluation of the operational grouping of troops (forces) was provided, showing that the proposed method ensures an average increase in accuracy and operational speed by 35%, while ensuring high convergence of the results at a level of 93.17%.

CONFLICT OF INTEREST

The authors declare that they have no conflict of interest in relation to this research, whether financial, personal, authorship or otherwise, that could affect the research and its results presented in this paper.

USE OF ARTIFICIAL INTELLIGENCE

The authors confirm that they did not use artificial intelligence technologies in creating the submitted work.

REFERENCES

1. Dudnyk, V., Sinenko, Y., Matsyk, M., Demchenko, Y., Zhyvotovskyi, R., Repilo, I. et al. (2020). Development of a method for training artificial neural networks for intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 3 (2 (105)), 37–47. https://doi.org/10.15587/1729-4061.2020.203301

2. Sova, O., Shyshatskyi, A., Salnikova, O., Zhuk, O., Trotsko, O., Hrokholskyi, Y. (2021). Development of a method for assessment and forecasting of the radio electronic environment. EUREKA: Physics and Engineering, 4, 30–40. https://doi.org/10.21303/2461-4262.2021.001940
3. Pievtsov, H., Turinskyi, O., Zhyvotovskyi, R., Sova, O., Zvieriev, O., Lanetskii, B., Shyshatskyi, A. (2020). Development of an advanced method of finding solutions for neuro-fuzzy expert systems of analysis of the radioelectronic situation. EUREKA: Physics and Engineering, 4, 78–89. https://doi.org/10.21303/2461-4262.2020.001353
4. Zuiev, P., Zhyvotovskyi, R., Zvieriev, O., Hatsenko, S., Kuprii, V., Nakonechnyi, O. et al. (2020). Development of complex methodology of processing heterogeneous data in intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 4 (9 (106)), 14–23. https://doi.org/10.15587/1729-4061.2020.208554
5. Kuchuk, N., Mohammed, A. S., Shyshatskyi, A., Nalapko, O. (2019). The Method of Improving the Efficiency of Routes Selection in Networks of Connection with the Possibility of Self-Organization. International Journal of Advanced Trends in Computer Science and Engineering, 8 (1.2), 1–6.
6. Shyshatskyi, A., Zvieriev, O., Salnikova, O., Demchenko, Ye., Trotsko, O., Neroznak, Ye. (2020). Complex Methods of Processing Different Data in Intellectual Systems for Decision Support System. International Journal of Advanced Trends in Computer Science and Engineering, 9 (4), 5583–5590. https://doi.org/10.30534/ijatcse/2020/206942020
7. Pozna, C., Precup, R.-E., Horvath, E., Petriu, E. M. (2022). Hybrid Particle Filter–Particle Swarm Optimization Algorithm and Application to Fuzzy Controlled Servo Systems. IEEE Transactions on Fuzzy Systems, 30 (10), 4286–4297. https://doi.org/10.1109/tfuzz.2022.3146986
8. Yang, X.-S., Deb, S. (2013). Cuckoo search: recent advances and applications. Neural Computing and Applications, 24 (1), 169–174. https://doi.org/10.1007/s00521-013-1367-1
9. Mirjalili, S. (2015). The Ant Lion Optimizer. Advances in Engineering Software, 83, 80–98. https://doi.org/10.1016/j.advengsoft.2015.01.010
10. Yu, J. J. Q., Li, V. O. K. (2015). A social spider algorithm for global optimization. Applied Soft Computing, 30, 614–627. https://doi.org/10.1016/j.asoc.2015.02.014
11. Mirjalili, S., Mirjalili, S. M., Lewis, A. (2014). Grey Wolf Optimizer. Advances in Engineering Software, 69, 46–61. https://doi.org/10.1016/j.advengsoft.2013.12.007
12. Koval, V., Nechyporuk, O., Shyshatskyi, A., Nalapko, O., Shknai, O., Zhyvylo, Y. et al. (2023). Improvement of the optimization method based on the cat pack algorithm. Eastern-European Journal of Enterprise Technologies, 1 (9 (121)), 41–48. https://doi.org/10.15587/1729-4061.2023.273786
13. Gupta, E., Saxena, A. (2015). Robust generation control strategy based on Grey Wolf Optimizer. Journal of Electrical Systems, 11, 174–188.
14. Chaman-Motlagh, A. (2015). Superdefect Photonic Crystal Filter Optimization Using Grey Wolf Optimizer. IEEE Photonics Technology Letters, 27 (22), 2355–2358. https://doi.org/10.1109/lpt.2015.2464332
15. Nuaekaew, K., Artrit, P., Pholdee, N., Bureerat, S. (2017). Optimal reactive power dispatch problem using a two-archive multi-objective grey wolf optimizer. Expert Systems with Applications, 87, 79–89. https://doi.org/10.1016/j.eswa.2017.06.009

16. Koval, M., Sova, O., Orlov, O., Shyshatskyi, A., Artabaiev, Y., Shknai, O. et al. (2022). Improvement of complex resource management of special-purpose communication systems. Eastern-European Journal of Enterprise Technologies, 5 (9 (119)), 34–44. https://doi.org/10.15587/1729-4061.2022.266009
17. Ali, M., El-Hameed, M. A., Farahat, M. A. (2017). Effective parameters' identification for polymer electrolyte membrane fuel cell models using grey wolf optimizer. Renewable Energy, 111, 455–462. https://doi.org/10.1016/j.renene.2017.04.036
18. Zhang, S., Zhou, Y. (2017). Template matching using grey wolf optimizer with lateral inhibition. Optik, 130, 1229–1243. https://doi.org/10.1016/j.ijleo.2016.11.173
19. Khouni, S. E., Menacer, T. (2023). Nizar optimization algorithm: a novel metaheuristic algorithm for global optimization and engineering applications. The Journal of Supercomputing, 80 (3), 3229–3281. https://doi.org/10.1007/s11227-023-05579-4
20. Saremi, S., Mirjalili, S., Lewis, A. (2017). Grasshopper Optimisation Algorithm: Theory and application. Advances in Engineering Software, 105, 30–47. https://doi.org/10.1016/j.advengsoft.2017.01.004
21. Braik, M. S. (2021). Chameleon Swarm Algorithm: A bio-inspired optimizer for solving engineering design problems. Expert Systems with Applications, 174, 114685. https://doi.org/10.1016/j.eswa.2021.114685
22. Thamer, K. A., Sova, O., Shaposhnikova, O., Yashchenok, V., Stanovska, I., Shostak, S. et al. (2024). Development of a solution search method using a combined bio-inspired algorithm. Eastern-European Journal of Enterprise Technologies, 1 (4 (127)), 6–13. https://doi.org/10.15587/1729-4061.2024.298205
23. Yapici, H., Cetinkaya, N. (2019). A new meta-heuristic optimizer: Pathfinder algorithm. Applied Soft Computing, 78, 545–568. https://doi.org/10.1016/j.asoc.2019.03.012
24. Duan, H., Qiao, P. (2014). Pigeon-inspired optimization: a new swarm intelligence optimizer for air robot path planning. International Journal of Intelligent Computing and Cybernetics, 7 (1), 24–37. https://doi.org/10.1108/ijicc-02-2014-0005
25. Shyshatskyi, A., Romanov, O., Shknai, O., Babenko, V., Koshlan, O., Pluhina, T. et al. (2023). Development of a solution search method using the improved emperor penguin algorithm. Eastern-European Journal of Enterprise Technologies, 6 (4 (126)), 6–13. https://doi.org/10.15587/1729-4061.2023.291008
26. Yang, X. S (2012). Flower pollination algorithm for global optimization. Unconventional computing and natural computation. Berlin, Heidelberg: Springer, 240–249. https://doi.org/10.1007/978-3-642-32894-7_27
27. Gomes, G. F., da Cunha, S. S., Ancelotti, A. C. (2018). A sunflower optimization (SFO) algorithm applied to damage identification on laminated composite plates. Engineering with Computers, 35 (2), 619–626. https://doi.org/10.1007/s00366-018-0620-8
28. Mehrabian, A. R., Lucas, C. (2006). A novel numerical optimization algorithm inspired from weed colonization. Ecological Informatics, 1 (4), 355–366. https://doi.org/10.1016/j.ecoinf.2006.07.003
29. Qi, X., Zhu, Y., Chen, H., Zhang, D., Niu, B. (2013). An idea based on plant root growth for numerical optimization. Intelligent Computing Theories and Technology. Berlin, Heidelberg: Springer, 571–578. https://doi.org/10.1007/978-3-642-39482-9_66
30. Bezuhlyi, V., Oliynyk, V., Romanenko, I., Zhuk, O., Kuzavkov, V., Borysov, O., Korobchenko, S., Ostapchuk, E., Davydenko, T., Shyshatskyi, A. (2021). Development of object state estimation method in intelligent

decision support systems. Eastern-European Journal of Enterprise Technologies, 5 (3 (113)), 54–64. https://doi.org/10.15587/1729-4061.2021.239854

31. Mahdi, Q. A., Shyshatskyi, A., Prokopenko, Y., Ivakhnenko, T., Kupriyenko, D., Golian, V. et al. (2021). Development of estimation and forecasting method in intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 3 (9 (111)), 51–62. https://doi.org/10.15587/1729-4061.2021.232718
32. Sova, O., Radzivilov, H., Shyshatskyi, A., Shevchenko, D., Molodetskyi, B., Stryhun, V. et al. (2022). Development of the method of increasing the efficiency of information transfer in the special purpose networks. Eastern-European Journal of Enterprise Technologies, 3 (4 (117)), 6–14. https://doi.org/10.15587/1729-4061.2022.259727
33. Zhang, H., Zhu, Y., Chen, H. (2013). Root growth model: a novel approach to numerical function optimization and simulation of plant root system. Soft Computing, 18 (3), 521–537. https://doi.org/10.1007/s00500-013-1073-z
34. Labbi, Y., Attous, D. B., Gabbar, H. A., Mahdad, B., Zidan, A. (2016). A new rooted tree optimization algorithm for economic dispatch with valve-point effect. International Journal of Electrical Power & Energy Systems, 79, 298–311. https://doi.org/10.1016/j.ijepes.2016.01.028
35. Murase, H. (2000). Finite element inverse analysis using a photosynthetic algorithm. Computers and Electronics in Agriculture, 29 (1-2), 115–123. https://doi.org/10.1016/s0168-1699(00)00139-3
36. Zhao, S., Zhang, T., Ma, S., Chen, M. (2022). Dandelion Optimizer: A nature-inspired metaheuristic algorithm for engineering applications. Engineering Applications of Artificial Intelligence, 114, 105075. https://doi.org/10.1016/j.engappai.2022.105075
37. Paliwal, N., Srivastava, L., Pandit, M. (2020). Application of grey wolf optimization algorithm for load frequency control in multi-source single area power system. Evolutionary Intelligence, 15 (1), 563–584. https://doi.org/10.1007/s12065-020-00530-5
38. Dorigo, M., Blum, C. (2005). Ant colony optimization theory: A survey. Theoretical Computer Science, 344 (2-3), 243–278. https://doi.org/10.1016/j.tcs.2005.05.020
39. Poli, R., Kennedy, J., Blackwell, T. (2007). Particle swarm optimization. Swarm Intelligence, 1 (1), 33–57. https://doi.org/10.1007/s11721-007-0002-0
40. Bansal, J. C., Sharma, H., Jadon, S. S., Clerc, M. (2014). Spider Monkey Optimization algorithm for numerical optimization. Memetic Computing, 6 (1), 31–47. https://doi.org/10.1007/s12293-013-0128-0
41. Yeromina, N., Kurban, V., Mykus, S., Peredrii, O., Voloshchenko, O., Kosenko, V. et al. (2021). The Creation of the Database for Mobile Robots Navigation under the Conditions of Flexible Change of Flight Assignment. International Journal of Emerging Technology and Advanced Engineering, 11 (5), 37–44. https://doi.org/10.46338/ijetae0521_05
42. Maccarone, A. D., Brzorad, J. N., Stone H. M. (2008). Characteristics And Energetics Of Great Egret And Snowy Egret Foraging Flights. Waterbirds, 31 (4), 541–549. https://doi.org/10.1675/1524-4695-31.4.541
43. Ramaji, I. J., Memari, A. M. (2018). Interpretation of structural analytical models from the coordination view in building information models. Automation in Construction, 90, 117–133. https://doi.org/10.1016/j.autcon.2018.02.025

44. Pérez-González, C. J., Colebrook, M., Roda-García, J. L., Rosa-Remedios, C. B. (2019). Developing a data analytics platform to support decision making in emergency and security management. Expert Systems with Applications, 120, 167–184. https://doi.org/10.1016/j.eswa.2018.11.023
45. Chen, H. (2018). Evaluation of Personalized Service Level for Library Information Management Based on Fuzzy Analytic Hierarchy Process. Procedia Computer Science, 131, 952–958. https://doi.org/10.1016/j.procs.2018.04.233
46. Chan, H. K., Sun, X., Chung, S.-H. (2019). When should fuzzy analytic hierarchy process be used instead of analytic hierarchy process? Decision Support Systems, 125, 113114. https://doi.org/10.1016/j.dss.2019.113114
47. Osman, A. M. S. (2019). A novel big data analytics framework for smart cities. Future Generation Computer Systems, 91, 620–633. https://doi.org/10.1016/j.future.2018.06.046
48. Nechyporuk, O., Sova, O., Shyshatskyi, A., Kravchenko, S., Nalapko, O., Shknai, O. et al. (2023). Development of a method of complex analysis and multidimensional forecasting of the state of intelligence objects. Eastern-European Journal of Enterprise Technologies, 2 (4 (122)), 31–41. https://doi.org/10.15587/1729-4061.2023.276168
49. Merrikh-Bayat, F. (2015). The runner-root algorithm: A metaheuristic for solving unimodal and multimodal optimization problems inspired by runners and roots of plants in nature. Applied Soft Computing, 33, 292–303. https://doi.org/10.1016/j.asoc.2015.04.048
50. Poliarush, O., Krepych, S., Spivak, I. (2023). hybrid approach for data filtering and machine learning inside content management system. Advanced Information Systems, 7 (4), 70–74. https://doi.org/10.20998/2522-9052.2023.4.09
51. Balochian, S., Baloochian, H. (2019). Social mimic optimization algorithm and engineering applications. Expert Systems with Applications, 134, 178–191. https://doi.org/10.1016/j.eswa.2019.05.035
52. Lenord Melvix, J. S. M. (2014). Greedy Politics Optimization: Metaheuristic inspired by political strategies adopted during state assembly elections. 2014 IEEE International Advance Computing Conference (IACC), 1157–1162. https://doi.org/10.1109/iadcc.2014.6779490
53. Moosavian, N., Roodsari, B. K. (2014). Soccer League Competition Algorithm, a New Method for Solving Systems of Nonlinear Equations. International Journal of Intelligence Science, 4 (1), 7–16. https://doi.org/10.4236/ijis.2014.41002
54. Hayyolalam, V., Pourhaji Kazem, A. A. (2020). Black Widow Optimization Algorithm: A novel meta-heuristic approach for solving engineering optimization problems. Engineering Applications of Artificial Intelligence, 87, 103249. https://doi.org/10.1016/j.engappai.2019.103249
55. Abualigah, L., Yousri, D., Abd Elaziz, M., Ewees, A. A., Al-qaness, M. A. A., Gandomi, A. H. (2021). Aquila Optimizer: A novel meta-heuristic optimization algorithm. Computers & Industrial Engineering, 157, 107250. https://doi.org/10.1016/j.cie.2021.107250
56. Hodlevskyi, M., Burlakov, G. (2023). Information technology of quality improvement planning of process subsets of the spice model. Advanced Information Systems, 7 (4), 52–59. https://doi.org/10.20998/2522-9052.2023.4.06
57. Askari, Q., Younas, I., Saeed, M. (2020). Political Optimizer: A novel socio-inspired meta-heuristic for global optimization. Knowledge-Based Systems, 195, 105709. https://doi.org/10.1016/j.knosys.2020.105709

58. Mohamed, A. W., Hadi, A. A., Mohamed, A. K. (2019). Gaining-sharing knowledge based algorithm for solving optimization problems: a novel nature-inspired algorithm. International Journal of Machine Learning and Cybernetics, 11 (7), 1501–1529. https://doi.org/10.1007/s13042-019-01053-x
59. Gödri, I., Kardos, C., Pfeiffer, A., Váncza, J. (2019). Data analytics-based decision support workflow for high-mix low-volume production systems. CIRP Annals, 68 (1), 471–474. https://doi.org/10.1016/j.cirp.2019.04.001
60. Harding, J. L. (2013). Data quality in the integration and analysis of data from multiple sources: some research challenges. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XL-2/W1, 59–63. https://doi.org/10.5194/isprsarchives-xl-2-w1-59-2013
61. Orouskhani, M., Orouskhani, Y., Mansouri, M., Teshnehlab, M. (2013). A Novel Cat Swarm Optimization Algorithm for Unconstrained Optimization Problems. International Journal of Information Technology and Computer Science, 5 (11), 32–41. https://doi.org/10.5815/ijitcs.2013.11.04
62. Karaboga, D., Basturk, B. (2007). A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. Journal of Global Optimization, 39 (3), 459–471. https://doi.org/10.1007/s10898-007-9149-x
63. Fister, I., Fister, I., Yang, X.-S., Brest, J. (2013). A comprehensive review of firefly algorithms. Swarm and Evolutionary Computation, 13, 34–46. https://doi.org/10.1016/j.swevo.2013.06.001
64. Sova, O., Radzivilov, H., Shyshatskyi, A., Shvets, P., Tkachenko, V., Nevhad, S. et al. (2022). Development of a method to improve the reliability of assessing the condition of the monitoring object in special-purpose information systems. Eastern-European Journal of Enterprise Technologies, 2 (3 (116)), 6–14. https://doi.org/10.15587/1729-4061.2022.254122
65. Khudov, H., Khizhnyak, I., Glukhov, S., Shamrai, N., Pavlii, V. (2024). The method for objects detection on satellite imagery based on the firefly algorithm. Advanced Information Systems, 8 (1), 5–11. https://doi.org/10.20998/2522-9052.2024.1.01
66. Owaid, S. R., Zhuravskyi, Y., Lytvynenko, O., Veretnov, A., Sokolovskyi, D., Plekhova, G. et al. (2024). Development of a method of increasing the efficiency of decision-making in organizational and technical systems. Eastern-European Journal of Enterprise Technologies, 1 (4 (127)), 14–22. https://doi.org/10.15587/1729-4061.2024.298568
67. Tyurin, V., Bieliakov, R., Odarushchenko, E., Yashchenok, V., Shaposhnikova, O., Lyashenko, A. et al. (2023). Development of a solution search method using an improved locust swarm algorithm. Eastern-European Journal of Enterprise Technologies, 5 (4 (125)), 25–33. https://doi.org/10.15587/1729-4061.2023.287316
68. Yakymiak, S., Vdovytskyi, Y., Artabaiev, Y., Degtyareva, L., Vakulenko, Y., Nevhad, S. et al. (2023). Development of the solution search method using the population algorithm of global search optimization. Eastern-European Journal of Enterprise Technologies, 3 (4 (123)), 39–46. https://doi.org/10.15587/1729-4061.2023.281007
69. Mohammed, B. A., Zhuk, O., Vozniak, R., Borysov, I., Petrozhalko, V., Davydov, I. et al. (2023). Improvement of the solution search method based on the cuckoo algorithm. Eastern-European Journal of Enterprise Technologies, 2 (4 (122)), 23–30. https://doi.org/10.15587/1729-4061.2023.277608

70. Raskin, L., Sukhomlyn, L., Sokolov, D., Vlasenko, V. (2023). Multi-criteria evaluation of the multi-factor stochastic systems effectiveness. Advanced Information Systems, 7 (2), 63–67. https://doi.org/10.20998/2522-9052.2023.2.09
71. Arora, S., Singh, S. (2018). Butterfly optimization algorithm: a novel approach for global optimization. Soft Computing, 23 (3), 715–734. https://doi.org/10.1007/s00500-018-3102-4

DOI: 10.15587/978-617-8360-23-8.CH4

4 DEVELOPMENT OF A POLYMODEL RESOURCE MANAGEMENT COMPLEX FOR INTELLIGENT DECISION SUPPORT SYSTEMS

Andrii Shyshatskyi, Ganna Plekhova, Danylo Pliekhov, Oleksii Nalapko, Oksana Dmytriieva, Ivan Starynskyi

ABSTRACT

Intelligent decision support systems are the object of research. The problem that is solved in the research is to increase the accuracy of modeling the process of functioning of intelligent decision support systems. The development of a polymodel resource management complex of intelligent decision support systems was carried out. The originality of the research is:

– in a comprehensive description of the process of functioning of intelligent decision support systems;

– the ability to simulate both a single process that takes place in intelligent decision support systems, and to comprehensively simulate those processes that take place in them;

– in establishing the conceptual dependencies of the process of functioning of intelligent decision support systems. This allows to describe the interaction of individual models at all stages of solving calculation tasks;

– descriptions of coordination processes in hybrid intelligent decision support systems, which increase the reliability of management decision-making;

– modeling of processes for solving complex calculation tasks in intelligent decision support systems, due to the conceptual description of the specified process;

– coordination of calculation processes in intelligent decision support systems, which achieves a decrease in the number of computing resources of systems;

– comprehensive dispute resolution, due to a complex of appropriate mathematical models.

The proposed polymodel complex should be used to solve the task of managing intelligent decision support systems characterized by a high degree of complexity.

KEYWORDS

Intelligent decision support systems, complex modeling, efficiency, reliability, coordination.

Intelligent decision support systems (IDSS) are an integral component of all spheres of human social activity and are used to solve a wide range of tasks, from entertainment to highly specific ones [1–3].

The main tasks to be solved by IDSS are [3–5]:

– solving various computing tasks in the interests of a wide range of consumers, regardless of their field of application;

– storage of calculation results and also intermediate results for user needs;

– support decision-making by the persons who make them;

– provide prerequisites for intelligent decision-making.

Trends in the development of modern IDSS are aimed at solving the following tasks [4–8]:

– increasing the efficiency of processing various types of data and their reliability;

– increasing the accuracy of modeling the process of their functioning;

– maintaining a balance between efficiency and reliability of the process of solving calculation tasks, etc.

At the same time, available scientific approaches to the synthesis and functioning of IDSS have insufficient accuracy and convergence. Said related to the following reasons [1–9]:

– the essential role of the human factor in the process of primary adjustment of IDSS;

– a large number of heterogeneous sources of information, which are subject to analysis and further processing during the functioning of the IDSS;

– IDSSs function under conditions of uncertainty, which causes a delay in their processing;

– the presence of a large number of destabilizing factors affecting the functioning of IDSS, etc.

This prompts the implementation of various strategies to enhance the efficiency of the IDSS in solving calculation tasks.

One of these options is the improvement of existing (development of new) mathematical models of the functioning of intelligent decision support systems.

The analysis of works [9–62] showed that the common shortcomings of the above-mentioned studies are:

– modeling of each approach is carried out only at a separate level of IDSS functioning;

– with a complex approach, as a rule, two components of the functioning of the IDSS are considered. This does not allow for a full assessment of the impact of management decisions on their further functioning;

– the models listed above, constituting the constituent parts of the above approaches, provide weak integration into each other, which prevents them from being combined to function together;

– the above models use a different mathematical apparatus, which does not require appropriate mathematical transformations, which in turn increases computational complexity and reduces the accuracy of modeling, etc.

Intelligent decision support systems are the object of research. The problem that is solved in the research is to increase the accuracy of modeling the process of functioning of intelligent decision support systems.

The subject of the study is the process of functioning of intelligent decision support systems using a set of mathematical models of their functioning.

The hypothesis of the study is the possibility of increasing the efficiency and accuracy of the functioning of intelligent decision support systems due to the development of a set of models of their functioning.

Modeling of the proposed method was carried out in the Microsoft Visual Studio 2022 software environment (USA). The hardware of the research process is AMD Ryzen 5.

Table 4.1 shows the composition of the heterogeneous model field and the methods of presenting the models.

Table 4.1 Models included in the heterogeneous model field

Model	Class of model and its characteristics	Implementation
Artificial neural networks	Artificial neural network (ANN) (search for hidden dependencies in statistical data and prediction of plan execution) – functional element. ANN with an evolving structure. Neuron transfer function: sigmoid. The number of inputs during experiments varied from 3 to 30, and the number of outputs from 1 to 10. The number of hidden neurons ranged from 1 to 8. Neuron transfer function: sigmoid. ANN training method – as in [2]. Average training error – 9%. Training sequence – 60 test tasks	Author's algorithm written in Microsoft Visual Studio 2022. Total code volume: 250 lines
Improved genetic algorithm (GA)	Improved GA [19] for solving an optimization problem – functional element. Population of 100 chromosomes. Evolution: crossover and mutation. Selection: combination of panmixia and ranking. Fitness (in %) – when fitness is below 50%, half of the population is eliminated and regenerated. If for ten generations fitness does not change but exceeds 92%, the best individual is considered the solution	Author's algorithm written in Microsoft Visual Studio 2022. Visualization algorithm implemented. Total code volume: 300 lines
Neuro-fuzzy expert systems	Production knowledge model for finding the decisive subgraph on an AND/OR graph. Forward reasoning. Knowledge base size of functional elements – 6–48 productions, and for the IDSS element – 15 productions. Fact base – up to 15 facts. Knowledge of experts and decision-makers was extracted by protocol analysis	Author's algorithm written in Microsoft Visual Studio 2022. Forward chaining used

4.1 CONCEPTUAL MODEL OF INTELLIGENT DECISION SUPPORT SYSTEM

A conceptual model is a model of the subject domain that defines a set of concepts, properties, and characteristics for describing this domain, as well as the laws of the processes occurring within it. The conceptual model, on the one hand, delimits the subject domain as a set of objects, connections, and relationships among them, as well as the procedures for transforming these objects during problem-solving. On the other hand, it introduces the developer's subjective views in the form of their knowledge and experience – concepts – into the modeling process.

Conceptual models of entities, for example, of tasks and intelligent decision support systems (IDSS), are constructed based on a conceptual model scheme containing 11 categories of concepts C, of which the following five are used:

Definition 1. A resource is a concept denoting an object that is at the disposal of the control subject for accomplishing tasks. The set of resources is denoted as $RES = C^{res} \subseteq C$.

Definition 2. A property is everything that is not within the boundaries of a given object. It is that which, while characterizing objects, does not form new objects. The set of properties is denoted as $PR = C^{pr} \subseteq C$.

Definition 3. An action is a concept denoting relation among resources as a result of activity, actions, and behavior. The set of actions is denoted as $ACT = C^{act} \subseteq C$.

Collective effects in intelligent decision support systems (IDSS) are presented in **Table 4.2.**

Table 4.2 Collective effects in IDSS

Effect	Brief description	Positive impact	Negative impact
Adaptation	Adjustment to the external environment or its modification for effective operation of the IDSS	Expands the range of tasks solved by the IDSS	Complicates analysis of IDSS performance
Boomerang	When information is distrusted, an opposite opinion to that contained in it arises	Unreliable information is not perceived or is considered deliberately false	Reliable information from an unreliable source may be regarded as false
Wave	Dissemination of ideas within the IDSS that correspond to the interests of its members	Collective refinement of ideas	Prolonged work of experts on unpromising ideas
Homeostasis	Maintenance of system parameters within limits away from critical values	Ensures long-term viability of the IDSS	Sometimes the IDSS in borderline states generates higher-quality decisions than under normal conditions
Group Egoism	The goals of the collective are more important than those of its members or society	None	The efficiency of the collective's activity may harm society
Conformism	The common opinion is truth; the opinion of an individual is nothing	None	Hinders the emergence of new approaches to problem-solving
Fashion (Imitation)	Voluntary adoption of the viewpoints on problems established within the collective	Basis for self-learning among collective members; facilitates mutual adaptation	Reduces the likelihood of original viewpoints and approaches to problem-solving
Ringelmann Effect	As the group size increases, the individual contribution to joint work decreases	Reduces the workload on individual IDSS participants	Decreases expert motivation for effective teamwork
Self-learning	Work of IDSS participants to improve their knowledge based on experience	Maintains the knowledge of IDSS participants in an up-to-date state	The acquired knowledge may be unsystematized, nonverbalized, or erroneous
Self-organization	Relationships among experts are dynamic and change during the work process	Adaptation to the external environment; each time a new relevant method is developed. Emergence of original approaches and synergy	Complicates analysis and external management of the collective
Synergy	Attainment of a collective result that individual experts cannot achieve independently	Emergence of a qualitatively superior collective result	Possible occurrence of negative synergy (dissynergy)
Social Facilitation	Enhancement of dominant reactions in the presence of others	Accelerates solutions to simple tasks for which the individual knows the answer	In complex tasks, increases the probability of erroneous responses

Definition 4. Value is a concept or number that indicates the quantity of measurement units. The set of values is denoted by $VAL = C^{val} \subseteq C$.

Definition 5. State is a concept that denotes the manifestation of processes occurring in a resource at a certain time. The set of states is denoted by $ST = C^{st} \subseteq C$.

A set of relations R is established between the concepts of these categories.

Definition 6. A relation is that which forms a thing from given elements (properties or other things). A relation is that which, being established between things, forms new things.

The fact of a relation being established between concepts is denoted by $r^{\alpha\beta}\left(c_1^{\alpha}, c^{\beta}\right)$. It is possible to distinguish relations between different categories of concepts: $R^{\alpha\,\beta} \subseteq R$ – the set of relations between concepts from the set C^{α} and the set C^{β}, where $\alpha, \beta \in \left\{"res", "pr", "act", "val"\right\}$.

Thus, a fragment of the conceptual model schema *sch*1 for structuring knowledge about the subject domain of the modeled task can be represented as follows

$$\begin{aligned} sch_1 = R^{res\,res}\left(RES,RES\right) \circ R^{pr\,pr}\left(PR,PR\right) \circ R^{act\,act}\left(ACT,ACT\right) \circ \\ \circ R^{val\,val}\left(VAL,VAL\right) \circ R^{st\,st}\left(ST,ST\right) \circ R^{res\,pr}\left(RES,PR\right) \circ \\ \circ R^{pr\,res}\left(PR,RES\right) \circ R^{res\,act}\left(RES,ACT\right) \circ R^{act\,res}\left(ACT,RES\right) \circ, \\ \circ R^{res\,st}\left(RES,ST\right) \circ R^{st\,res}\left(ST,RES\right) \circ R^{pr\,act}\left(PR,ACT\right) \circ \\ \circ R^{act\,pr}\left(ACT,PR\right) \circ R^{pr\,val}\left(PR,VAL\right) \circ R^{val\,pr}\left(VAL,PR\right) \circ \end{aligned} \tag{4.1}$$

where the symbol ∘ – denotes concatenation.

The micro-level conceptual model of the Intelligent Decision Support System (IDSS) can be expressed as follows

$$\widetilde{dss} = R^{res\,res}\left(prt^{dm}, env\right) \circ R^{res\,res}\left(PRT, PRT\right), \tag{4.2}$$

where prt^{dm} – the knowledge model of the decision-maker (DM); $env \in RES$ – external environment; $PRT = \left\{prt_1, \dots, prt_n, prt^{dm}\right\}$, $PRT \subseteq RES$ – the set of participants of the IDSS, including the decision-maker (DM) prt^{dm}; $R^{res\,res}$ – the set of "resource–resource" relations among the participants of the IDSS, as well as between the decision-maker (DM) and the external environment.

In work [13], it is noted that each participant $prt \in PRT$ of the IDSS has its own objective pr^{gsu}, which may coincide with or contradict the objectives of other participants. During the discussion, the experts exchange data pr^{dat}, knowledge pr^{knw}, explanations pr^{exp} and partial solutions pr^{dec} of the joint task. Thus, they perform a set of actions related to the transmission ACT^{itr} and reception ACT^{iac} of information, a set of professional functions ACT^{prt}, and exert influence on other participants of the IDSS and members of the surrounding environment by performing actions ACT^{conf}. Each expert has their own model *resmod* of the external environment, including the control object, as well as their own set of methods RES^{met} for problem solving. Considering the heterogeneous nature of complex tasks, for their successful solution the IDSS must include experts of various specializations, with different sets of problem-solving methods, that is $RES_i^{met} \neq RES_j^{met}$, where $i, j = 1,\dots,n$, $i \neq j$ – the index of a participant in the set PRT.

The conceptual model of an IDSS expert is expressed as follows

$$\begin{gathered} prt_i = r_1^{res\ pr}\left(prt, pr^{gsu}\right) \circ r_1^{res\ pr}\left(prt, pr^{dat}\right) \circ r_1^{res\ pr}\left(prt, pr^{knw}\right) \circ \\ \circ r_1^{res\ pr}\left(prt, pr^{\exp}\right) \circ r_1^{res\ pr}\left(prt, pr^{dec}\right) \circ r_2^{res\ act}\left(prt, ACT^{prt}\right) \circ \\ \circ r_2^{res\ act}\left(prt, ACT^{itr}\right) \circ r_2^{act\ pr}\left(ACT^{itr}, pr^{dat}\right) \circ r_2^{act\ pr}\left(ACT^{itr}, pr^{knw}\right) \circ, \\ \circ r_2^{act\ pr}\left(ACT^{itr}, pr^{\exp}\right) \circ r_2^{act\ pr}\left(ACT^{itr}, pr^{dec}\right) \circ r_2^{res\ act}\left(prt, ACT^{iac}\right) \circ \\ \circ r_2^{act\ pr}\left(ACT^{iac}, pr^{dat}\right) \circ r_2^{act\ pr}\left(ACT^{iac}, pr^{knw}\right) \circ r_2^{act\ pr}\left(ACT^{iac}, pr^{\exp}\right) \circ \\ \circ r_2^{act\ pr}\left(ACT^{iac}, pr^{dec}\right) \circ r_2^{res\ act}\left(prt, ACT^{conf}\right) \circ r_3^{res\ res}\left(prt, res^{\mathrm{mod}}\right) \circ \\ \circ r_3^{res\ res}\left(prt, RES^{met}\right), \end{gathered} \tag{4.3}$$

where $r_1^{res\ pr}$ — the "have property" relation, which establishes the correspondence between an IDSS participant and their properties;
$r_2^{res\ act}$ — the "perform" relation, which links a subject and the action they perform;
$r_2^{act\ pr}$— the "have property" relation, which links an action with its property;
$r_3^{res\ res}$ — the "include" relation, which links a whole and its parts.

Many relations *Rres res* in (4.2) consist of subsets of relations of various classes: cooperation $R_{coop}^{res\ res}$, competition $R_{comp}^{res\ res}$, neutrality $R_{neut}^{res\ res}$, trust $R_{trus}^{res\ res}$, pressure and conformism $R_{conf}^{res\ res}$, coordination $R_{coor}^{res\ res}$, dispute $R_{disp}^{res\ res}$, and others $R_{oth}^{res\ res}$. The subset of relations $R_{oth}^{res\ res}$ is introduced into the model to make it complete and extensible. Thus, the set $R^{res\ res}$ can be represented by the expression

$$\begin{gathered} R^{res\ res} = R_{coop}^{res\ res} \cup R_{comp}^{res\ res} \cup R_{neut}^{res\ res} \cup R_{trus}^{res\ res} \cup R_{conf}^{res\ res} \cup \\ \cup R_{coor}^{res\ res} \cup R_{disp}^{res\ res} \cup R_{oth}^{res\ res}. \end{gathered} \tag{4.4}$$

The composition of relations from the set $R^{res\ res}$ and its subsets is not known in advance and is determined during the operation of the IDSS in accordance with the interaction rules $INT \subseteq RES$ because of its self-organization. Owing to the dynamism of the links among experts and self-organization, the IDSS is capable of generating a new solution method relevant to the prevailing conditions, and the conceptual model of the IDSS as a self-organized entity, a method for solving a complex task, can be represented by the expression

$$\begin{gathered} RES_{dss}^{met} = R^{res\ res}\left(RES_1^{met}, RES_2^{met}\right) \circ \ldots \circ R^{res\ res}\left(RES_1^{met}, RES_n^{met}\right) \circ \\ \circ R^{res\ res}\left(RES_2^{met}, RES_1^{met}\right) \circ \ldots \circ R^{res\ res}\left(RES_2^{met}, RES_n^{met}\right) \circ \\ R^{res\ res}\left(RES_n^{met}, RES_1^{met}\right) \circ \ldots \circ R^{res\ res}\left(RES_n^{met}, RES_{n-1}^{met}\right), \end{gathered} \tag{4.5}$$

where the method RES_{dss}^{met}, generated by the IDSS in the process of solving a current task, represents an interconnected set of method sets RES_i^{met}, i=1,...,n, used by the experts in solving their partial tasks.

In solving the current task, the intensity and orientation of the relations $R^{res\,res}$ among the IDSS experts, and consequently among the methods they employ, change, leading to the development, in accordance with (4.5), of a new method relevant to the complex task, that is, a synergistic effect arises. The external manifestation of this effect is that the IDSS produces solutions of higher quality compared to the opinions of individual experts.

Taking the above into account, the macro-level model of the IDSS can be represented as follows

$$\widehat{dss} = \left(PRT, env, INT, \widetilde{DSS}, EFF\right), \tag{4.6}$$

where PRT – the set of IDSS participants described by the conceptual model (4.3);
env – the environment in which the IDSS operates;
INT – the set of elements structuring the interactions among experts;
$\widehat{dss}$ – set of IDSS micro-level models (4.5) corresponding to the specific functions of the experts within the IDSS and to the relations established among them;
EFF is the set of conceptual models of macro-level (collective) effects in the IDSS (**Table 4.1**): adaptation ad, boomerang bo, wave wa, homeostasis ho, group egoism ge, groupthink gt, fashion fa, Ringelmann effect re, self-learning sl, self-organization so, synergy se, and social facilitation sf. Let's consider in more detail the models of these macro-level effects.

Two types of adaptation are distinguished: passive and active. In the first case, the adapting system changes so as to perform its functions in the given environment in the best possible way. The conceptual model of such adaptation is expressed as follows

$$\begin{aligned} ad_p = {} & r_3^{res\,res}\left(dss, PRT\right) \circ r_2^{res\,act}\left(PRT, ACT^{iac}\right) \circ r_1^{act\,res}\left(ACT^{iac}, env\right) \circ \\ & \circ R_1^{res\,res}\left(\widetilde{DSS}, \widetilde{DSS}\right) \circ r_1^{res\,pr}\left(\widetilde{DSS}, PR^{cr}\right) \circ r_1^{pr\,val}\left(PR^{cr}, VAL^{cr}\right) \circ \\ & \circ r_1^{val\,val}\left(VAL^{cr}, VAL^{cr\,go}\right), \end{aligned} \tag{4.7}$$

where PR^{cr} – the set of criteria for evaluating the effectiveness of the IDSS;
VAL^{cr} – the set of values of critical parameters of the IDSS for micro-level models;
$VAL^{cr\,go}$ – the set of target values of critical parameters of the IDSS;
$r_1^{pr\,val}$ – the "have value" relation;
$r_1^{val\,val}$ – the relation of proximity between two values.

Active adaptation implies a change of the environment in order to maximize the efficiency criterion or an active search for such an environment. The conceptual model of active adaptation for the IDSS is as follows

$$\begin{aligned} ad_a = {} & r_3^{res\,res}\left(dss, PRT\right) \circ r_2^{res\,act}\left(PRT, ACT^{iac}\right) \circ r_1^{act\,res}\left(ACT^{iac}, env\right) \circ \\ & \circ R_1^{res\,res}\left(ENV, ENV\right) \circ r_1^{res\,pr}\left(\widetilde{DSS}, PR^{cr}\right) \circ r_1^{pr\,val}\left(PR^{cr}, VAL^{cr}\right) \circ \\ & \circ r_1^{val\,val}\left(VAL^{cr}, VAL^{cr\,go}\right) \circ r_2^{res\,act}\left(dss, ACT^{inf}\right) \circ r_1^{act\,res}\left(ACT^{inf}, env\right), \end{aligned} \tag{4.8}$$

where $ENV \subseteq RES$ – the set of external environments suitable for the operation of the IDSS; ACT^{inf} – the set of IDSS influences on the application environment.

The boomerang effect (bo) is the ignoring of, or identification as false, information originating from unreliable sources

$$bo = r_3^{res\ res}\left(dss, PRT\right) \circ r_2^{res\ act}\left(PRT, ACT_{trus}^{iac}\right) \circ R_{trus}^{res\ res}\left(PRT, env\right) \circ \\ \circ R_{trus}^{res\ res}\left(PRT, PRT\right), \tag{4.9}$$

where ACT_{trus}^{iac} – a set of actions for obtaining information that considers the relations of trust among the IDSS participants, as well as between the participants and information sources from the external environment.

According to **Table 4.1**, the wave effect (wa) is a mechanism for the dissemination of ideas and objectives within the IDSS that correspond to the interests of its members, transmitted to IDSS participants primarily from the "inner circle" of the source expert. Subsequently, these participants may modify the idea and transmit it to the IDSS participants within their own "inner circle". The wave effect is formally expressed as follows

$$wa = r_3^{res\ res}\left(dss, PRT\right) \circ r_2^{res\ act}\left(PRT, ACT_{trus}^{itr}\right) \circ R_{trus}^{res\ res}\left(PRT, PRT\right), \tag{4.10}$$

where ACT_{trus}^{itr} – the set of actions for transmitting information that considers the relations of trust among the IDSS participants.

The conceptual model of homeostasis (ho) in the IDSS is expressed as follows

$$ho = r_3^{res\ res}\left(dss, PRT\right) \circ r_2^{res\ act}\left(PRT, ACT^{iac}\right) \circ r_1^{act\ res}\left(ACT^{iac}, env\right) \circ \\ \circ R_1^{res\ res}\left(\widetilde{DSS}, \widetilde{DSS}\right) \circ r_1^{res\ pr}\left(\widetilde{DSS}, PR^{cr}\right) \circ r_1^{pr\ val}\left(PR^{cr}, VAL^{cr}\right) \circ \\ \circ r_1^{val\ val}\left(VAL^{cr}, VAL^{cr\ all}\right), \tag{4.11}$$

where $VAL^{cr\ all}$ – the set of permissible values of the critical parameters of the IDSS.

The group egoism effect (ge) consists in the IDSS disregarding the objectives of society and of individual members of the IDSS

$$ge = r_3^{res\ res}\left(dss, PRT\right) \circ r_2^{res\ act}\left(PRT, ACT^{conf}\right) \circ r_1^{act\ res}\left(ACT^{conf}, PRT\right) \circ \\ \circ r_1^{act\ res}\left(ACT^{conf}, env\right) \circ R_{conf}^{res\ res}\left(PRT, PRT\right), \tag{4.12}$$

where $r_1^{act\ res}$ – the "have as object" relation, which links an action with the object toward which it is directed.

The groupthink effect (gt) is the suppression of opinions of IDSS participants that differ from the opinions of the majority of the IDSS members

$$gt = r_3^{res\ res}\left(dss, PRT\right) \circ r_2^{res\ act}\left(PRT, ACT^{conf}\right) \circ r_1^{act\ res}\left(ACT^{conf}, PRT\right) \circ \\ \circ R_{conf}^{res\ res}\left(PRT, PRT\right). \tag{4.13}$$

The fashion effect (fa) consists in the voluntary adoption of the viewpoint on a problem that has become established within the collective

$$fa = r_3^{res\ res}\left(dss, PRT\right) \circ r_2^{res\ act}\left(PRT, ACT^{iac}\right) \circ r_2^{act\ pr}\left(ACT^{iac}, PR^{dec}\right). \tag{4.14}$$

According to **Table 4.1**, the Ringelmann effect (re) is the decrease in the intensity of individual work as the group size increases

$$\begin{aligned} re = {} & r_3^{res\ res}\left(dss, PRT\right) \circ r_2^{res\ act}\left(PRT, ACT^{prt}\right) \circ r_2^{act\ pr}\left(ACT^{prt}, PR^{efi}\right) \circ \\ & \circ r_2^{act\ pr}\left(ACT^{prt}, PR^{efc}\right) \circ r_1^{pr\ pr}\left(PR^{efi}, PR^{efc}\right), \end{aligned} \tag{4.15}$$

where PR^{efi} – the efficiency of performing an action in individual work, determined individually for each IDSS and each task. In general, efficiency is understood as an indicator that considers the assessment of the speed of decision-making and the quality of proposed solutions; PR^{efc} – the efficiency of performing an action during collective work; $r_1^{pr\ pr}$ – the "be greater than" relation.

The conceptual model of decision-maker (DM) self-learning sl_{dm} is expressed as follows

$$\begin{aligned} sl_{dm} = {} & r_3^{res\ res}\left(dss, PRT\right) \circ r_2^{res\ act}\left(PRT, ACT^{iac}\right) \circ r_1^{act\ res}\left(ACT^{iac}, env\right) \circ \\ & \circ R_1^{res\ res}\left(\widetilde{DSS}, \widetilde{DSS}\right) \circ r_1^{res\ pr}\left(\widetilde{dss}, PR^{cr}\right) \circ r_1^{pr\ val}\left(PR^{cr}, VAL^{cr\ pl}\right) \circ \\ & \circ r_1^{pr\ val}\left(PR^{cr}, VAL^{cr\ fct}\right) \circ r_3^{res\ res}\left(prt^{dm}, res^{fdb}\right) \circ r_1^{val\ val}\left(VAL^{cr\ pl}, VAL^{cr\ fct}\right) \circ \\ & \circ r_2^{pr\ act}\left(res^{fdb}, ACT^{lrn}\right) \circ r_1^{act\ res}\left(ACT^{lrn}, res^{rul}\right) \circ r_3^{res\ res}\left(res^{rul}, res^{ienv}\right) \circ \\ & \circ r_3^{res\ res}\left(res^{rul}, res^{idss}\right) \circ r_3^{res\ res}\left(res^{rul}, res^{ifct}\right), \end{aligned} \tag{4.16}$$

where $VAL^{cr\ pl}$ – the set of planned values of the IDSS efficiency criteria for the selected micro-level model $\widetilde{dss}$;
$VAL^{cr\ fct}$ – the set of actual values of the IDSS efficiency criteria for the selected micro-level model $\widetilde{dss}$;
res^{fdb} – the fuzzy knowledge base of the decision-maker (DM) for selecting micro-level IDSS models from the set $\widetilde{DSS}$;
ACT^{lrn} – learning and adjustment of the rules of the decision-maker's (DM's) fuzzy knowledge base res^{fdb};
res^{rul} – a rule of the decision-maker's (DM's) fuzzy knowledge base for selecting micro-level IDSS models from the set $\widetilde{DSS}$;
res^{ienv} – information about the external environment;
res^{idss} – information about the micro-level model $\widetilde{dss}$;
res^{ifct} – information about the actual values of the IDSS efficiency criteria corresponding to the selected model $\widetilde{dss}$.

Self-organization of the IDSS (so) is a specific effect in which the IDSS collective, without apparent external causes, creates or modifies the interrelations among participants and the organizational structures.

$$so = r_3^{res\ res}\left(dss, PRT\right) \circ r_2^{res\ act}\left(PRT, ACT^{iac}\right) \circ r_1^{act\ res}\left(ACT^{iac}, env\right) \circ \\ \circ R_1^{res\ res}\left(\widetilde{DSS}, \widetilde{DSS}\right), \tag{4.17}$$

where $r_1^{act\ res}$ – the "have as object" relation between an action and its resources;
$R_1^{res\ res}$ – the set of relations between the preceding micro-level model and the subsequent one in the course of their transformation.

The synergy effect (se) is the result of the interrelations among the IDSS participants during their collaborative work on a task, that is, the generation of an organizational structure relevant to the problem being solved. This effect in the IDSS is manifested in obtaining a collective solution of higher quality than any of the individual ones

$$se = r_3^{res\ res}\left(dss, PRT\right) \circ r_2^{res\ act}\left(PRT, ACT^{iac}\right) \circ r_1^{act\ res}\left(ACT^{iac}, env\right) \circ \\ \circ R_1^{res\ res}\left(\widetilde{DSS}, \widetilde{DSS}\right) \circ r_4^{res\ res}\left(PRT, RES^{dec}\right) \circ r_1^{act\ pr}\left(RES^{dec}, PR^{qua}\right) \circ \\ \circ r_4^{res\ res}\left(dss, res_{dss}^{dec}\right) \circ r_1^{res\ pr}\left(dss, pr_{dss}^{qua}\right) \circ r_1^{pr\ pr}\left(pr_{dss}^{qua}, PR^{qua}\right), \tag{4.18}$$

where RES^{dec} – the set of solutions to the task assigned to the IDSS, proposed by the experts as a result of individual work;
PR^{qua} – the set of quality indicators of the experts' individual solutions;
res_{dss}^{dec} – the solution produced by the IDSS as a result of the experts' collaborative work;
pr_{dss}^{qua} – the quality of the solution produced by the IDSS;
$r_4^{res\ res}$ – the relation that links an expert or the IDSS with the solution produced.

As shown in **Table 4.1**, social facilitation (SF) involves the enhancement of dominant responses in the presence of other experts; that is, it contributes to the acceleration of decision-making

$$sf = r_3^{res\ res}\left(dss, PRT\right) \circ r_2^{res\ act}\left(PRT, ACT^{prt}\right) \circ r_1^{act\ pr}\left(ACT^{prt}, PR^{spi}\right) \circ \\ \circ r_2^{act\ pr}\left(ACT^{prt}, PR^{spc}\right) \circ r_1^{pr\ pr}\left(PR^{spc}, PR^{spi}\right), \tag{4.19}$$

where PR^{spi} – the speed of performing an action during individual work; PR^{spc} – the speed of performing an action during collective work.

Analysis of expressions (4.7)–(4.19) has shown that certain macro-level effects are interrelated. For example, expressions (4.7), (4.11), (4.16), and (4.18) can be transformed using expression (4.17) as follows:

$$ad_p = so \circ r_1^{res\ pr}\left(\widetilde{DSS}, PR^{cr}\right) \circ r_1^{pr\ val}\left(PR^{cr}, VAL^{cr}\right) \circ r_1^{val\ val}\left(VAL^{cr}, VAL^{cr\ go}\right), \tag{4.20}$$

$$ho = so \circ r_1^{res\ pr}\left(\widetilde{DSS}, PR^{cr}\right) \circ r_1^{pr\ val}\left(PR^{cr}, VAL^{cr}\right) \circ r_1^{val\ val}\left(VAL^{cr}, VAL^{cr\ all}\right), \tag{4.21}$$

$$\begin{aligned} sl_{dm} = so \circ r_1^{res\ pr}\left(\widetilde{dss}, PR^{cr}\right) \circ r_1^{pr\ val}\left(PR^{cr}, VAL^{cr\ pl}\right) \circ \\ \circ r_1^{pr\ val}\left(PR^{cr}, VAL^{cr\ fct}\right) \circ r_3^{res\ res}\left(prt^{dm}, res^{fdb}\right) \circ r_1^{val\ val}\left(VAL^{cr\ pl}, VAL^{cr\ fct}\right) \circ \\ \circ r_2^{pr\ act}\left(res^{fdb}, ACT^{lrn}\right) \circ r_1^{act\ res}\left(ACT^{lrn}, res^{rul}\right) \circ r_3^{res\ res}\left(res^{rul}, res^{ienv}\right) \circ \\ \circ r_3^{res\ res}\left(res^{rul}, res^{idss}\right) \circ r_3^{res\ res}\left(res^{rul}, res^{ifct}\right); \end{aligned} \tag{4.22}$$

$$\begin{aligned} se = so \circ r_4^{res\ res}\left(PRT, RES^{dec}\right) \circ r_1^{act\ pr}\left(RES^{dec}, PR^{qua}\right) \circ \\ \circ r_4^{res\ res}\left(dss, res_{dss}^{dec}\right) \circ r_1^{res\ pr}\left(dss, pr_{dss}^{qua}\right) \circ r_1^{pr\ pr}\left(pr_{dss}^{qua}, PR^{qua}\right), \end{aligned} \tag{4.23}$$

Expressions (4.20)–(4.23) show that self-organization plays a special and fundamental role among the collective effects in the IDSS — it is the prerequisite for the emergence of other effects that positively influence the performance of the IDSS, such as adaptation, homeostasis, self-learning, and synergy [18].

4.2 COORDINATION MODELS IN HYBRID INTELLIGENT DECISION SUPPORT SYSTEMS

Coordination is a process that takes place in the IDSS during the solution of complex tasks and represents the sequence of analysis of the intermediate results of the solution of partial tasks and the issuance of controlling influences. Coordination is carried out by the decision-maker (DPR), but it can also be initiated by experts. In this work, the initiator of the coordination is DPR. The concept of "coordination" in relation to IDSS has not yet been investigated.

The study of real IDSSs determined the development of a new model for solving a complex task and a method for modeling the solution of complex tasks with the coordination of partial tasks in order to apply them to IDSS design.

4.2.1 MATHEMATICAL MODEL FOR SOLVING A COMPLEX CALCULATION TASK

Within the systems approach, tasks are traditionally considered as systems [12, 14] composed of individual interrelated subtasks that are connected and interact with one another. The order of interconnection and interaction among elements in an HIDSS (hybrid intelligent decision support system) is determined by its structure.

Let's denote the task-system as pr^{bu}, an individual task – pr^{bh}. Then $PRB^h = \left\{prb_1^h, \ldots, prb_{N_h}^h\right\}$ – the set of individual tasks included in pr^{bu}; $\widehat{PRB}^u = \left\{\widehat{prb}_1^u, \ldots, \widehat{prb}_{N_u}^u\right\}$ – the set of decompositions of tasks pr^{bu} [5]; $R^h = \left\{r_{wq} \mid w, q = 1, \ldots, N_h; q \neq w\right\}$ – the set of relations among individual tasks; N_h – the cardinality of the set PRB^h.

The model of a computational task of the IDSS can be represented as

$$prb^u = < PRB^h, \widehat{PRB}^u, R^h >, \tag{4.24}$$

and the model of each partial computational task as [5]

$$prb^h = < GL^h, DAT^h, MET^h >, \tag{4.25}$$

where GL^h – the final goal; DAT^h – input data; MET^h – conditions that specify how DAT^h are transformed into GL^h.

Model (4.24) satisfies all the properties of an IDSS:

– it consists of a set of elementary tasks PRB^h, among which relations R^h; are established; the connections are organized, which is reflected in the set of decompositions $\widehat{PRB}^u$;

– when solving the overall system task, the individual elementary tasks are predominantly isolated from the environment or its state is fixed, that is, the requirement is met that the internal connections within the system are much stronger than those with the external environment;

– a simple summation of the solutions of individual tasks does not yield a solution to the overall task as a whole [9, 10].

Model (4.25) has certain shortcomings. The main one is the inadequate representation of relations among the elements of the IDSS. Considering only the set of relations R^h among the partial tasks is insufficient. Studies of IDSSs have shown that, in most cases, experts are unable to provide professional solutions to partial tasks while taking into account the data on the complex task specified by the decision-maker (DM). Typically, there is a shortage of resources, particularly time, and errors occur in the formulation of the goal. Modification of the initial conditions of model (4.24) is impossible due to the absence of a crucial element – the image of the DM, which performs the function of a coordinator and reformulates the experts' goals depending on the situation.

The problem-solving process is thus considered as a system with a coordinator prb^k, whose function is to monitor and manage the process of solving individual partial tasks $prb_1^h, \ldots, prb_{N_h}^h$ by the experts during collective discussion. The coordinator is linked by relations $R^{hk} = \left\{ r^{kw} \mid w = 1, \ldots, N_h \right\}$ with each task prb^h in the IDSS prb^u, through which information is collected about the state of the process of solving an individual task by an expert. At certain moments in time, it also issues coordinating influences to modify the input data set – resources and goals. In this case, the model of a complex task with coordination is expressed as follows

$$prb^{uk} = < PRB^h, \widehat{PRB}^u, prb^k, R^h, R^{hk} >, \tag{4.26}$$

where prb^k – the coordinator; $R^{hk} = \left\{ r^{kw} \mid w = 1, \ldots, N_h \right\}$ – the sets of relations between the coordinator and the individual tasks.

A comparison of (4.24) and (4.26) shows that (4.26) is of a more general nature and reduces to (4.24) when the coordinator task is omitted from model (4.26), that is, in the case when the decision-maker (DM) in the IDSS does not perform coordination during the process of solving a complex task.

The coordinator element may be represented as a "coordinating task" (k-task), which should be "added" to the decomposition $\widehat{prb}^{u} \in \widehat{PRB}^{u}$ of the complex task prb^u, to adequately represent the specific features of planning tasks in the model.

Let $MET^{*} = \left\{met_1, \ldots, met_{N_{MET^*}}\right\}$ – be the set of conditions. Then, a correspondence ψ_1 can be defined

$$\psi_1 : SOL_1^h \otimes SOL_2^h \otimes MET^* \rightarrow SOL^u. \tag{4.27}$$

The elements of the correspondence ψ_1 – are tuples $\left(\left(sol_{\alpha}^{h1} sol_{\beta}^{h2} met_{\gamma}\right), sol_{\eta}^{u}\right)$, where $\alpha = 1,\ldots,N_{sh1}$; $\beta = 1,\ldots,N_{sh2}$; $\gamma = 1,\ldots,N_{MET}$; $\eta = 1,\ldots,N_{su}$, with the first component being a three-component vector consisting of the solution $sol_{\alpha}^{h1} \in SOL_1^h$ of task prb_1^h, the solution $sol_{\beta}^{h2} \in SOL_2^h$ of task prb_2^h and the coordinating condition $met_{\gamma} \in MET^*$, and the second component being the solution sol_{η}^{u} of the task prb^u.

The correspondence ψ_1 is not a function; it cannot be written analytically or computed, since the coordination conditions and the results of solving individual partial tasks are most often represented in natural language.

Let, as a result of solving the partial tasks prb_1^h and prb_2^h the solutions $sol_1^{h1} \in SOL_1^h$, and $sol_1^{h2} \in SOL_2^h$, and $\left\{\left(sol_1^{h1}, sol_1^{h2}\right)\right\} \otimes MET^* \nrightarrow SOL^u$, and let, that is, the obtained solutions sol_1^{h1}, and sol_1^{h2} for all $met_{\gamma} \in MET^*$ do not lead to the solution of task prb^u. The symbol «$\nrightarrow$» denotes the absence of a mapping from the set on the left-hand side of the symbol to the set on its right-hand side. In this case, it is necessary to re-solve tasks prb_1^h and prb_2^h. However, in the IDSS, there is often insufficient time to re-solve the tasks, so reasoning about the prb^U complex task is divided into separate, logically complete intermediate stages [13, 14], and at the end of these stages, the integrated result of solving the complex task is systematically verified – that is, an iterative process is organized. Consequently, the solutions of the partial task prb^h (the experts' lines of reasoning) are also divided into parts.

In this example, during the process of solving tasks prb_1^h and prb_2^h the following intermediate results will be obtained:

$$\begin{aligned} &sol_{11}^{h1} \Rightarrow sol_{12}^{h1} \Rightarrow \ldots \Rightarrow sol_{1s-1}^{h1} \Rightarrow sol_{1N_{sol}}^{h1} = sol_1^{h1}, \\ &sol_{11}^{h2} \Rightarrow sol_{12}^{h2} \Rightarrow \ldots \Rightarrow sol_{1s-1}^{h2} \Rightarrow sol_{1N_{sol}}^{h2} = sol_1^{h2}, \end{aligned} \tag{4.28}$$

where N_{sol} – the number of iteration steps into which the partial tasks are divided; and sol_1^{h1} and sol_1^{h2} – the results of solving the partial tasks prb_1^h and prb_2^h, respectively, obtained through the sequence of steps $1,\ldots, N_{sol}$.

Based on the coordinator's verification of the results obtained at a particular step, the relevance of influencing the course of solving the individual partial tasks prb_1^h prb_2^h is determined, so that the process of solving the complex task leads to the desired result – the goal. This influence is referred to as coordinating, and for simplicity, let's further denote the result of an intermediate stage without the first lower index, that is, sol_l^{h1} and sol_l^{h2}, where $l = 1, \ldots, N_{sol}$.

Following [17], let's introduce the set of coordinating influences

$$E = \left\{e_1^{\alpha}, \ldots, e_{N_{prt}}^{\alpha}\right\}, \tag{4.29}$$

where α – the type of coordinating influence, $\alpha=1,\ldots,6$. Let's consider each of the six types.

Integral coordination (α=1) – the decision-maker (DM) establishes various constraints (standards) on the input parameters $in_i^{h1} \in IN^{h1} \subseteq DAT^h$ of the partial task prb_1^h for a certain period of time

$$\int_0^T \left(in_i^{h1}(t) \right) dt = in_i^{h1H}, \tag{4.30}$$

where in_i^{h1H} – the standard for the input parameter $in_i^{h1} \in IN^{h1}$, $i = 1,\dots,N_{inh1}$; IN^{h1} – the set of input parameters of the partial task prb_1^h; $[0,T]$ – the time interval.

Precise coordination (α = 2) imposes constraints on the input parameters of the partial task so that at each moment of time t they are equal to the specified value

$$in_i^{h1}(t) = in_i^{h1Set}, \tag{4.31}$$

where $in_i^{h1}(t)$ – the input parameter; in_i^{h1Set} – the specified value of the parameter; t – an arbitrary moment in time when the fulfillment of the condition is verified $t \in \left[0,+\infty\right]$.

Interval coordination (α = 3) requires that the input parameter in_i^{h1} of the partial task (input data) belong to a specified interval

$$in_i^{h1} \in \left[val_{\min}^{h1i}, val_{\max}^{h1i} \right], \tag{4.32}$$

where $val_{\min}^{h1i}, val_{\max}^{h1i}$ – the interval boundaries.

Linguistic coordination (α = 4) is a condition specified in natural language. Temporal coordination, or synchronization of the solution of partial tasks (α = 5), to determine after what period an intermediate result must be provided. sol_l^{h1}, where l = 1, …, N_{sol} the results of solving the partial tasks are issued at certain time intervals

$$sol_l^{h1} \overset{\tau}{\Rightarrow} sol_{l+1}^{h1}, \tag{4.33}$$

where τ – the time interval after which the solution is issued; sol_l^{h1} and sol_{l+1}^{h1} – the results of solving the task prb_1^h after the i-th and i+1-th stages of solving the complex task, respectively.

Let's denote the situation in which the expert's line of reasoning does not change as a "null action", α = 6. For example, the decision-maker (DM) considers that it is unnecessary to influence the course of solving the partial tasks by the expert.

Since the results of solving the partial tasks are most often issued in natural language, the coordinating influences $e_1^{\alpha}, \dots, e_{N_{prt}}^{\alpha}$ are also most often presented in the same way.

Then, taking the above into account, it is possible to establish the correspondence

$$\psi_2 : \left\{ \left(sol_l^{h1}, sol_l^{h2} \right) \right\} \otimes MET^* \to E, \tag{4.34}$$

where l = 1,…, N_{sol} –1. The maximum value of the index l is taken as N_{sol} –1, since after stage N_{sol} it is no longer possible to coordinate the solution of the partial tasks – the final result has been obtained.

The elements of the correspondence ψ_2 – are pairs $\left(\left(sol_l^{h1}, sol_l^{h2}, met_\gamma\right), e_q^\alpha\right)$, for $l = 1,\ldots, N_{sol} - 1$; $\gamma = 1,\ldots, N_{MET}$; $q = 1,\ldots, N_{prt}$, where the first component is a three-component vector consisting of the solution $sol_l^{h1} \in SOL_1^h$ of the task prb_1^h, the solution $sol_l^{h2} \in SOL_2^h$ the task prb_2^h, and the coordinating condition $met_\gamma \in MET^*$, and the second component is the coordinating influence $e_q^\alpha \in E$. Analogous to (4.26), the correspondence (4.34) is not a function. It is multivalued, since it is possible to apply to the same partial task prb^h to apply several coordinating actions $e_q^\alpha \in E$.

Since there is a limit on the number of steps, when $l = N_{sol}$, there must be a correspondence

$$\psi_3 : \left\{\left(sol_l^{h1}, sol_l^{h2}\right)\right\} \otimes MET^* \to SOL^u. \tag{4.35}$$

The elements of the correspondence ψ_3 – are pairs of the form $\left(\left(sol_l^{h1}, sol_l^{h2}, met_\gamma\right), sol_\eta^u\right)$, where $l = 1,\ldots, N_{sol} - 1$, $\gamma = 1,\ldots,N_{MET}$; $\eta = 1,\ldots,N_{su}$ with the first component being a three-component vector consisting of the solution of task, the solution $sol_l^{h1} \in SOL_1^h$ of task prb_1^h, the solution $sol_l^{h2} \in SOL_2^h$ the task prb_2^h and the coordinating condition $met_\gamma \in MET^*$, and the second component being the solution $sol_\eta^u \in SOL^u$ task met^u. If ψ_3 is absent, that is, if, as a result of the search for the elements of ψ_3, it is found that $\psi_3 = \varnothing$, then the decision-maker (DM) must modify the set of coordination conditions MET*: introduce new conditions and remove some of the old ones.

The correspondence ψ_3 is a subset of the set $\psi_1, \psi_3 \subseteq \psi_1$, since the only difference is that ψ_3 specifies the concrete results of solving tasks prb_1^h and prb_2^h.

Since not all elements of the correspondence ψ_3 have as their second component $sol_\eta^u \in SOL^u$, hat, that satisfy the objectives of solving prb^u, let's denote by DAT_{ψ_3} the set of elements of the correspondence ψ_3, which second component satisfies the objectives of solving prb^u $DAT_{\psi_3} \in \psi_3$.

Taking the above into account, and considering model (4.25), the model of the k-task can be written as follows

$$prb^k = < SOL_1^h, SOL_2^h, \psi_2, DAT_{\psi_3} >, \tag{4.36}$$

where SOL_1^h, SOL_2^h – the input data for the coordinator task prb^k, expressed as a combination of numbers, words, and expressions;

$DAT_{\psi_3} \in \psi_3$ – the final goal of solving the coordinator task prb^k;

ψ_2 – the set of conditions that specify how the coordinating influences (4.34) are formed after each step, as a result of the application of which, after the final step, DAT_{ψ_3} can be obtained.

On the basis of the above, let's give the following definition of the coordination process: an iterative (multistage) process during which, after each iteration, the decision-maker (DM) analyzes the integrated result of solving the set of partial tasks. A coordinating influence is selected for the line of reasoning of each expert so that, upon completion of the process of solving the complex task, a maximally comprehensive overall result of its solution is obtained.

It may also be noted that as the number of partial tasks increases, the relevance of coordinating their solutions grows, since the number of relations (such as information exchange, use of common variables, or common constraints) among the task elements increases combinatorially.

In the present work, the decision-maker models do not consider: B_{prof} — the base of professional knowledge; B_{theor} — the base of theoretical knowledge; B_{prec} — the case base (experience).

Let's consider how the IDSS functions according to (4.37). Let the decision-maker be given a task prb^u, which he or she reduces to partial tasks $prb_1^h, \ldots, prb_{N_h}^h$. By analyzing (4.23) and (4.35), the following conclusions can be made: GL^h is contained in B_{prec} and B_{facts} — experience combined with facts allows the expert to determine what result should be obtained; MET^h is contained in B_{prof}, B_{theor}, B_{prec}, MET_{prti}, S_{prti} and In_{prti}; DAT^h is contained in B_{facts}.

In traditional IDSSs, described, for example, in [2], each expert, prt_q, q=1,...,N_{prt} receiving his or her partial task $prt_j^h, j = 1, \ldots, N_h$, finds its solution using his or her professional knowledge B_{prof} and theoretical knowledge B_{theor}. After completing the solution process, the expert provides the result $sol^{h\,j} \in SOL_j^h$, where SOL_j^h — the set of results obtained from solving the task prt_j^h, which can be represented as the correspondence ψ_4

$$\psi_4 : DAT^h \otimes B^U \to SOL^h,\ B^U = B_{prof} \cup B_{theor}. \tag{4.37}$$

The elements of the correspondence ψ_4 are tuples $\left(\left\{dat_\sigma^h\right\}, \left\{b_\beta^u\right\}, sol_\gamma^h\right)$, where σ = 1,..., N_{dath}; β = 1,..., N_b; γ = 1,..., N_{sh}, in which the first component is a two-component vector consisting of the list of input data $\left\{dat_\sigma^h\right\}, dat_\sigma^h \in DAT^h$ and the list of knowledge used by the expert $\left\{b_\beta^u\right\}, b_\beta^u \in B^U$ (professional knowledge — production rules; theoretical knowledge — analytical dependencies), and the second component is the result $sol_\gamma^u \in SOL^h$ of solving the task prb^h.

The correspondence ψ_4 is not a function (it cannot be represented analytically or computed by numerical methods), since the expert's knowledge and the results of solving the task element can be expressed in natural language. It is ambiguous, because with an incomplete set of input data, the expert may propose several alternative results; it is subjective, since each solution of task prb^h corresponds to at least one element from and it is not injective, as not every element of $DAT^h \otimes B^U$ corresponds to a solution of task prb^h.

Let denote the number of stages into which the experts divide the process of solving partial tasks, and let ΔN_{sol}, and sol_l^h — be the result of solving the partial task at the l-th stage, l = 1,..., N_{sol}. A time interval Δt, is allocated to each stage, since in practical tasks the total time T for solving the complex task prb^u, is strictly limited, and with Δt being constant, the number of stages is determined by the formula

$$N_{sol} = T / \Delta t. \tag{4.38}$$

It should be noted that in the process of solving a partial task prb^h, due to the coordinating influences of the decision maker, the input data DAT^h in (4.40) may be modified — additional information may be introduced, or outdated information may be replaced with new information. Let DAT_l^h the input data for the l-th stage, l = 1,...,N_{sol}. So DAT_1^h — the output data obtained from the decision maker, where $DAT_1^h = DAT^h$, and DAT_l^h, l = 2,...,N_{sol} — the output data of the subsequent stages. The index l denotes the stage number at which the output data are used. Let's define DAT_{l+1}^h — the output data of the (l+1)-th stage obtained after the coordinating influences of the decision maker concerning the modification of the data of the l-th stage.

4.2.2 CONCEPTUAL MODEL OF COORDINATION IN INTELLIGENT DECISION SUPPORT SYSTEMS

In the previous section, the coordinator model (4.36) was obtained. In an IDSS, the decision-maker (DM) functions as this element: it decomposes a complex task into a series of partial tasks, provides input data to the experts, and collects the solution results.

The drawback of existing IDSSs lies in the fact that coordination is performed only once — at the end of the problem-solving process — when the DM, after aggregating the results of solving the partial tasks into a single solution, draws a conclusion about its adequacy. If the integrated result is assessed as unsatisfactory, the possibility of solving the task anew may be lost. Therefore, it is relevant to develop IDSSs in which coordination occurs continuously throughout the process of solving a complex task.

Based on the IDSS model [6] and the model of a complex task with coordination, it is possible to construct the model of an IDSS with coordination

$$DSS =< PRT, prt^{dm}, R^{dm} >, \quad (4.39)$$

where $PRT = \{prt_q | q = 1,\ldots,N_{prt}\}$ — a set of expert models; prt^{dm} — the decision-maker model; $R^{dm} = \{r^{dm\ q} | \ldots, N_{prt}\}$ — the relations between the decision-maker and the experts, for example, relations of information exchange. Each expert works strictly within his or her own domain of knowledge $S_{prtq} \in S$, where S — the set of all domains of knowledge necessary for solving a complex task, and does not engage in any partial tasks outside his or her own domain $S_q \cap S_w = \varnothing$, for $q,w = 1,\ldots,N_{prt}$; $q \neq w$. Based on the considerations in [5] and taking into account that in real tasks the partial tasks are solved by experts step by step, the expert model can be expressed as

$$prt_q =< B_{prof}, B_{theor}, B_{prec}, B_{facts}, MET_{prtq}, S_{prtq}, In_{prtq}, \Delta t >, \quad (4.40)$$

where B_{prof} — production base of professional knowledge; B_{theor} — production base of theoretical knowledge; B_{prec} — case base (experience); B_{facts} — fact base; MET_{prtq} — set of reasoning methods; S_{prtq} — description of the expert's domain of knowledge, for example, in mathematics this includes the description of the mathematical language, basic concepts, and operations; In_{prtq} — interpreter that ensures the execution of a sequence of rules for solving a problem based on facts and rules stored in the databases and knowledge bases; Δt — the period during which experts provide intermediate solutions.

The decision-maker model can be constructed by analogy with (4.38)

$$prt^{dm} =< B_{prof}, B_{theor}, B_{prec}, B_{facts}, B_{ext}, MET_{prtdm}, S_{prtdm}, In_{prtdm}, E, T >, \quad (4.41)$$

where B_{ext} — production knowledge base concerning how to perform reduction, aggregation, comparison, and coordination; E — the set of coordinating processes; T — the time required to solve the complex task.

Expression (4.39), in comparison with (4.38), has significant differences. The production knowledge base B_{ext} concerns how the decision-maker manages the process of solving a complex task. This knowledge comes from other experts. The set E describes how the decision-maker can coordinate the work of the experts.

The sequence scheme of the stages of the expert's work on finding a solution to the partial task π^h can be described as follows

$$DAT_l^h \otimes B^U \otimes \left\{sol_1^h\right\} \otimes, \ldots, \otimes \left\{sol_{l-1}^h\right\} \Rightarrow \left\{sol_l^h\right\},\ l = 1, \ldots, N_{sol}. \qquad (4.42)$$

Output data DAT_l^h, $l = 1,\ldots,N_{sol}$ at each stage are supplemented by coordinating influences $e^\alpha \in E$, issued by the decision maker to the expert, which are determined based on the integrated result of the task solution prb^u at the (l—1)-th stage. In some cases, the decision maker may issue several coordinating influences to each expert. Let's assume that each expert receives one coordinating influence of a single type. Let's define the correspondence ψ_5

$$\psi_5 \left\{sol_l^u\right\} \otimes B_{ext} \rightarrow E,\ l = 1, \ldots, N_{sol} - 1. \qquad (4.43)$$

The maximum value of l equals N_{sol}-1, because after N_{sol} the stage, it is no longer possible to apply coordination, since the final result has been obtained; sol_l^u — the integrated result of the task solution prb^u at the l-th stage; $E = \left\{\varepsilon_1, \ldots, \varepsilon_{N_\varepsilon}\right\}$ — a set of vectors of the form $\left(e_1^1, \ldots, e_{N_{prt}}^6\right)$, each component of which is a coordinating action for the expert, $e_q^\alpha \in E$, $q = 1,\ldots, N_{prt}$.

Since the knowledge about integration is included in B_{ext} the decision maker (4.40), the integrated result sol_l^u of solving the complex task prb^u can be expressed as follows

$$\left\{sol_l^{h\,1}\right\} \otimes \ldots \otimes \left\{sol_l^{h\,N_h}\right\} \otimes B_{ext} \rightarrow \left\{sol_l^u\right\}, \qquad (4.44)$$

where $sol_l^{h\,1}, \ldots, sol_l^{h\,N_h}$ of solving the partial tasks $prt_1^h, \ldots, prt_{N_h}^h$ accordingly.

The elements of the correspondence ψ_5 — tuples $\left(\left(sol_l^u, \left\{b_{ext}^\mu\right\}\right), \varepsilon_p\right)$, so $l = 1,\ldots, N_{sol}$, $\mu = 1, \ldots, \mu_{N_\mu}$, $p = 1,\ldots, N\varepsilon$ where the first component is a two-component vector consisting of the integrated result sol_l^u solution of the task prb^u at the l-th stage and the list of the DM's knowledge used concerning how to perform comparisons, and $e \in E$ for the expert.

At the N_{sol}-th stage ($l = N_{sol}$) the vector of coordinating influences $\left(e_1^1, \ldots, e_{N_{prt}}^6\right)$, α = 6, i.e. DM that is, the DM does not issue coordinating influences to the experts but only aggregates (performs the integration of the solutions to the prt^h tasks into a single, integrated solution sol_l^u of the complex task prt^u) the results of their work. If the obtained integrated result sol_l^u does not satisfy the DM, it must revise the initial data of the task prb^u. It is necessary to change DAT_l^h for all prt^h or change the list of its knowledge B_{ext} and the experts' knowledge B_{prof} (models (4.40) and (4.39)), and after that, initiate the repeated operation of the DSS. The correspondence ψ_5 is not a function (cannot be expressed analytically or computed), since the DM's knowledge and the integrated result of the task solution prb^u can be represented in natural language. It is unambiguous since each expert is assigned a specific coordinating influence e_q^α, and therefore, the correspondence ψ_5 uniquely determines only one vector $e \in E$. It is subjective, because to each vector $e \in E$ there corresponds at least one element $\left\{sol_l^u\right\} \otimes B_{ext}$, and not injective, because not every element $\left\{sol_l^u\right\} \otimes B_{ext}$ corresponds to a vector $e \in E$.

The analysis of the above-described model of the IDSS with coordination allows the following conclusion to be drawn. In this case, the errors in solving the complex task will be detected and corrected before obtaining the result of solving the complex task prb^u. Previously, this required repeated solutions.

4.2.3 MATHEMATICAL MODEL OF THE FUNCTIONAL HYBRID SYSTEM WITH COORDINATION

In [5], the following conceptual model of the IDSS, based on the automaton approach [7–9], is presented, designed for solving a complex task prb^u

$$\begin{gathered} res_A^u = R_1^{res\ met}\left(res_A^u, met^u\right) \circ R_1^{res\ pr}\left(res_A^u, pr^{ui}\right) \circ R_1^{res\ pr}\left(res_A^u, pr^{uo}\right) \circ \\ \circ R_1^{res\ st}\left(res_A^u, st^u\right) \circ R_1^{st\ st}\left(st^u(t), st^u(t+1)\right) \circ R_1^{pr\ st}\left(pr^{ui}(t), st^u(t+1)\right) \circ \\ \circ R_1^{st\ pr}\left(st^{up}(t), pr^{uo}(t)\right) \circ R_1^{res\ res}\left(RES^e, RES^e\right) \circ R_1^{pr\ pr}\left(pr^{ui}, PR^{ei}\right) \circ \\ \circ R_2^{pr\ st}\left(PR^{eo}, pr^{uo}\right), \end{gathered} \tag{4.45}$$

where t – model time, $t \in \mathbb{N}$;
$\circ$ – concatenation symbol;
res_A^u – the IDSS-aggregate as a resource for solving a heterogeneous task;
met^u – the integrated method for solving a heterogeneous task;
pr^{ui} – output data $DATu$ [5] solution of a complex task prb^u, that are transmitted to the input of one or several elements res^e, constructed according to scheme (4.45) in accordance with the decomposition $\widehat{prb}^u$ task prb^u;
pr^{uo} – the output of one or several elements res^e, constructed according to scheme (4.45) in accordance with $\widehat{prb}^u$, which is the goal GL^u of solving the task prb^u;
$st^u(t)$ – the state of the IDSS at time t;
RES^e – a nonempty set composed of elements res^e, constructed in accordance with scheme (4.45);
PR^{ei}, PR^{eo} – the set of properties "input" and "output" of the elements from RES^e accordingly;
$R_1^{st\ st}$, $R_1^{pr\ st}$, $R_1^{st\ pr}$ – relations of the functioning of the IDSS;
$R_1^{res\ res}$ – relations of integration [5] of the elements;
$R_1^{pr\ pr}$ – relations between the inputs of the IDSS and the inputs of the elements;
$R_2^{pr\ pr}$ – relations between the outputs of the elements and the outputs of the IDSS.

The element res^e models the solution of a homogeneous partial task or performs auxiliary operations, constructed according to an autonomous method met^e and possesses the properties $PR^e \subseteq PR$, the most important of which are "input" pr^{ei}, "output" pr^{eo} *i* "state" st^i. Conceptual model of an IDSS element

$$\begin{gathered} res^e = R_1^{res\ met}\left(res^e, met^e\right) \circ R_1^{res\ pr}\left(res^e, pr^{ei}\right) \circ R_1^{res\ pr}\left(res^e, pr^{eo}\right) \circ \\ \circ R_1^{res\ st}\left(res^e, st^e\right) \circ R_1^{st\ st}\left(st^e(t), st^e(t+1)\right) \circ R_1^{pr\ st}\left(pr^{ei}(t), st^e(t+1)\right) \circ \\ \circ R_1^{st\ pr}\left(st^e(t), pr^{eo}(t)\right) \end{gathered} \tag{4.46}$$

where $R_1^{st\ st}$ $R_1^{pr\ st}$, $R_1^{st\ pr}$ – the "state – state", "input – state", and "state – output" relations, respectively. Among the set of $MET^e = \left\{ met_y^e \mid y = 1, \ldots, N_{met} \right\}$ autonomous methods, it is possible to distinguish met_1^e : analytical computations, met_2^e neurocomputations, met_3^e fuzzy computations, met_4^e reasoning based on experience, evolutionary computations, met_6^e statistical computations, met_7^e and logical reasoning. If between an element res^e and an autonomous method, met_y^e a relation is established $R_1^{res\ met}\left(res^e, met_y^e\right)$, it is possible to denote the element $res^{e\,y}$.

Relations $R_1^{pr\ pr}$ $R_2^{pr\ pr}$ (4.44) are defined on sets of variables DAT^u, GL^u, and on sets of variables DAT^h, GL^h of the partial tasks included in the complex task.

In [5], three possible cases are given:

1) a set of variables for prb^u coincides with the set of variables for prb^h, so $DAT^u = DAT^h$, $GL^u = GL^h$;

2) the set of variables for prb^h – a subset of the corresponding set prb^u, so $DAT^h = DAT^u$, $GL^h = GL^u$;

3) the set of variables of a subset of the corresponding set prb^h, so $DAT^h = DAT^u$, $GL^h = GL^u$.

Since the automaton approach is used for modeling, the state of the automaton is influenced only by the input signal. The output signal depends on the state of the automaton at the previous moment of automaton time and on the input signal.

The extension of models (4.43) and (4.44) is carried out based on the following considerations. In the process of coordination, the intermediate states of the solutions to partial tasks are monitored [11]. In the adopted notations (4.43), (4.44), these states are understood as the states (solution results) of the functional elements res^e, that simulate the solutions of partial tasks prb^h. From the analysis of these states, the properties of the "input" change during coordination pr^{ei} of one or several elements res^e.

To take this fact into account, let's introduce into the conceptual model of the IDSS (4.43), (4.44) the triple $R_1^{st\ pr}\left(st^u(t), pr^{ui}(t+1)\right)$. In other words, based on the state of the IDSS $st^u(t)$ at time t, the output data change $pr^{ui}(t+1)$ for the IDSS, but already now in time t+1, that is, for the next iteration. Many $R_1^{st\ pr}$ establish relationships between the state $st^u(t)$ hybrid res_A^u (4.43) at the current model time t and the state of the inputs of one or several elements res^e at the next step.

To make the necessary change to the inputs pr^{ei} of one or several functional elements res^e for (4.44) let's introduce the triple $R_1^{st\ act}\left(st^u, act^{ek}\right)$, where $ACT^{ek} = \left\{ act_1^{ek\,\alpha}, \ldots, act_{N_{prt}}^{ek\,\alpha} \right\}$ – a set of concepts denoting coordinating actions, which is identical to the set of coordinating actions E (4.30), where α – the type of coordinating influence, $\alpha = 1, \ldots, 6$.

The modified conceptual model for the IDSS with coordination

$$res_A^u = res_A^u \circ R_1^{st\ pr}\left(st^u(t), pr^{ui}(t+1)\right), \tag{4.47}$$

and the modified model of the IDSS element

$$res^e = res^e \circ R_1^{st\ act}\left(st^u, act^{ek}\right). \tag{4.48}$$

Relationships $R_1^{st\ pr}$ and $R_1^{st\ act}$ are not predetermined, just as $R_1^{st\ st}$ $R_1^{pr\ st}$, $R_1^{st\ pr}$ are recorded in the course of the IDSS operation and are the result of solving the k-task prb^k (4.33).

Let's consider an example of an IDSS consisting of three elements $res_1^{e\,1}$, $res_2^{e\,6}$, $res_3^{e\,7}$ for solving partial tasks, which it is possible to call functional [5], and one coordinating (technological) element $res_k^{e\,7}$ for solving the k-task, which determines the order of interaction of the functional elements. The input of the IDSS receives the initial data DAT^u, divided among the functional elements according to the decomposition $\widehat{prb}^u \in \widehat{PRB}^u$ of solving a complex task prb^u. At the output, it is possible to obtain the results of the operation of the functional elements $res_1^{e\,1}$, $res_2^{e\,6}$, $res_3^{e\,7}$, integrated into the overall solution SOL^u of the complex task prb^u.

At each moment in time, t_l the state of all elements is recorded (polled) $res_q^{e\,y}$. After that, $res_k^{e\,7}$ based on the state $st^u(t_l)$, the IDSS issues a coordinating action $act_q^{ek\,\alpha} \in ACT^{ek}$ for each element $res_q^{e\,y}$. In the process of processing by the technological element $res_k^{e\,7}$ state $st^u\left(t_l\right)$ of the IDSS, that is, the solution of the k-task prb^k the state changes of the technological element $res_k^{e\,7}$. Moreover, the time τ', allocated for such processing, must not exceed the period after which the state of the IDSS is recorded

$$\tau' \le T / N_{sol}, \tag{4.49}$$

where T – the time allocated for solving the complex task prb^u; N_{sol} – the total number of stages. The transitions between the states of the functional elements of the IDSS occur abruptly, since between the moments in time t_l this state $res_q^{e\,y}$ does not change.

Below is the conceptual model of the operation of the IDSS constructed according to (4.45) and (4.47)

$$\begin{aligned} & st^{e\,k}\left(t_0'\right) \Rightarrow \begin{Bmatrix} st^{e\,1}\left(t_0\right) \\ st^{e\,2}\left(t_0\right) \end{Bmatrix} \rightarrow \begin{Bmatrix} st^{e\,1}\left(t_1\right) \\ st^{e\,2}\left(t_1\right) \end{Bmatrix} \Rightarrow st^{e\,k}\left(t_0'\right) \rightarrow st^{e\,k}\left(t_1'\right) \Rightarrow \\ & \Rightarrow \begin{Bmatrix} st^{e\,1}\left(t_1\right) \\ st^{e\,2}\left(t_1\right) \end{Bmatrix} \rightarrow \begin{Bmatrix} st^{e\,1}\left(t_2\right) \\ st^{e\,2}\left(t_2\right) \end{Bmatrix} \Rightarrow st^{e\,k}\left(t_1'\right) \rightarrow st^{e\,k}\left(t_2'\right) \Rightarrow \ldots \Rightarrow \\ & \Rightarrow \begin{Bmatrix} st^{e\,1}\left(t_{p-1}\right) \\ st^{e\,2}\left(t_{p-1}\right) \end{Bmatrix} \rightarrow \begin{Bmatrix} st^{e\,1}\left(t_p\right) \\ st^{e\,2}\left(t_p\right) \end{Bmatrix} \Rightarrow st^{e\,k}\left(t_{p-1}'\right) \rightarrow st^{e\,k}\left(t_p'\right), \end{aligned} \tag{4.50}$$

where "$\Rightarrow$" denote the relations $R^{st\,st}$, that link states from different subspaces and define the transition from one homogeneous space to others during the functioning of the IDSS; "$\rightarrow$" – the transition between states within the corresponding subspace. The transitions "$\Rightarrow$" from the subspace of the technological element $res_q^{e\,7}$ model the issuance of coordinating actions from the decision maker to the experts. And the set of transitions "$\Rightarrow$" and "$\rightarrow$" allows modeling and tracing the process of self-organization during the operation of the IDSS.

In (4.49), curly brackets denote the beginning and completion of the parallel operation of the functional elements. From the model, it is evident that after each fixation of "$\}\Rightarrow$" of states, functional element $res_q^{e\,y}$ control is transferred to the technological element $res_q^{e\,7}$, and after it changes its state, control is transferred to a group of functional elements of the IDSS.

This model is related to the conceptual model presented in [5]

$$\left\{ \begin{array}{l} st^{e1}(t) \to st^{e1}(t+1) \to \ldots \to st^{e1}(t+n) \\ st^{e2}(t) \to st^{e2}(t+1) \to \ldots \to st^{e2}(t+n) \end{array} \right\}. \quad (4.51)$$

The model (4.49) is based on the idea that the same homogeneous task can be solved in parallel by different functional elements of the IDSS. The relations of integration among the elements arise as internal nonverbal images in the user's memory, allowing them to compare the dynamics of modeling a complex task from different viewpoints, which makes it possible to perceive aspects that cannot be revealed through modeling with a single model. In model (4.48), another assumption is developed: the inclusion of the DM model within the mathematical model of the IDSS leads to the emergence of a self-organization effect.

4.3 CONCEPTUAL MODEL OF CONSISTENCY IN INTELLIGENT DECISION SUPPORT SYSTEMS

Consistency is understood as the degree of similarity among the goals of the IDSS participants. According to [13], a goal is a state of affairs that the decision-maker (DM) seeks to achieve, and which has a certain subjective value for them. In [3], a goal is defined as an ideal anticipation of the result of activity that acts as its regulator, while in [7] it is described as a situation or set of situations that must be achieved during the functioning of the system within a specified time frame. Generalizing these definitions, it is possible to identify the main characteristics of a goal: it represents the state of the control object, acts as a regulator of activity, has a temporal nature (a function of time), and is subjectively valuable to the DM.

Definition 7. Goal pr^{gsu} of the expert as a control subject res^{su} – state st^{pou} of the control object res^{su}, which has value (utility) for the expert pr^{csu}, that determines its activity (sequence of actions) act^{dsu}, which must be achieved within a period of time pr^{t}.

The scheme of conceptual goal models can be represented in the form of

$$\begin{array}{c} pr^{gsu} = R^{res\ st}\left(res^{ou}, st^{pou}\right) \circ R^{res\ pr}\left(res^{su}, pr^{csu}\right) \circ R^{res\ act}\left(res^{ou}, act^{dsu}\right), \\ act^{dsu} = R^{act\ act}\left(ACT^{su}, ACT^{su}\right) \circ R^{act\ pr}\left(ACT^{su}, PR^{t}\right), \end{array} \quad (4.52)$$

where $R^{res\ st}$ – the "resource – state" relations, which assign to the control object its state;
$R^{res\ pr}$ – the "resource – property" relations, which determine the subjective usefulness of the state of the control object for the expert (the control subject);
$R^{pr\ act}$ – the "property – action" relations, which assign to the target state a sequence of actions act^{dsu};
ACT^{su} – the set of possible actions of the expert;
$R^{act\ act}$ – the "action – action" relations that determine the order of actions $act^{dsu} \in ACT^{su}$ in the sequence act^{dsu};
$R^{act\ pr}$ – the "action – property" relations between actions with CT^{su} and the time of their execution PR^{t}.

The state st^{pou} of the control object res^{ou} is determined by the values of its properties

$$st^{pou} = R^{res\ pr}\left(res^{ou}, PR^{ou}\right) \circ R^{pr\ val}\left(PR^{ou}, VAL^{ou}\right),$$

where $R^{res\ pr}$ — the "resource — property" relations, which define the set of properties of the control object, and $R^{pr\ val}$ — the "property — value" relations, where each property of the control object is associated with a set of values. One of the properties in the set Pr^{ou} may represent the time associated with the functioning of the control object. In this case, the expert's goal also becomes dynamic and changes over time.

Since, as noted above, the properties of the control object are considered variable when recording cause-and-effect relationships in one or another modeling method, several tools may be used in goal setting. This leads to the complexity of modeling decision-making when it is necessary to compare partial goals described by different methods. Such a situation arises, for example, when Pareto-optimal solutions exist, and it is necessary to select only one of them. Let's assume that there is a control object with two properties pr_1^{ou} i pr_2^{ou}, as well as two states of the control object st_1^{pou}, and st_2^{pou}, so st_1^{pou} closer to the target state, st^{pou}, than st_2^{pou}, according to the first criterion pr_1^{ou}, and st_2^{pou}, according to the second one pr_2^{ou}.

If the properties are represented by different variables (for example, stochastic and fuzzy linguistic ones) processed by different methods, it will be difficult to select one of the solutions. However, if the properties are represented by variables of the same type, it is possible to define a metric in a two-dimensional space of vectors representing the admissible states of the control object and determine the distance between st_1^{pou} and st^{pou}, and also st_2^{pou} and st^{pou}, after which they can be compared with one another. To avoid such situations, it is possible to choose a single method for representing all properties that define the state of the control object, and consequently, those used in describing the goals of the decision-maker (DM) and the experts. Analysis has shown that the apparatus of fuzzy set theory [10] is relevant for this purpose.

Definition 8. A fuzzy goal of an expert pr^{gsu} — a fuzzy set defined on the set of states of the control object $ST^{pou} \subseteq ST$, with a membership function $\mu^{pr^{gsu}}\left(st^{pou}\right)$, or, for brevity $\mu gsu\,(st^{pou})$.

The membership function $\mu^{gsu}\,(st^{pou})$ takes values on the set of real numbers within the interval [0; 1]. The greater its value, the closer the state of the control object st^{pou} is to the expert's goal st^{gsu}. The state st^{pou} of the control object is described by a set of its properties $PR^{ou} = \left\{pr_1^{ou}, \ldots, pr_{N_{pr}^{ou}}^{ou}\right\}$, represented by variables belonging to one of the classes listed in [5], that is

$$\mu^{gsu}\left(st^{pou}\right) = \mu^{gsu}\left(pr_1^{ou}, \ldots, pr_{N_{pr}^{ou}}^{ou}\right). \tag{4.53}$$

The value of the membership function is determined by substituting into (4.52) the values from the set VAL^{ou} of the control object's properties corresponding to this state, that is, it is described by the expression $\mu^{gsu}\left(val_1^{ou}, \ldots, val_{N_{val}^{ou}}^{ou}\right)$.

A fuzzy goal of an expert can be represented using one of the methods for constructing membership functions of fuzzy sets considered in [7]. The choice of method is determined by the IDSS developer. Below, to describe the causal relationships between goals and the interaction relations of experts, direct methods [7] for constructing fuzzy goals are used.

When the experts' goals are formalized, pairwise comparison can be performed, and the degree of closeness can be determined. One of the options for determining the degree of closeness between experts' goals is the calculation of the Euclidean or Hamming distance between fuzzy sets [4, 5].

However, their application to determining the degree of similarity of experts' goals is problematic: they are computed only under the condition of convergence of the series or integrals used in them. Otherwise, when $val_{\min}^{ou}=-\infty$ or $val_{\max}^{ou}=\infty$ where $val_{\min}^{ou}$ and $val_{\max}^{ou}$ the minimum and maximum values of the property pr^{ou}, that describes the state st^{pou}, the distance will be equal to infinity, even if one set includes another. In this case, a measure of similarity of fuzzy goals is proposed [6, 10, 12]

$$s(A,B)=0,5\cdot\left(\frac{\int_{val_{\min}^{ou}}^{val_{\max}^{ou}}\mu_{A\cap B}^{gsu}\left(pr^{ou}\right)d\left(pr^{ou}\right)}{\int_{val_{\min}^{ou}}^{val_{\max}^{ou}}\mu_{A}^{gsu}\left(pr^{ou}\right)d\left(pr^{ou}\right)}+\frac{\int_{val_{\min}^{ou}}^{val_{\max}^{ou}}\mu_{A\cap B}^{gsu}\left(pr^{ou}\right)d\left(pr^{ou}\right)}{\int_{val_{\min}^{ou}}^{val_{\max}^{ou}}\mu_{B}^{gsu}\left(pr^{ou}\right)d\left(pr^{ou}\right)}\right). \tag{4.54}$$

Analysis shows that, unlike the Euclidean or Hamming distance, relation (4.53) should be considered a measure of similarity between fuzzy sets rather than a distance between them, since it does not satisfy some of the conditions (specifically, (4.54) and (4.55)) required of a distance function in mathematics:

$$d(X,Y)\geq 0,$$

$$d(X,Y)=d(Y,X),$$

$$d(X,Z)\leq d(X,Y)+d(Y,Z), \tag{4.55}$$

$$d(X,X)=0. \tag{4.56}$$

After determining the measure of similarity between the experts' goals, it becomes possible to define the type of relations among them based on the level of consistency. Let's represent this as fuzzy sets on the universe of values of the similarity measure of goals s (on the set of real numbers within the interval [0;1]). The study identifies three types of relations according to the degree of consistency: competition, neutrality, and cooperation. The greater the value of the measure of similarity of the experts' goals (4.53), the closer their interaction.

Thus, the membership function of the fuzzy set "cooperation" should attain its maximum value at $s = 1$, while the membership function of the fuzzy set "competition" should attain its maximum at $s = 0$. The maximum of the membership function of the fuzzy set "neutrality" should be equidistant from these maxima, that is, located at the point $s = 0.5$. The membership functions of the fuzzy sets representing the relations of competition "*competition*" (*s*) =, of neutrality $\mu_{\text{"neutrality"}}(s)=\left(1+\left(6\cdot\left(s-0,5\right)\right)^{8}\right)^{-1}$, and cooperation $\mu_{\text{"cooperation"}}(s)=\left(1+\left(6\cdot\left(s-0,5\right)\right)^{8}\right)$.

Let's represent the relations between the participants of the IDSS according to the degree of consistency of the linguistic variable cl — "type of relation"

$$cl = \left\langle \beta^{cl}, T^{cl}, U^{cl}, G^{cl}, M^{cl} \right\rangle, \tag{4.57}$$

where β^{cl}= "type of relation" — the designation of the linguistic variable;
T^{cl}= {"competition"; "neutrality"; "cooperation"} — the set of names of the linguistic values of the variable (term set), which constitute the designations of the fuzzy variable;
U^{cl} =[0;1] — the domain of definition (universe) of fuzzy variables included in the definition of the linguistic variable;
$G^{cl} = \varnothing$ — a syntactic procedure that describes the process of formation of new terms from the elements of the set *T*;
$M^{cl} = \left\{ \mu_{\text{"competition"}}(s), \mu_{\text{"neutrality"}}(s), \mu_{\text{"cooperation"}}(s) \right\}$ — a semantic procedure that assigns to each term of the set *T* and to the terms formed by the procedure *G* a fuzzy set [6, 10, 12].

The value of the linguistic variable cl (type of relations) is the term with the maximum value of the membership function. To calculate it, it is necessary to determine the value of the membership function for each fuzzy set representing the relations and compare them with one another.

The fuzzy set with the maximum value of the membership function corresponds to the type of relations established between the pair of experts. It is possible to define the mapping "relation classifier" $rcl : \left(prt_i, prt_j\right) \rightarrow T^{cl}$, ag_i, $ag_j \in AG^*$, $i \neq j$, which assigns to each pair of participants of the IDSS (prt_i prt_j), one of the terms t_k^{cl} of the linguistic variable *cl*, that is, the type of relation. The mapping is defined as follows:

$$rcl : \left(prt_i, prt_j\right) = \underset{t_k^{cl} \in T^{cl}}{\text{argmax}} \left(\mu_{t_k^{cl}} \left(s\left(pr_p^{gsu}, pr_q^{gsu} \right) \right) \right), \tag{4.58}$$

so $r_1^{res\ pr}\left(prt_i, pr_p^{gsu}\right), r_1^{res\ pr}\left(prt_j, pr_q^{gsu}\right), i \neq j.$

Many values of this mapping form the matrix RCL, which classifies the relations among the IDSS participants. The rows and columns of the matrix represent the participants, and the elements $rcl_{i\,j} = rcl\left(prt_i, prt_j\right)$ — the class of relations among them. This matrix is used to identify the collective decision-making situation for the complex task.

Depending on the classes of relations present in the IDSS, three collective decision-making situations (micro-level IDSS models) can be distinguished for the task:

1. The cooperation situation $\widetilde{dss}_{coop}$, when the IDSS consists of cooperative and neutral participants and there are no competitive relations.

2. The neutrality situation $\widetilde{dss}_{neut}$ occurs when all relations in the IDSS are neutral.

3. The competition situation $\widetilde{dss}_{comp}$ occurs when the IDSS contains at least one pair of experts with a competitive relationship.

In such IDSSs, neutral and cooperative participants may also be present. In the presence of cooperative participants, they are regarded as a single notional participant; in this case, all remaining participants are either competitive or neutral.

Thus, the process of self-organization based on goal analysis can be divided into two parts: identification of the current collective decision-making situation (the micro-level model of the IDSS) and selection, from the set of possible situations, of the desired collective decision-making situation that is relevant to the conditions of the given task. Taking this into account, the self-organization model (4.27) can be rewritten as follows

$$\begin{aligned} so^{goa} = {} & r_2^{res\ act}\left(dss, ACT^{sen}\right)\circ r_1^{act\ res}\left(ACT^{sen}, env\right)\circ R_1^{res\ res}\left(\widetilde{DSS}, \widetilde{DSS}\right)\circ \\ & \circ r_3^{res\ res}\left(dss, prt^{dm}\right)\circ r_2^{res\ act}\left(prt^{dm}, act_{ia}\right)\circ r_1^{act\ res}\left(act_{ia}, \widetilde{dss}_{cur}\right)\circ \\ & \circ r_2^{res\ act}\left(prt^{dm}, act_{ac}\right)\circ r_1^{act\ res}\left(act_{ac}, \widetilde{DSS}\right)\circ r_2^{act\ res}\left(act_{ac}, \widetilde{dss}_{des}\right), \end{aligned} \tag{4.59}$$

where act_{ia} — the DM's action "identification of the current collective decision situation";
act_{ac} — the DM's action "selection of the desired collective decision situation from the set of possible ones";
$\widetilde{dss}_{cur}$ — the current collective decision situation (micro-level model of the IDSS);
$\widetilde{dss}_{des}$ — the collective decision situation desired by the DM in terms of the task parameters and its knowledge about the effectiveness of a particular situation from the set $\widetilde{DSS}$ of possible in the IDSS;
$r_2^{res\ act}$ — the "performs" relation, which links a subject and the action it performs;
$r_1^{act\ res}$ — the "has as an object" relation, which links an action and its resource;
$r_2^{act\ res}$ — the "has as a result" relation, which links an action and the result of its execution.

The first stage of identification act_{ia} (4.58) collective decision situations — formalization of the experts' goals considering the definition of the fuzzy goal (4.52). After the fuzzy goals of all experts have been determined, the next stage of identification is performed, act_{ia} (4.58), collective decision situations — pairwise comparison of goals and determination of their degree of consistency using measure (4.53).

Next, the type of relations between the experts is determined according to the degree of consistency using the linguistic variable cl "type of relation" (4.56).

The final stage of identification act_{ia} (4.58) collective decision situations — recognition of the collective decision situation using the matrix CL′. Depending on the classes of relations present in the matrix CL′, three collective decision-making situations are distinguished: cooperation $\widetilde{dss}_{coop}$, neutrality $\widetilde{dss}_{neut}$, and competition $\widetilde{dss}_{comp}$.

After identifying the current collective decision-making situation, the decision-maker (DM) selects act_{ac} (4.58) from the set of possible collective decision-making situations that correspond to the conditions of the given task. Depending on the task parameters and their knowledge of the effectiveness of a particular collective decision situation, the decision-maker (DM) may seek to establish one of them. This is necessary to increase the efficiency of the IDSS operation or to attempt to change it if the discussion reaches an impasse.

CONCLUSIONS

The study proposes a polymodel complex for managing the resources of intelligent decision support systems. The novelty of the proposed polymodel complex is:

– in a comprehensive description of the process of functioning of intelligent decision support systems. This allows to increase the accuracy of modeling intelligent decision support systems for subsequent management decisions;

– descriptions of both static and dynamic processes that occur in intelligent decision support systems;

– ability to simulate both a single process that takes place in intelligent decision support systems, and to comprehensively simulate those processes that take place in them;

– in establishing the conceptual dependencies of the process of functioning of intelligent decision support systems. This allows to describe the interaction of individual models at all stages of solving calculation tasks;

– descriptions of coordination processes in hybrid intelligent decision support systems, which improves the reliability of management decision-making;

– modeling of processes for solving complex calculation tasks in intelligent decision support systems, due to the conceptual description of the specified process;

– coordination of calculation processes in intelligent decision support systems, which achieves a decrease in the number of computing resources of systems;

– complex dispute resolution, due to a complex of appropriate mathematical models.

The proposed polymodel complex should be used to solve the task of resource management of intelligent decision support systems characterized by a high degree of complexity.

CONFLICT OF INTEREST

The authors declare that they have no conflict of interest in relation to this research, whether financial, personal, authorship or otherwise, that could affect the research and its results presented in this paper.

USE OF ARTIFICIAL INTELLIGENCE

The authors confirm that they did not use artificial intelligence technologies in creating the submitted work.

REFERENCES

1. Dudnyk, V., Sinenko, Y., Matsyk, M., Demchenko, Y., Zhyvotovskyi, R., Repilo, I. et al. (2020). Development of a method for training artificial neural networks for intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 3 (2 (105)), 37–47. https://doi.org/10.15587/1729-4061.2020.203301

2. Sova, O., Shyshatskyi, A., Salnikova, O., Zhuk, O., Trotsko, O., Hrokholskyi, Y. (2021). Development of a method for assessment and forecasting of the radio electronic environment. EUREKA: Physics and Engineering, 4, 30–40. https://doi.org/10.21303/2461-4262.2021.001940
3. Pievtsov, H., Turinskyi, O., Zhyvotovskyi, R., Sova, O., Zvieriev, O., Lanetskii, B., Shyshatskyi, A. (2020). Development of an advanced method of finding solutions for neuro-fuzzy expert systems of analysis of the radioelectronic situation. EUREKA: Physics and Engineering, 4, 78–89. https://doi.org/10.21303/2461-4262.2020.001353
4. Zuiev, P., Zhyvotovskyi, R., Zvieriev, O., Hatsenko, S., Kuprii, V., Nakonechnyi, O. et al. (2020). Development of complex methodology of processing heterogeneous data in intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 4 (9 (106)), 14–23. https://doi.org/10.15587/1729-4061.2020.208554
5. Kuchuk, N., Mohammed, A. S., Shyshatskyi, A., Nalapko, O. (2019). The Method of Improving the Efficiency of Routes Selection in Networks of Connection with the Possibility of Self-Organization. International Journal of Advanced Trends in Computer Science and Engineering, 8 (1.2), 1–6.
6. Shyshatskyi, A., Zvieriev, O., Salnikova, O., Demchenko, Ye., Trotsko, O., Neroznak, Ye. (2020). Complex Methods of Processing Different Data in Intellectual Systems for Decision Support System. International Journal of Advanced Trends in Computer Science and Engineering, 9 (4), 5583–5590. https://doi.org/10.30534/ijatcse/2020/206942020
7. Pozna, C., Precup, R.-E., Horvath, E., Petriu, E. M. (2022). Hybrid Particle Filter–Particle Swarm Optimization Algorithm and Application to Fuzzy Controlled Servo Systems. IEEE Transactions on Fuzzy Systems, 30 (10), 4286–4297. https://doi.org/10.1109/tfuzz.2022.3146986
8. Yang, X.-S., Deb, S. (2013). Cuckoo search: recent advances and applications. Neural Computing and Applications, 24 (1), 169–174. https://doi.org/10.1007/s00521-013-1367-1
9. Mirjalili, S. (2015). The Ant Lion Optimizer. Advances in Engineering Software, 83, 80–98. https://doi.org/10.1016/j.advengsoft.2015.01.010
10. Yu, J. J. Q., Li, V. O. K. (2015). A social spider algorithm for global optimization. Applied Soft Computing, 30, 614–627. https://doi.org/10.1016/j.asoc.2015.02.014
11. Mirjalili, S., Mirjalili, S. M., Lewis, A. (2014). Grey Wolf Optimizer. Advances in Engineering Software, 69, 46–61. https://doi.org/10.1016/j.advengsoft.2013.12.007
12. Koval, V., Nechyporuk, O., Shyshatskyi, A., Nalapko, O., Shknai, O., Zhyvylo, Y. et al. (2023). Improvement of the optimization method based on the cat pack algorithm. Eastern-European Journal of Enterprise Technologies, 1 (9 (121)), 41–48. https://doi.org/10.15587/1729-4061.2023.273786
13. Gupta, E., Saxena, A. (2015). Robust generation control strategy based on Grey Wolf Optimizer. Journal of Electrical Systems, 11, 174–188.
14. Chaman-Motlagh, A. (2015). Superdefect Photonic Crystal Filter Optimization Using Grey Wolf Optimizer. IEEE Photonics Technology Letters, 27 (22), 2355–2358. https://doi.org/10.1109/lpt.2015.2464332
15. Nuaekaew, K., Artrit, P., Pholdee, N., Bureerat, S. (2017). Optimal reactive power dispatch problem using a two-archive multi-objective grey wolf optimizer. Expert Systems with Applications, 87, 79–89. https://doi.org/10.1016/j.eswa.2017.06.009

16. Koval, M., Sova, O., Orlov, O., Shyshatskyi, A., Artabaiev, Y., Shknai, O. et al. (2022). Improvement of complex resource management of special-purpose communication systems. Eastern-European Journal of Enterprise Technologies, 5 (9 (119)), 34–44. https://doi.org/10.15587/1729-4061.2022.266009
17. Ali, M., El-Hameed, M. A., Farahat, M. A. (2017). Effective parameters' identification for polymer electrolyte membrane fuel cell models using grey wolf optimizer. Renewable Energy, 111, 455–462. https://doi.org/10.1016/j.renene.2017.04.036
18. Zhang, S., Zhou, Y. (2017). Template matching using grey wolf optimizer with lateral inhibition. Optik, 130, 1229–1243. https://doi.org/10.1016/j.ijleo.2016.11.173
19. Khouni, S. E., Menacer, T. (2023). Nizar optimization algorithm: a novel metaheuristic algorithm for global optimization and engineering applications. The Journal of Supercomputing, 80 (3), 3229–3281. https://doi.org/10.1007/s11227-023-05579-4
20. Saremi, S., Mirjalili, S., Lewis, A. (2017). Grasshopper Optimisation Algorithm: Theory and application. Advances in Engineering Software, 105, 30–47. https://doi.org/10.1016/j.advengsoft.2017.01.004
21. Braik, M. S. (2021). Chameleon Swarm Algorithm: A bio-inspired optimizer for solving engineering design problems. Expert Systems with Applications, 174, 114685. https://doi.org/10.1016/j.eswa.2021.114685
22. Yapici, H., Cetinkaya, N. (2019). A new meta-heuristic optimizer: Pathfinder algorithm. Applied Soft Computing, 78, 545–568. https://doi.org/10.1016/j.asoc.2019.03.012
23. , H., Qiao, P. (2014). Pigeon-inspired optimization: a new swarm intelligence optimizer for air robot path planning. International Journal of Intelligent Computing and Cybernetics, 7 (1), 24–37. https://doi.org/10.1108/ijicc-02-2014-0005
24. Yang, X. S (2012). Flower pollination algorithm for global optimization. Unconventional computing and natural computation. Berlin, Heidelberg: Springer, 240–249. https://doi.org/10.1007/978-3-642-32894-7_27
25. Gomes, G. F., da Cunha, S. S., Ancelotti, A. C. (2018). A sunflower optimization (SFO) algorithm applied to damage identification on laminated composite plates. Engineering with Computers, 35 (2), 619–626. https://doi.org/10.1007/s00366-018-0620-8
26. Mehrabian, A. R., Lucas, C. (2006). A novel numerical optimization algorithm inspired from weed colonization. Ecological Informatics, 1 (4), 355–366. https://doi.org/10.1016/j.ecoinf.2006.07.003
27. Qi, X., Zhu, Y., Chen, H., Zhang, D., Niu, B. (2013). An idea based on plant root growth for numerical optimization. Intelligent Computing Theories and Technology. Berlin, Heidelberg: Springer, 571–578. https://doi.org/10.1007/978-3-642-39482-9_66
28. Bezuhlyi, V., Oliynyk, V., Romanenko, I., Zhuk, O., Kuzavkov, V., Borysov, O., Korobchenko, S., Ostapchuk, E., Davydenko, T., Shyshatskyi, A. (2021). Development of object state estimation method in intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 5 (3 (113)), 54–64. https://doi.org/10.15587/1729-4061.2021.239854
29. Mahdi, Q. A., Shyshatskyi, A., Prokopenko, Y., Ivakhnenko, T., Kupriyenko, D., Golian, V. et al. (2021). Development of estimation and forecasting method in intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 3 (9 (111)), 51–62. https://doi.org/10.15587/1729-4061.2021.232718

30. Sova, O., Radzivilov, H., Shyshatskyi, A., Shevchenko, D., Molodetskyi, B., Stryhun, V. et al. (2022). Development of the method of increasing the efficiency of information transfer in the special purpose networks. Eastern-European Journal of Enterprise Technologies, 3 (4 (117)), 6–14. https://doi.org/10.15587/1729-4061.2022.259727
31. Zhang, H., Zhu, Y., Chen, H. (2013). Root growth model: a novel approach to numerical function optimization and simulation of plant root system. Soft Computing, 18 (3), 521–537. https://doi.org/10.1007/s00500-013-1073-z
32. Labbi, Y., Attous, D. B., Gabbar, H. A., Mahdad, B., Zidan, A. (2016). A new rooted tree optimization algorithm for economic dispatch with valve-point effect. International Journal of Electrical Power & Energy Systems, 79, 298–311. https://doi.org/10.1016/j.ijepes.2016.01.028
33. Murase, H. (2000). Finite element inverse analysis using a photosynthetic algorithm. Computers and Electronics in Agriculture, 29 (1-2), 115–123. https://doi.org/10.1016/s0168-1699(00)00139-3
34. Zhao, S., Zhang, T., Ma, S., Chen, M. (2022). Dandelion Optimizer: A nature-inspired metaheuristic algorithm for engineering applications. Engineering Applications of Artificial Intelligence, 114, 105075. https://doi.org/10.1016/j.engappai.2022.105075
35. Paliwal, N., Srivastava, L., Pandit, M. (2020). Application of grey wolf optimization algorithm for load frequency control in multi-source single area power system. Evolutionary Intelligence, 15 (1), 563–584. https://doi.org/10.1007/s12065-020-00530-5
36. Dorigo, M., Blum, C. (2005). Ant colony optimization theory: A survey. Theoretical Computer Science, 344 (2-3), 243–278. https://doi.org/10.1016/j.tcs.2005.05.020
37. Poli, R., Kennedy, J., Blackwell, T. (2007). Particle swarm optimization. Swarm Intelligence, 1 (1), 33–57. https://doi.org/10.1007/s11721-007-0002-0
38. Bansal, J. C., Sharma, H., Jadon, S. S., Clerc, M. (2014). Spider Monkey Optimization algorithm for numerical optimization. Memetic Computing, 6 (1), 31–47. https://doi.org/10.1007/s12293-013-0128-0
39. Yeromina, N., Kurban, V., Mykus, S., Peredrii, O., Voloshchenko, O., Kosenko, V. et al. (2021). The Creation of the Database for Mobile Robots Navigation under the Conditions of Flexible Change of Flight Assignment. International Journal of Emerging Technology and Advanced Engineering, 11 (5), 37–44. https://doi.org/10.46338/ijetae0521_05
40. Maccarone, A. D., Brzorad, J. N., Stone H. M. (2008). Characteristics And Energetics Of Great Egret And Snowy Egret Foraging Flights. Waterbirds, 31 (4), 541–549. https://doi.org/10.1675/1524-4695-31.4.541
41. Ramaji, I. J., Memari, A. M. (2018). Interpretation of structural analytical models from the coordination view in building information models. Automation in Construction, 90, 117–133. https://doi.org/10.1016/j.autcon.2018.02.025
42. Pérez-González, C. J., Colebrook, M., Roda-García, J. L., Rosa-Remedios, C. B. (2019). Developing a data analytics platform to support decision making in emergency and security management. Expert Systems with Applications, 120, 167–184. https://doi.org/10.1016/j.eswa.2018.11.023
43. Chen, H. (2018). Evaluation of Personalized Service Level for Library Information Management Based on Fuzzy Analytic Hierarchy Process. Procedia Computer Science, 131, 952–958. https://doi.org/10.1016/j.procs.2018.04.233

44. Chan, H. K., Sun, X., Chung, S.-H. (2019). When should fuzzy analytic hierarchy process be used instead of analytic hierarchy process? Decision Support Systems, 125, 113114. https://doi.org/10.1016/j.dss.2019.113114
45. Osman, A. M. S. (2019). A novel big data analytics framework for smart cities. Future Generation Computer Systems, 91, 620–633. https://doi.org/10.1016/j.future.2018.06.046
46. Nechyporuk, O., Sova, O., Shyshatskyi, A., Kravchenko, S., Nalapko, O., Shknai, O. et al. (2023). Development of a method of complex analysis and multidimensional forecasting of the state of intelligence objects. Eastern-European Journal of Enterprise Technologies, 2 (4 (122)), 31–41. https://doi.org/10.15587/1729-4061.2023.276168
47. Merrikh-Bayat, F. (2015). The runner-root algorithm: A metaheuristic for solving unimodal and multimodal optimization problems inspired by runners and roots of plants in nature. Applied Soft Computing, 33, 292–303. https://doi.org/10.1016/j.asoc.2015.04.048
48. Poliarush, O., Krepych, S., Spivak, I. (2023). hybrid approach for data filtering and machine learning inside content management system. Advanced Information Systems, 7 (4), 70–74. https://doi.org/10.20998/2522-9052.2023.4.09
49. Balochian, S., Baloochian, H. (2019). Social mimic optimization algorithm and engineering applications. Expert Systems with Applications, 134, 178–191. https://doi.org/10.1016/j.eswa.2019.05.035
50. Lenord Melvix, J. S. M. (2014). Greedy Politics Optimization: Metaheuristic inspired by political strategies adopted during state assembly elections. 2014 IEEE International Advance Computing Conference (IACC), 1157–1162. https://doi.org/10.1109/iadcc.2014.6779490
51. Moosavian, N., Roodsari, B. K. (2014). Soccer League Competition Algorithm, a New Method for Solving Systems of Nonlinear Equations. International Journal of Intelligence Science, 4 (1), 7–16. https://doi.org/10.4236/ijis.2014.41002
52. Hayyolalam, V., Pourhaji Kazem, A. A. (2020). Black Widow Optimization Algorithm: A novel meta-heuristic approach for solving engineering optimization problems. Engineering Applications of Artificial Intelligence, 87, 103249. https://doi.org/10.1016/j.engappai.2019.103249
53. Abualigah, L., Yousri, D., Abd Elaziz, M., Ewees, A. A., Al-qaness, M. A. A., Gandomi, A. H. (2021). Aquila Optimizer: A novel meta-heuristic optimization algorithm. Computers & Industrial Engineering, 157, 107250. https://doi.org/10.1016/j.cie.2021.107250
54. Hodlevskyi, M., Burlakov, G. (2023). Information technology of quality improvement planning of process subsets of the spice model. Advanced Information Systems, 7 (4), 52–59. https://doi.org/10.20998/2522-9052.2023.4.06
55. Askari, Q., Younas, I., Saeed, M. (2020). Political Optimizer: A novel socio-inspired meta-heuristic for global optimization. Knowledge-Based Systems, 195, 105709. https://doi.org/10.1016/j.knosys.2020.105709
56. Mohamed, A. W., Hadi, A. A., Mohamed, A. K. (2019). Gaining-sharing knowledge based algorithm for solving optimization problems: a novel nature-inspired algorithm. International Journal of Machine Learning and Cybernetics, 11 (7), 1501–1529. https://doi.org/10.1007/s13042-019-01053-x
57. Gödri, I., Kardos, C., Pfeiffer, A., Váncza, J. (2019). Data analytics-based decision support workflow for high-mix low-volume production systems. CIRP Annals, 68 (1), 471–474. https://doi.org/10.1016/j.cirp.2019.04.001

58. Harding, J. L. (2013). Data quality in the integration and analysis of data from multiple sources: some research challenges. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XL-2/W1, 59–63. https://doi.org/10.5194/isprsarchives-xl-2-w1-59-2013
59. Orouskhani, M., Orouskhani, Y., Mansouri, M., Teshnehlab, M. (2013). A Novel Cat Swarm Optimization Algorithm for Unconstrained Optimization Problems. International Journal of Information Technology and Computer Science, 5 (11), 32–41. https://doi.org/10.5815/ijitcs.2013.11.04
60. Karaboga, D., Basturk, B. (2007). A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. Journal of Global Optimization, 39 (3), 459–471. https://doi.org/10.1007/s10898-007-9149-x
61. Voznytsia, A., Sharonova, N., Babenko, V., Ostapchuk, V., Neronov, S., Feoktystov, S. et. al (2025). Development of methods for intelligent assessment of parameters in decision support systems. Eastern-European Journal of Enterprise Technologies, 4 (4 (136)), 73–82. https://doi.org/10.15587/1729-4061.2025.337528
62. Sova, O., Radzivilov, H., Shyshatskyi, A., Shvets, P., Tkachenko, V., Nevhad, S. et. al (2022). Development of a method to improve the reliability of assessing the condition of the monitoring object in special-purpose information systems. Eastern-European Journal of Enterprise Technologies, 2 (3 (116)), 6–14. https://doi.org/10.15587/1729-4061.2022.254122

DOI: 10.15587/978-617-8360-23-8.CH5

5 DEVELOPMENT OF METHODS FOR EVALUATING COMPLEX ORGANIZATIONAL AND TECHNICAL SYSTEMS USING THE THEORY OF ARTIFICIAL INTELLIGENCE

Andrii Shyshatskyi, Oleksandr Zhuk, Pavlo Zhuk, Andrii Veretnov, Olena Shaposhnikova, Yaroslav Melnyk

ABSTRACT

Complex organizational and technical systems are the object of the study. The problem that is solved in the study is an increase in the efficiency of the assessment of the operation process yof complex organizational and technical systems (OTS) while maintaining a given level of reliability.

Methods of evaluating complex organizational and technical systems using the theory of artificial intelligence were developed. The originality of the study is:

– in full coverage of critical events occurring during the OTS operation. This is achieved due to the use of the Dempster-Schafer theory, which achieves the completeness of the assessment of the entire spectrum of critical events in the OTS;

– in a comprehensive description of the process of OTS operation. This makes it possible to increase the accuracy of OTS modeling for subsequent management decisions;

– adapt to the type and duration of abnormalities due to multi-level adaptation of the artificial immune system;

– in the ability to carry out initial adjustment of OTS knowledge bases using an improved genetic algorithm. This allows to reduce the computational complexity during the further formation of the OTS knowledge base by reducing the metric of rule formation in the OTS knowledge base;

– in the ability to model the nature of the development of atypical events in the OTS due to the use of time series, which achieves the possibility of developing preventive measures to minimize the impact of the specified events on the process of OTS operation;

– in the gradual reduction of the metric of the formation of the knowledge base about the states of OTS, due to the training of agents of the improved genetic algorithm. This allows to reduce the number of computing resources of the subsystem for assessing the OTS state operation.

The proposed methods provide an average increase in efficiency from 16% to 23%, while ensuring high convergence of the obtained results at the level of 93.17%.

KEYWORDS

Reliability of technical systems, complex technical systems, efficiency of assessment, comprehensive assessment.

Organizational and technical systems (OTS), as a separate class of systems, are becoming more and more widespread regardless of the field of use and the tasks solved by them [1, 2].

However, for the correct and full application of OTS, it is necessary to use in their evaluation subsystem the appropriate mathematical and software that evaluates their condition and the very process of performing the tasks set by OTS [2].

The following tasks are distinguished among the tasks solved by the subsystems for assessing the OTS state [2, 3]:

– forecasting the trend of changes in the operation process of the OTS state;

– detection of deviations of OTS operation parameters at the initial stages;

– formation of trends of further deviation of the parameters of the OTS operation;

– detection of faulty OTS elements in real time;

– determination of control influences to bring the OTS state to nominal, etc.

Depending on the depth of available knowledge about the physical essence of the processes of changing the state of operation of OTS, different types of models are used: deterministic, probabilistic, fuzzy, etc.

A feature of the models of the first type is a single trajectory that determines the relationship between the OTS state and the nature of the deviation of its parameters from the nominal ones.

In the second case, the probabilistic properties of causation must be taken into account due to the hypothetical nature of the transformation operator. In the third case, it is necessary to operate with the concept of uncertainty when building a diagnostic model.

As the number of OTS elements (their constituent parts) increases, the difficulty of identifying the reasons for the deviation of their parameters from the nominal ones increases. This creates serious prerequisites for the use of neuro-fuzzy expert systems (NFES) in subsystems for assessing the OTS state.

The use of NFES in subsystems for assessing the OTS state provides support for decision-making by the persons who make them for decision-making, regardless of their level of training [4, 5].

In practice, two modes of NFES operation may be defined, when used in the subsystem for assessing their condition [6]:

1. The OTS is managed by the decision maker in such a way as to focus on specific anomalous deviations in their condition.

2. The system continuously monitors the OTS state and gives recommendations to the decision-maker when there are grounds for this. Special methods and techniques are used in the NFES to address these issues [4, 5].

The NFES structure is usually considered as consisting of a database (DB), a knowledge base (KB) and some management system [3]. DB is a set of current states of OTS and observed signs. KB contains decision-making rules that combine basic fundamental knowledge in this subject area and heuristics obtained as a result of the activities of specialists. In addition, KB includes the concepts of classes and relationships in the specified subject area.

Taking into account the above, one of the options for increasing the effectiveness of the assessment of the OTS state is the improvement of existing (development of new) methods of assessing their state using neuro-fuzzy expert systems.

The analysis of works [9–74] showed that the common shortcomings of the above-mentioned studies are:

– assessment of the OTS state is carried out only at a separate level of their operation, or only at a separate element of OTS;

– with a comprehensive approach to the OTS assessment, as a rule, one or two components of the process of their operation are considered. This does not allow to fully assess the impact of management decisions on the further OTS operation;

– the approaches listed above (methods, techniques), provide weak integration into each other (or make it impossible at all), which does not allow them to be combined with each other for a joint assessment of the operation of the OTS state;

– the above approaches to assessing the OTS state operation use a different mathematical apparatus, which requires appropriate mathematical transformations, which in turn increase computational complexity and reduce the accuracy of assessing the OTS state operation, etc.

The purpose of the study is to develop methods for evaluating complex organizational and technical systems using the theory of artificial intelligence.

This will make it possible to obtain an assessment of the state of operation of complex OTS at different levels of their operation (separate elements of OTS) for the development of subsequent management decisions. This will make it possible to develop (improve) the software of modern and promising OTS by integrating this method into the corresponding software.

Complex organizational and technical systems are the object of the study. The problem that is solved in the study is an increase in the efficiency of the assessment of the operation process of complex organizational and technical systems while maintaining a given level of reliability.

The subject of the study is the process of evaluating complex organizational and technical systems using the theory of artificial intelligence.

The hypothesis of the study is the possibility of increasing the efficiency of the operation of complex organizational and technical systems while maintaining the given level of reliability of their assessment due to the development of a method for assessing the state of their operation.

5.1 DEVELOPMENT OF A METHOD FOR EVALUATING COMPLEX ORGANIZATIONAL AND TECHNICAL SYSTEMS USING THE THEORY OF ARTIFICIAL INTELLIGENCE

OTS should be considered as a complex dynamic system. A dynamic system can be in two states: stationary and non-stationary.

The stationarity of a dynamic system lies in the immutability of its parameters and structure, but under the influence of disturbances that change its state, the OTS can turn into a non-stationary state.

The transition process determines the new steady state of the established OTS, which does not depend on the initial one. Bifurcation is a variant of the development of a situation where OTS moves from resilience to chaos [3, 4].

Thus, it is the task of finding anomalies during the OTS operation [5–8].

To solve the task of detecting the bifurcation point in the continuous process of OTS operation, it is necessary to evaluate the continuous flow of their state variables from sensors, as well as other sources of information extraction.

The evaluation is carried out at regular intervals Δt. The evaluation is carried out at regular intervals T values form multidimensional (D – measurable) a time series that reflects the dynamics of the OTS state operation.

OTS consists of a set of D sensors (sources of information). Thus, for any sensor $d = 1, 2, \ldots, D$ time series $y_1^d, y_2^d, \ldots, y_t^d$ it is a set of values of the OTS state y_t^d, what are measured at the moment of time t. Limitations in the form of upper ones are imposed on the values of these parameters y_u^d and lower ones y_l^d border.

As an OTS for simulation, the communication and informatization system of the operational grouping of troops (forces) has been adopted in this study. The operational group of troops (forces) was formed according to the state of martial law (typical state). Mode of operation of the communication and information systems system – defence operation.

A computational experiment of the proposed method was conducted in the Microsoft Visual Studio 2022 software environment (USA). The hardware of the research process is AMD Ryzen 5.

The method of evaluating complex organizational and technical systems using the theory of artificial intelligence structurally and logically consists of three main procedures that are performed sequentially:

– procedure for processing streaming data on the OTS state;

– the procedure for forming hypotheses about the reasons for deviations of OTS indicators from nominal ones;

– the procedure for forming the knowledge base of a neuro-fuzzy expert system.

Action 1. Entering initial data about the OTS and the conditions of its operation.

At this stage, the following initial data on OTS are entered:

– the number of component parts (communication nodes and dedicated means of communication) that are part of the OTS;

– the bandwidth of each element of the OTS (component communication and informatization system);

– the type of traffic transmitted by each element of the OTS;

– topology of placement of OTS elements on the terrain;

– the number of means of destructive influence on the OTS (in this case, the number of means of radio-electronic countermeasures and cyber warfare);

– frequency-energy characteristics of means of destructive influence on OTS (means of radio-electronic countermeasures);

– the type of means of fire damage that operate in the OTS lane;

– the number of means of fire damage that operate in the OTS lane;

– intensity of fire damage (applications (hit)/per hour) by each means of fire damage, etc.

To search for bifurcations of the OTS state, a flow data analysis procedure using a double sliding window is used, the essence of which is to check the stationarity conditions based on sample data for short time series [1, 2].

The procedure for processing streaming data on the OTS state consists of the following interrelated actions.

Action 2. Formation of the output time series for each sensor (sensor).

Forming the output time series of the size for a given sensor d

$$HY^{d}=\left[y_{1}^{d},y_{2}^{d},\ldots,y_{H}^{d}\right], \tag{5.1}$$

where H multiple N.

Action 3. Division of the obtained time series and their subsequent transformation.

Division of the obtained time series by N tuples size h. Receiving $k=\overline{1;N}$ time series of the species

$$Y^{d,k}=\left[y_{1}^{d,k},y_{2}^{d,k},\ldots,y_{h}^{d,k}\right]. \tag{5.2}$$

Action 4. Processing of received data tuples.

Processing of each received tuple $Y^{d,k}$ using the size sliding window algorithm l. At the output, it is possible to obtain a set of tuples of the form

$$Y^{d,k}=\left[y_{1}^{d,k},y_{2}^{d,k},\ldots,y_{h-l+1}^{d,k}\right]. \tag{5.3}$$

Action 5. Obtaining average values of data results and their squares.

Obtaining average values and squares of average values for each tuple $Y^{d,k}$. Formation of two tuples of the species

$$\left[\overline{y_{d,1}},\overline{y_{d,2}},\ldots,\overline{y_{d,k}},\ldots,\overline{y_{d,N}}\right] \text{ and } \left[\overline{y_{d,1}^{2}},\overline{y_{d,2}^{2}},\ldots,\overline{y_{d,k}^{2}},\ldots,\overline{y_{d,N}^{2}}\right]. \tag{5.4}$$

Action 6. Checking the obtained sequences for the presence of a trend.

To check the obtained sequences for the presence of a trend, this study uses a modification of the Foster-Steward criterion. For this, sets are calculated u_k, v_k, u_k^2 and v_k^2 according to formulas:

$$u_{k}=\begin{cases}1\leftarrow if\,\overline{y_{k}}>\overline{y_{k-1}},\overline{y_{k-2}},\ldots,\overline{y_{1}}\\ 0\leftarrow \text{else}\end{cases}, \tag{5.5}$$

$$v_{k}=\begin{cases}1\leftarrow if\,\overline{y_{k}}<\overline{y_{k-1}},\overline{y_{k-2}},\ldots,\overline{y_{1}}\\ 0\leftarrow \text{else}\end{cases}, \tag{5.6}$$

$$u_{k}^{2}=\begin{cases}1\leftarrow if\,\overline{y_{k}^{2}}>\overline{y_{k-1}^{2}},\overline{y_{k-2}^{2}},\ldots,\overline{y_{1}^{2}}\\ 0\leftarrow \text{else}\end{cases}, \tag{5.7}$$

$$v_{k}^{2}=\begin{cases}1\leftarrow if\,\overline{y_{k}^{2}}<\overline{y_{k-1}^{2}},\overline{y_{k-2}^{2}},\ldots,\overline{y_{1}^{2}}\\ 0\leftarrow \text{else}\end{cases}. \tag{5.8}$$

Action 7. Non-stationarity hypothesis testing.

The next stage for testing the hypothesis of the absence of stationarity in time series is two statistics:

$$W=\sum_{k=2}^{N}\left(u_k-v_k\right), \tag{5.9}$$

$$F=\sum_{k=2}^{N}\left(u_k+v_k\right), \tag{5.10}$$

and similarly, for squares:

$$W^2=\sum_{k=2}^{N}\left(u_k^2-v_k^2\right), \tag{5.11}$$

$$F^2=\sum_{k=2}^{N}\left(u_k^2+v_k^2\right). \tag{5.12}$$

Action 8. Definition of values t_W, t_F, t_{W^2} and t_{F^2} by formulae

$$t_W=\frac{W}{\sigma_W},\ \ t_{W^2}=\frac{W^2}{\sigma_W},\ \ t_F=\frac{F-\mu}{\sigma_F},\ \ t_{F^2}=\frac{F^2-\mu}{\sigma_F}, \tag{5.13}$$

where

$$\sigma_W=\left(2\times\sum_{k=2}^{N}\frac{1}{k}\right)^{0,5},\ \ \sigma_F=\left(\mu-4\times\sum_{k=2}^{N}\frac{1}{k^2}\right)^{0,5},\ \ \mu=2\times\sum_{k=2}^{N}\frac{1}{k}. \tag{5.14}$$

Action 9. Description of normalized values and their comparison with nominal ones.

In the absence of a trend, the normalized values of statistics are roughly described by the Student distribution with the number of degrees of freedom *df=N*. The obtained values are compared with the calculated values module t_W, t_F, t_{W^2} and t_{F^2} and if the obtained values are exceeded, the transition of the process to a non-stationary state is recorded.

The second main procedure of this method is the procedure for forming hypotheses about the reasons for deviations of OTS indicators from nominal ones. The Dempster-Schafer theory [2] is a general framework for decision-making with uncertainty and allows evidence from different sources to be combined and to arrive at a certain degree of confidence in the presence of one event or another.

Action 10. Analysis of diagnostic variables about the state of operation of OTS. The analysis of the specified data consists in the formation of hypotheses about the causes of the pre-emergency OTS state using the theory of evidence.

This uses data obtained using the double sliding window algorithm and a matrix of fuzzy expert evaluations

$$\Lambda = \begin{matrix} & A_1 & A_2 & \cdots & A_r \\ d_1 & m_{11} & m_{12} & \cdots & m_{1r} \\ d_2 & m_{21} & m_{22} & \cdots & m_{2r} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ d_n & m_{n1} & m_{n2} & \cdots & m_{nr} \end{matrix}, \tag{5.15}$$

where r – the number of possible hypotheses; n – the number of diagnostic indicators to be analyzed; d – diagnostic indicator; A – hypothesis; m – expert assessment.

Action 11. Formation of a hypothesis about the OTS state.

To form hypotheses, it is necessary to perform the following steps.

Action 11.1. Selection from the matrix Λ only those lines d_n, in the streaming data of which bifurcations were found.

Action 11.2. Calculation of indicator functions P_n.

The measured diagnostic variable about the state is presented in the form of an interval number $D_n = \left[\underline{d_n}, \overline{d_n}\right]$, where $\underline{d_n}$ – lower limit, $\overline{d_n}$ – upper limit. Range of normative values of diagnostic variables $S_n = \left[\underline{\delta_n}, \overline{\delta_n}\right]$, where $\underline{\delta_n}$ – lower limit, $\overline{\delta_n}$ – upper limit.

For the case when the crisis OTS state occurs when the interval of the measured diagnostic variable is released $D_n = \left[\underline{d_n}, \overline{d_n}\right]$ beyond the upper limit of the range of its normative values $\overline{\delta_n}$ the following indicator function is used:

$$P_n = \begin{cases} 0, \text{ if } \overline{d_n} \le \overline{\ddot{a}_n}, \\ 1, \text{ if } \underline{d_n} \le \overline{\ddot{a}_n}, \\ \dfrac{\overline{d_n} - \overline{\ddot{a}_n}}{\overline{d_n} - \underline{d_n}}, \text{ if } \underline{d_n} < \overline{\ddot{a}_n} < \overline{d_n}. \end{cases} \tag{5.16}$$

For the case when the crisis OTS state occurs at the output of the interval D_n at the lower end of the range of its normative values $\underline{\delta_n}$ the following indicator function is used:

$$P_n = \begin{cases} 0, \text{ if } \underline{d_n} \le \underline{\ddot{a}_n}, \\ 1, \text{ if } \overline{d_n} \le \underline{\ddot{a}_n}, \\ \dfrac{\underline{\ddot{a}_n} - \underline{d_n}}{\overline{d_n} - \underline{d_n}}, \text{ if } \underline{d_n} < \underline{\ddot{a}_n} < \overline{d_n}. \end{cases} \tag{5.17}$$

Action 11.3. Calculation of normalized values of basic probabilities using the formula

$$\tilde{m}_{nr} = \frac{m_{nr}}{\sum_{i=1}^{r} m_{ni}}. \quad (5.18)$$

Action 11.4. Redistribution of probability values.

Then, using the value of the indicator function, the probability values are redistributed using the formulas

$$m_{nr} = m_{nr} \times P_n \text{ and } m_{n*} = 1 - P_n. \quad (5.19)$$

Action 11.5. Combining hypotheses.

Evidence theory is used to combine several hypotheses. To combine different evidence with probability distributions m_1 and m_2 in favor of one hypothesis, the Dempster-Schafer rule is used

$$m_1 \oplus m_2(A) = \frac{1}{1 - M(\varnothing)} \times \sum_{Y \cap Z = A} m_1(Y) \times m_2(Z), \quad (5.20)$$

where

$$M(\varnothing) = \sum_{Y \cap Z = \varnothing} m_1(Y) \times m_2(Z). \quad (5.21)$$

Action 11.6. Determination of the degree of confidence and the degree of plausibility.

According to the theory of evidence, estimates of the degree of confidence are determined $Bel(A_r)$ and the degree of plausibility $Pl(A_r)$ acceptance of hypotheses using formulas:

$$Bel(A_r) = \sum \{ m_n(C) \mid C \subseteq A_r \}, \quad (5.22)$$

$$Pl(A_r) = 1 - Bel(\overline{A_r}) = 1 - \sum \{ m_n(C) \mid C \cap A_r \neq \varnothing \}, \quad (5.23)$$

where C – a set of events.

Trust functions are calculated based on the obtained basic probabilities $Bel(A_r)$ and plausibility $Pl(A_r)$ for all analyzed hypotheses, and the most likely one is determined.

The final procedure in this method is the procedure for forming the knowledge base of a neuro-fuzzy expert system.

Action 12. Primary customization of the knowledge base using an improved genetic algorithm.

With the improved genetic algorithm proposed in study [19], the primary formation of the knowledge base takes place.

Action 13. Formation of a knowledge base about the OTS state.

At the specified stage, knowledge bases about the OTS state are formed on the basis of expressions (5.1)–(5.23). Formally, the model of the neuro-fuzzy knowledge (NFK) base of the OTS state can be written as follows (5.24)

$$\{P_n\} = \{\text{Rule}\}, \tag{5.24}$$

where Rule – rule of NFK.

Each NFK rule is defined as follows (5.25)

$$\text{Rule} = < C \to S >, \tag{5.25}$$

where C – condition of the rule on the OTS state; S – consequence of the rule on the OTS state.

A recursive mechanism for describing nodes and finite vertices of the OTS state decision tree was used. The condition parameter of the rule on the OTS state C defined as follows (5.26)

$$\tilde{N} = < C_l, R, C_r >, \tag{5.26}$$

where C_l – the left node of the condition of the OTS state rule; R – the relationship between the nodes of the OTS state rules; C_r – the right node of the condition of the OTS state rule.

Next, let's consider in detail the given parameters according to which the formation of the knowledge base about the OTS state is carried out:

$$C_l = FC_l \| \text{Null} \| C, \tag{5.27}$$

$$C_r = FC_r \| \text{Null} \| C, \tag{5.28}$$

where FC_l – the left final three of the condition of the rule about the OTS state; FC_r – the right final three conditions of the rule on the OTS state.

Expressions (5.27) and (5.28) make it possible to describe the conditions of OTS operation with different degrees of nesting:

$$FC_l = < L, Z, W >, \tag{5.29}$$

$$FC_r = < L, Z, W >, \tag{5.30}$$

where L – linguistic variable of the OTS state; Z – condition sign $Z = \{<, >, <=, >=, =, != \}$; W – the value of the condition of the OTS state, which is determined as follows (5.31)

$$W = L \| V, \tag{5.31}$$

where L – linguistic variable of the OTS state; V – fixed value (5.32)

$$V = T_i \| \text{const}, \tag{5.32}$$

where T_i – the value of a fuzzy variable from the term sets of a linguistic variable; const – constant.

This procedure allows the use of not only linguistic variables, but also classical variables. In this case, their value can also be compared with constants [3]. R – a set of possible relations between nodal vertices $R \subset \left(C_l \times C_r\right)$ or $R : C_l \to C_r$.

Similar to the parameter C the parameter is determined S – consequence of the OTS state rule

$$S = < S_l, R, S_r > , \tag{5.33}$$

where Sl – the left node of the consequence of the OTS state rule; R – the relationship between the nodes of the consequence of the OTS state rule; Sr – the right node of the consequence of the rule:

$$S_l = FS_l \| \text{Null} \| S, \tag{5.34}$$

$$S_r = FS_r \| \text{Null} \| S, \tag{5.35}$$

where FSl – the left final three consequence of the OTS state rule; FSr – the right final three consequence of the state rule of the OTS. Formulas (5.34) and (5.35) describe consequences with varying degrees of nesting:

$$FS_l = < L, \text{Op}, W >, \tag{5.36}$$

$$FS_r = < L, \text{Op}, W >, \tag{5.37}$$

where L – linguistic variable of the OTS state; Op – operation to assess the OTS state $\text{Op} = \{:=\}$; W – the meaning of the consequence of the rule on the OTS state.

Action 14. Determination of the amount of necessary computing resources for the assessment of the OTS state.

In order to prevent looping of calculations during calculations on actions 1–13 of this method, and to increase the efficiency of calculations, the load of computing resources is additionally determined. If the specified computational complexity threshold is exceeded, the number of software and hardware resources that must be additionally attracted is determined using the method proposed in work [19].

Action 15. Training of knowledge bases of agents of the improved genetic algorithm.

At this stage, knowledge bases of agents of the improved genetic algorithm are trained to increase its convergence. As a teaching method, the deep learning method proposed in the work [19] is used.

End.

To determine the effectiveness of the proposed method, a computational experiment of its work was conducted to solve the task of assessing the OTS state (state of the communication and informatization system) of the operational group of troops (forces).

Let *n*-number of rules in a neuro-fuzzy expert system, m_i – the number of conditions in *i*-th rules $\left(i=1,..,n\right)$, *k* – the number of different linguistic variables involved in the terms of the rules, t_i – the power of the term set *i*-th a linguistic variable involved in the conditions of the rules, *s* – the number of relationships between variables in conditions.

Separate parts of the computational experiment using the proposed method are given in **Tables 5.1** and **5.2**. The general computational experiment is laid out on more than 196 sheets, in this section only its final part is presented.

Table 5.1 The value of complexity estimates

	n	m_{ave}	k	t_{ave}	S	Classic NFS [19]	NFS with Rete [19]	NFS with Treat [19]	NFS with Rete II [19]	NFS with the proposed method
RB1	20	9	8	4	10	150	140	145	124	**98**
RB 2	400	9	8	4	10	1500	1420	1590	1280	**1020**
RB 3	800	9	8	4	10	2905	2737	2820	2350	**1916**
RB 4	1600	9	8	4	10	5726	5549	5666	4990	**4050**
RB 5	3200	9	8	4	10	11000	9568	9850	8540	**6354**
RB 6	6400	9	8	4	10	19738	17597	17966	15800	**12430**
RB 7	12800	9	8	4	10	37918	34679	35291	31560	**25660**
RB 8	25600	9	8	4	10	74008	70264	71292	61690	**49505**
RB 9	51200	9	8	4	10	140561	129170	133421	115000	**86594**
RB 10	102400	9	8	4	10	251007	217590	225666	180429	**134140**

Table 5.2 Comparative results of the process of assessing the OTS state

	With use method	Without use method
Efficiency of the process of assessing the state of the group		
Better case, sec.	39 – 203	56 – 507.1
Worse case, sec.	155.1 – 2501.5	482.8 – 5977
Reliability of the decisions received		
Better case, sec.	0.89 – 1.0	0.64 – 0.85
Worse case, sec.	0.8 – 1.0	0.617 – 0.75

From the analysis of **Table 5.1** and **5.2**, it can be concluded that the proposed method provides an increase in efficiency by an average of 23%, while ensuring a high convergence of the obtained results at the level of 93.17%.

The advantages of the proposed method of evaluating complex organizational and technical systems are as follows:

– full coverage of critical events occurring during the OTS operation (actions 9–11.6, expressions (5.15)–(5.23)). This is achieved due to the use of the Dempster-Schafer theory, which achieves the completeness of the assessment of the entire spectrum of critical events in the OTS, in comparison with works [2, 8];

– the ability to take into account the uncertainty about the received information from various sources of information about the OTS state (actions 9–11.6, expressions (5.15)–(5.23)). This is achieved due to the use of the Dempster-Schafer theory, which achieves the completeness of the assessment of the entire spectrum of critical events in the OTS, in comparison with works [1, 3];

– comprehensively describe the operation process of OTS (expressions (5.1)–(5.37)), compared to works [4, 6]. This makes it possible to increase the accuracy of OTS modeling for subsequent management decisions;

– allows to describe OTS in a dynamic form (expressions (5.1)–(5.37)), compared to works [7, 9];

– carry out initial adjustment of OTS knowledge bases using an improved genetic algorithm (action 12). This allows to reduce the computational complexity in the further formation of the OTS knowledge base by reducing the metric of rule formation in the OTS knowledge base in comparison with works [4, 7];

– conduct modeling of the nature of the development of atypical events in the OTS due to the use of time series (actions (2)–(11.6)), which achieves the possibility of developing preventive measures to minimize the impact of these events on the process of OTS operation, in comparison with works [5, 10];

– by gradually reducing the metric of the formation of a knowledge base about the states of OTS, due to the training of agents of the improved genetic algorithm (action 12). This makes it possible to reduce the number of computing resources of the subsystem for assessing the OTS state operation, in comparison with works [8, 11];

– the ability to work with opinions of experts of different physical origins and units of measurement, which achieves the elimination of the problem of dimensionality during the operation of the subsystem for assessing the OTS state (actions 11.5–11.6), in comparison with works [7, 13];

– increase the efficiency of obtaining an assessment of the OTS state (actions 1–15), due to the reduction of the decision space, compared to works [5, 14].

The disadvantages of the proposed method include:

– greater computational complexity of performing computational operations in OTS compared to known research;

– the need for additional calculations when working with data of various sizes.

The proposed method will allow to:

– simulate the process of OTS operation;

– determine effective measures to increase the efficiency of the assessment of the OTS state;

– reduce the use of computing resources of the OTS state assessment subsystem.

The limitations of the study are the need to take into account the delay time for collecting and proving information from OTS sensors (sensors).

The proposed method should be used as software for automated troop control systems such as "Dzvin-AS", "Oreanda-PS", as well as integrated information systems such as "Delta".

5.2 DEVELOPMENT OF A METHOD FOR DETECTING ANOMALIES IN ORGANIZATIONAL AND TECHNICAL SYSTEMS

The method of detecting anomalies in the OTS consists of the following sequence of actions:

Action 1. Entering output data.

At this stage, the initial data available about the OTS are entered, namely: the number and type of means that are part of it, the type of data circulating in the OTS, available computing resources, the number and type of connections between each element of the OTS, information on technical characteristics of control channels and data transmission, information on the application environment, etc.

Action 2. Verification of OTS parameters.

At this stage, the parameters of the OTS are verified with the help of a bio-inspired algorithm. In case of detection of deviations from the input data, the output data is adjusted using the results of the bio-inspired algorithm.

Action 3. Determination of destabilizing factors affecting OTS. In the specified action, the initial identification of attacks inherent in the OTS takes place:

$$CBT_{\mu}=\begin{cases}\left\langle F^{(i)}_{jL_{\mu}R_{\mu}},CBT_{L_{\mu}},CBT_{R_{\mu}}\right\rangle, & \text{if } \#\mu\geq 2,\\ \mu, & \text{if } \#\mu=1,\end{cases} \tag{5.38}$$

where $\mu=\{0,\ldots,m\}$ – the original set of anomaly class labels; $L_{\mu}\subsetneq\mu$ – an arbitrarily generated or defined subset; $\mu\left(\#L_{\mu}<\#\mu\right)$, $R\mu=\mu\backslash L\mu$ – left classification subtree; $CBT_{R_{\mu}}$ – right classification subtree; $F^{(i)}_{jL_{\mu}R_{\mu}}$ – a nodal detector trained on plural elements $\left\{\left(x_l,0\right)\middle|\overline{c_l}\in L_{\mu}\right\}_{l=1}^{M}\cup\left\{x_l,1\middle|\overline{c_l}\in R_{\mu}\right\}_{l=1}^{M}$.

Action 4. Initiation of an artificial immune system.

In the initialization step, a population of N candidate antibodies is randomly generated $Ab(t)=\{Ab_1(t), Ab_2(t),\ldots, Ab_N(t)\}$, where Ab_i — it's i-th agent (antibody) on t-th iterations. The affinity of these antibodies is assessed by the function Aff(). At each iteration, any antibody is cloned to form offspring, after which all clones except the parent undergo mutation. Only the clone with the highest affinity remains.

To improve the accuracy of solving calculation tasks and the speed of convergence, this study proposes an artificial immune system with deep learning mechanisms to solve the task of detecting anomalies in the OTS.

Action 5. Antibody pre-selection.

At this stage, an initial selection of antibodies to each of the swarms is carried out using an improved genetic algorithm proposed in work [20].

Action 6. Distribution of agents of the artificial immune system between swarms.

At the swarm renewal stage, antibodies that are candidates for the solution are redistributed between the elite swarm and the general swarm. Elite swarm antibodies undergo an affinity-dependent cloning operator and a self-learning mutation operator, where the search radius is updated adaptively using a specially designed mechanism.

Action 7. Antibody cloning. In this study, the number of clones created from one parent antibody is determined by its affinity.

The higher the affinity of the parent antibody, the more offspring it produces.

At the same time, the number of clones is a nonlinear function of the affinity of the parent antibody. The procedure for calculating the antibody cloning procedure is given below:

$$Aff_{\max} = \max\{Aff(Ab_i(t)),\ i = 1, 2,\dots, N), \tag{5.39}$$

$$Aff_{\min} = \min\{Aff(Ab_i(t)),\ i = 1, 2,\dots, N), \tag{5.40}$$

$$Aff^*(i) = (Aff(Ab_i(t)) - Aff_{\min})/(Aff_{\max} - Aff_{\min}), \tag{5.41}$$

$$Nc_i\,(t) = \mathrm{round}((N_{c\max} - N_{c\min})\ Aff(A_{bi}(t))\ n + N_{c\min}), \tag{5.42}$$

where $N_{c\max}$ and $N_{c\min}$ — maximum and minimum number of antibody descendants; n — power factor of control function. All antibodies, including generic and elite swarms, perform this cloning operator once on each iteration.

Action 8. Antibody mutation.

Elite swarm antibodies have higher affinity and play the role of memory cells responding much more aggressively and faster in the secondary immune response.

Consequently, these elite swarm antibodies play an important role in local search and undergo a self-learning mutation. On the other hand, general swarm antibodies play the role of global search led by elite swarm antibodies, so all general swarm antibodies undergo deep training to accelerate convergence.

If $Aff(Ab_i(t)) < Aff(Ab_i(t))$, then the affinity of one antibody of the general swarm is less than the affinity of an antibody selected from an elite swarm. In this case, the general swarm antibody is trained on the basis of the selected elite swarm antibody. Said case is described by the following mathematical expression

$$\Delta A_{bi}(t) = \mathrm{rand} * (A_{bj}(t) - A_{bi}(t)). \tag{5.43}$$

If $Aff(Ab_i(t)) < Aff(Ab_e(t))$ the affinity of one total swarm antibody is greater than that of the selected elite swarm antibody, but less than that of the best elite swarm antibody. Training in this case is described as follows

$$\Delta Ab_i(t) = \mathrm{rand} * (Ab_e(t) - Ab_i(t)). \tag{5.44}$$

Otherwise, the general swarm antibody performs a deep mutation

$$\Delta Ab_i(t) = \text{rand}n * \lambda_i(t), \tag{5.45}$$

where rand – a uniformly distributed random variable; rand$n$ – a normally distributed random variable with a mean of 0 and a standard deviation of 1; $Ab_e(t)$ – the best elite swarm antibody by affinity; $\lambda_i(t)$ – antibody search radius $Ab_i(t)$ for t-th iterations.

In this artificial immune system, the search radius is fixed, which impairs the speed of convergence and accuracy of the solution. This is because any elite antibody can easily go beyond the optimum if it is located close to it, but has too large a search radius.

On the other hand, if the search radius is too small, the rate of convergence of artificial immune systems is significantly reduced.

Thus, the search radius $\lambda_i(t)$ can be updated dynamically according to the following rule:

$$\lambda_i(t) = \begin{cases} \lambda_i(t-1), \text{if Aff}\left(Ab_i(t) > Ab_i(t-1)\right), \\ \dfrac{\lambda_i(t-1)}{2}, \text{ otherwise } \dfrac{\lambda_i(t-1)}{2} \geq \lambda_{\min}, \\ \lambda_0(k), \text{ otherwise,} \end{cases} \tag{5.46}$$

where $\lambda_{\min}$ – the lower limit of the search radius and is defined as:

$$\lambda_0(k) = \begin{cases} \dfrac{\lambda_0(k-1)}{2}, \text{if } \dfrac{\lambda_0(k-1)}{2} \geq \lambda_{\min}, \\ \lambda_0(0), \text{otherwise.} \end{cases} \tag{5.47}$$

For each antibody $Ab_i(t)$, according to (5.46) and (5.47), the initial search radius is given by λ_0, the value of which is equal to half of the threshold value $\text{Th}_s(t)$

$$\lambda_0(k) = \text{Th}_s/2. \tag{5.48}$$

Because after performing the suppression operator, the distance between any two antibodies exceeds $\text{Th}_s(t)$, value $\lambda_0 = \text{Th}_s(t)/2$ allows to cover the search space as much as possible without overlap.

If the affinity of the antibody $Ab_i(t)$ improves after mutation, its search radius $\lambda_i(t)$ stored. Otherwise $\lambda_i(t)$ halved:

$$\text{Aff}(Ab_i(t)) > \text{Aff}(Ab_i(t-1)), \tag{5.49}$$

$$\lambda_i(t) = \lambda_i(t-1). \tag{5.50}$$

However, the search radius cannot be less than $\lambda_{min}(t)$, because a search radius that is too small can significantly reduce the rate of convergence. Therefore, if $\lambda_i(t)$ becomes smaller than $\lambda_{min}(t)$, its value is set equal $\lambda_0(k)$, which decreases to half the previous value $\lambda_0(k\text{-}1)$. At the same time $\lambda_0(k)$ also cannot be less than $\lambda_{min}(t)$, and if it does, its value is reset $\lambda_0(k)$.

Action 9. Deep learning of the artificial immune system.

The learning mechanism of the artificial immune system is described as follows

$$p\left(Ab_j\left(t\right)\right)=\frac{Aff^*\left(Ab_j\left(t\right)\right)}{\sum_{\text{³}=1}^{N_{\text{àê³ò}}} Aff^*\left(Ab_i\left(t\right)\right)}. \tag{5.51}$$

Obviously, the probability of selection is uneven: the higher the affinity of the antibody, the greater the probability of its selection, so the roulette method is used to select an elite swarm antibody for training.

Let's suppose that the optimal values of the multimodal function differ significantly in affinity. In this case, antibodies with higher affinity are more likely to evolve into the global optimum, so this learning algorithm will demonstrate faster convergence.

Action 10. Antibody suppression.

The proposed artificial immune system uses a dynamic mechanism of suppression, which is described by the following mathematical expressions:

$$D_{max} = \max\{ D_{i,j}(t)|i,j = 1,2,\ldots, N,i \neq j), \tag{5.52}$$

$$D_{min} = \min\{D_{i,j}(t)|i,j = 1,2,\ldots, N,i \neq j), \tag{5.53}$$

$$\text{Th}_s(t) = D_{min} + \xi(D_{max} - D_{min}), \tag{5.54}$$

where $\text{Th}_s(t)$ – a threshold that is proportional to the similarity of the antibody population; Di,j – euclidean distance between i-м and j-th antibodies on t-th iterations; $\xi \in (0, 1)$ – control parameter. After the application of the suppression operator, a certain amount of randomly generated antibodies is added to the population to maintain its size at N.

Action 11. Swarm update.

As is known, some general swarm antibodies can achieve higher affinity than elite swarm antibodies due to the elitist learning mechanism.

Accordingly, the composition of swarms must be reviewed, allowing the preferred antibodies of the total swarm to transition to the elite swarm. During swarm renewal, all antibodies are sorted by affinity in descending order. Then the first *N*elite antibodies move to the elite swarm, while the rest remain in the general swarm. It is important to note that after initial initialization, all antibodies undergo a deep learning mechanism until the suppression operator is triggered. This enables all antibodies to evolve in a fairly small area around them, which helps in the search for local extrema.

Action 12. Determination of the amount of necessary computing resources, intelligent decision support system.

In order to prevent looping of calculations on actions 1–11 of this method, and to increase the efficiency of calculations, the system load is additionally determined. If the specified computational complexity threshold is exceeded, the number of software and hardware resources that must be additionally attracted is determined using the method proposed in work [20].

The end of the algorithm.

The effectiveness of the proposed method of detecting anomalies in OTS based on artificial intelligence technologies is given in the **Table 5.3.**

Table 5.3 Evaluation of the effectiveness of the proposed method of detecting anomalies in OTS

Algorithm name	Accuracy	Convergence	Efficiency, sec	Percentage of system resources deployed
[6]	77.92%	80.23%	5.23E + 04	100
[7]	75.71%	77.4%	3.72E + 04	100
[8]	77.01%	76.66%	6.40E + 02	100
[9]	76.17%	81.15%	5.06E + 01	100
[10]	80.31%	80.67%	7.07E + 02	100
[11]	70.05%	81.03%	6.16E + 03	100
[12]	70.28%	75.18%	5.19E + 03	100
[13]	75.24%	73.12%	7.04E + 02	100
[14]	77.41%	74.2%	7.55E + 07	100
[15]	75.16%	74.28%	5.42E + 05	100
[16]	81.44%	85.9%	5.56E + 04	100
Proposed method	**97.3%**	**95.23%**	**5.23E + 04**	**80**

From the analysis of **Table 5.3**, it can be concluded that the proposed method provides an increase in accuracy by an average of 16%, an increase in efficiency by an average of 12%, while ensuring a high convergence of the obtained results at the level of 95.23%.

The advantages of the proposed method are due to the following:

– verifies OTS parameters (action 2) using an improved bat flock algorithm, compared to works [6–10]. This allows to minimize the error of entering incorrect data for the operation of data on the OTS state;

– there is a primary identification of the anomalies that are inherent in the specified OTS using the classification tree (action 3), compared to works [7, 9];

– the possibility of adaptation to the type and duration of anomalies due to multi-level adaptation of the artificial immune system (actions 1–12), compared to works [8, 12];

– primary selection of antibodies to each of the swarms of the artificial immune system is carried out using an improved genetic algorithm (action 5), compared to works [11, 13];

– the ability to train general swarm antibodies with elite swarm antibodies, which ensures the possibility of deep learning (action 9), compared to works [9, 12];

– to perform replacement of unsearchable individuals by updating the antibody population (actions 11), compared to works [9, 16];

– the ability to simultaneously find a solution in different directions (actions 1–12, **Table 5.3**);

– by the possibility of calculating the required number of computing resources to be attracted in case of impossibility of making calculations with available computing resources (action 12), compared to works [9, 13].

The disadvantages of the proposed method include:

– less accuracy of evaluation by a single anomaly evaluation parameter;

– loss of validity of received solutions when finding a solution in several directions at the same time;

– lower evaluation accuracy compared to other anomaly detection methods.

The specified method will allow to:

– determine the optimal indicator of anomaly detection depending on the OTS in which it is used;

– identify effective measures to improve the effectiveness of anti-anomalies in OTS;

– increase the speed of processing heterogeneous data while ensuring the given reliability of decision-making during their processing;

– reduce the use of computing resources of decision support systems.

The limitations of the study are the need to take into account the delay time for collecting and proving information from the constituent parts of organizational and technical systems.

The proposed approach should be used to solve the task of managing complex technical systems characterized by a high degree of complexity.

CONCLUSIONS

The study proposes a method of evaluating complex organizational and technical systems using the theory of artificial intelligence. The novelty of the proposed method is:

– in full coverage of critical events occurring during the OTS operation. This is achieved due to the use of the Dempster-Schafer theory, which achieves the completeness of the assessment of the entire spectrum of critical events in the OTS;

– in a comprehensive description of the process of OTS operation. This makes it possible to increase the accuracy of OTS modeling for subsequent management decisions;

– in the description of OTS in a dynamic form;

– in the ability to carry out initial adjustment of OTS knowledge bases using an improved genetic algorithm. This allows to reduce the computational complexity during the further formation of the OTS knowledge base by reducing the metric of rule formation in the OTS knowledge base;

– in the ability to model the nature of the development of atypical events in the OTS due to the use of time series, which achieves the possibility of developing preventive measures to minimize the impact of the specified events on the process of OTS operation;

– in the gradual reduction of the metric of the formation of the knowledge base about the states of OTS, due to the training of agents of the improved genetic algorithm. This allows to reduce the number of computing resources of the subsystem for assessing the OTS state operation;

– in the ability to work with opinions of experts of different physical origins and units of measurement, which achieves the elimination of the problem of dimensionality during the operation of the subsystem for assessing the OTS state.

The proposed method provides an increase in efficiency by an average of 23%, while ensuring high convergence of the obtained results at the level of 93.17%, which is confirmed by the results of a numerical experiment.

The method implementation algorithm is defined, thanks to additional and improved procedures, which allows to:

– verify OTS parameters using an improved bat flock algorithm. This allows to minimize the error of entering incorrect data for the operation of data on the OTS state;

– perform the initial identification of anomalies that are inherent in the specified OTS using the classification tree;

– adapt to the type and duration of abnormalities due to multi-level adaptation of the artificial immune system;

– conduct initial selection of antibodies to each of the swarms of the artificial immune system using an improved genetic algorithm;

– train antibodies of the general swarm with antibodies of the elite swarm, which ensures the possibility of deep learning;

– replace unsearchable individuals by updating the antibody population;

– simultaneously search for a solution in different directions;

– calculate the required number of computing resources that must be attracted in case of impossibility of making calculations with available computing resources.

An example of the use of the proposed method was carried out using the example of detecting anomalies in the OTS, which showed an increase in accuracy by an average of 16%, an increase in efficiency by an average of 12%, while ensuring high convergence of the obtained results at the level of 95.23%.

CONFLICT OF INTEREST

The authors declare that they have no conflict of interest in relation to this research, whether financial, personal, authorship or otherwise, that could affect the research and its results presented in this paper.

USE OF ARTIFICIAL INTELLIGENCE

The authors confirm that they did not use artificial intelligence technologies in creating the submitted work.

REFERENCES

1. Dudnyk, V., Sinenko, Y., Matsyk, M., Demchenko, Y., Zhyvotovskyi, R., Repilo, I. et al. (2020). Development of a method for training artificial neural networks for intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 3 (2 (105)), 37–47. https://doi.org/10.15587/1729-4061.2020.203301
2. Sova, O., Shyshatskyi, A., Salnikova, O., Zhuk, O., Trotsko, O., Hrokholskyi, Y. (2021). Development of a method for assessment and forecasting of the radio electronic environment. EUREKA: Physics and Engineering, 4, 30–40. https://doi.org/10.21303/2461-4262.2021.001940
3. Pievtsov, H., Turinskyi, O., Zhyvotovskyi, R., Sova, O., Zvieriev, O., Lanetskii, B., Shyshatskyi, A. (2020). Development of an advanced method of finding solutions for neuro-fuzzy expert systems of analysis of the radioelectronic situation. EUREKA: Physics and Engineering, 4, 78–89. https://doi.org/10.21303/2461-4262.2020.001353
4. Zuiev, P., Zhyvotovskyi, R., Zvieriev, O., Hatsenko, S., Kuprii, V., Nakonechnyi, O. et al. (2020). Development of complex methodology of processing heterogeneous data in intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 4 (9 (106)), 14–23. https://doi.org/10.15587/1729-4061.2020.208554
5. Kuchuk, N., Mohammed, A. S., Shyshatskyi, A., Nalapko, O. (2019). The Method of Improving the Efficiency of Routes Selection in Networks of Connection with the Possibility of Self-Organization. International Journal of Advanced Trends in Computer Science and Engineering, 8 (1.2), 1–6.
6. Shyshatskyi, A., Zvieriev, O., Salnikova, O., Demchenko, Ye., Trotsko, O. and Neroznak, Ye. et al. (2020). Complex Methods of Processing Different Data in Intellectual Systems for Decision Support System. International Journal of Advanced Trends in Computer Science and Engineering, 9 (4), 5583–5590. https://doi.org/10.30534/ijatcse/2020/206942020
7. Pozna, C., Precup, R.-E., Horvath, E., Petriu, E. M. (2022). Hybrid Particle Filter—Particle Swarm Optimization Algorithm and Application to Fuzzy Controlled Servo Systems. IEEE Transactions on Fuzzy Systems, 30 (10), 4286–4297. https://doi.org/10.1109/tfuzz.2022.3146986
8. Yang, X.-S., Deb, S. (2013). Cuckoo search: recent advances and applications. Neural Computing and Applications, 24 (1), 169–174. https://doi.org/10.1007/s00521-013-1367-1
9. Mirjalili, S. (2015). The Ant Lion Optimizer. Advances in Engineering Software, 83, 80–98. https://doi.org/10.1016/j.advengsoft.2015.01.010
10. Yu, J. J. Q., Li, V. O. K. (2015). A social spider algorithm for global optimization. Applied Soft Computing, 30, 614–627. https://doi.org/10.1016/j.asoc.2015.02.014
11. Mirjalili, S., Mirjalili, S. M., Lewis, A. (2014). Grey Wolf Optimizer. Advances in Engineering Software, 69, 46–61. https://doi.org/10.1016/j.advengsoft.2013.12.007
12. Koval, V., Nechyporuk, O., Shyshatskyi, A., Nalapko, O., Shknai, O., Zhyvylo, Y. et al. (2023). Improvement of the optimization method based on the cat pack algorithm. Eastern-European Journal of Enterprise Technologies, 1 (9 (121)), 41–48. https://doi.org/10.15587/1729-4061.2023.273786
13. Gupta, E and Saxena, A (2015). "Robust generation control strategy based on Grey Wolf Optimizer". Journal of Electrical Systems. Vol. 11, pp.174–188.

14. Chaman-Motlagh, A. (2015). Superdefect Photonic Crystal Filter Optimization Using Grey Wolf Optimizer. IEEE Photonics Technology Letters, 27 (22), 2355–2358. https://doi.org/10.1109/lpt.2015.2464332
15. Nuaekaew, K., Artrit, P., Pholdee, N., Bureerat, S. (2017). Optimal reactive power dispatch problem using a two-archive multi-objective grey wolf optimizer. Expert Systems with Applications, 87, 79–89. https://doi.org/10.1016/j.eswa.2017.06.009
16. Koval, M., Sova, O., Orlov, O., Shyshatskyi, A., Artabaiev, Y., Shknai, O. et al. (2022). Improvement of complex resource management of special-purpose communication systems. Eastern-European Journal of Enterprise Technologies, 5 (9 (119)), 34–44. https://doi.org/10.15587/1729-4061.2022.266009
17. Ali, M., El-Hameed, M. A., Farahat, M. A. (2017). Effective parameters' identification for polymer electrolyte membrane fuel cell models using grey wolf optimizer. Renewable Energy, 111, 455–462. https://doi.org/10.1016/j.renene.2017.04.036
18. Zhang, S., Zhou, Y. (2017). Template matching using grey wolf optimizer with lateral inhibition. Optik, 130, 1229–1243. https://doi.org/10.1016/j.ijleo.2016.11.173
19. Khouni, S. E., Menacer, T. (2023). Nizar optimization algorithm: a novel metaheuristic algorithm for global optimization and engineering applications. The Journal of Supercomputing, 80 (3), 3229–3281. https://doi.org/10.1007/s11227-023-05579-4
20. Saremi, S., Mirjalili, S., Lewis, A. (2017). Grasshopper Optimisation Algorithm: Theory and application. Advances in Engineering Software, 105, 30–47. https://doi.org/10.1016/j.advengsoft.2017.01.004
21. Braik, M. S. (2021). Chameleon Swarm Algorithm: A bio-inspired optimizer for solving engineering design problems. Expert Systems with Applications, 174, 114685. https://doi.org/10.1016/j.eswa.2021.114685
22. Yapici, H., Cetinkaya, N. (2019). A new meta-heuristic optimizer: Pathfinder algorithm. Applied Soft Computing, 78, 545–568. https://doi.org/10.1016/j.asoc.2019.03.012
23. Duan, H., Qiao, P. (2014). Pigeon-inspired optimization: a new swarm intelligence optimizer for air robot path planning. International Journal of Intelligent Computing and Cybernetics, 7 (1), 24–37. https://doi.org/10.1108/ijicc-02-2014-0005
24. Yang, X.-S.; Durand-Lose, J., Jonoska, N. (Eds.) (2012). Flower Pollination Algorithm for Global Optimization. Unconventional Computation and Natural Computation, 240–249. https://doi.org/10.1007/978-3-642-32894-7_27
25. Gomes, G. F., da Cunha, S. S., Ancelotti, A. C. (2018). A sunflower optimization (SFO) algorithm applied to damage identification on laminated composite plates. Engineering with Computers, 35 (2), 619–626. https://doi.org/10.1007/s00366-018-0620-8
26. Mehrabian, A. R., Lucas, C. (2006). A novel numerical optimization algorithm inspired from weed colonization. Ecological Informatics, 1 (4), 355–366. https://doi.org/10.1016/j.ecoinf.2006.07.003
27. Qi, X., Zhu, Y., Chen, H., Zhang, D., Niu, B.; Huang, D. S., Jo, K. H., Zhou, Y. Q., Han, K. (Eds.) (2013). An Idea Based on Plant Root Growth for Numerical Optimization. Intelligent Computing Theories and Technology. Berlin, Heidelberg: Springer, 571–578. https://doi.org/10.1007/978-3-642-39482-9_66
28. Bezuhlyi, V., Oliynyk, V., Romanenko, I., Zhuk, O., Kuzavkov, V., Borysov, O. et al. (2021). Development of object state estimation method in intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 5 (3 (113)), 54–64. https://doi.org/10.15587/1729-4061.2021.239854

29. Mahdi, Q. A., Shyshatskyi, A., Prokopenko, Y., Ivakhnenko, T., Kupriyenko, D., Golian, V. et al. (2021). Development of estimation and forecasting method in intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 3 (9 (111)), 51–62. https://doi.org/10.15587/1729-4061.2021.232718
30. Sova, O., Radzivilov, H., Shyshatskyi, A., Shevchenko, D., Molodetskyi, B., Stryhun, V. et al. (2022). Development of the method of increasing the efficiency of information transfer in the special purpose networks. Eastern-European Journal of Enterprise Technologies, 3 (4 (117)), 6–14. https://doi.org/10.15587/1729-4061.2022.259727
31. Zhang, H., Zhu, Y., Chen, H. (2013). Root growth model: a novel approach to numerical function optimization and simulation of plant root system. Soft Computing, 18 (3), 521–537. https://doi.org/10.1007/s00500-013-1073-z
32. Labbi, Y., Attous, D. B., Gabbar, H. A., Mahdad, B., Zidan, A. (2016). A new rooted tree optimization algorithm for economic dispatch with valve-point effect. International Journal of Electrical Power & Energy Systems, 79, 298–311. https://doi.org/10.1016/j.ijepes.2016.01.028
33. Murase, H. (2000). Finite element inverse analysis using a photosynthetic algorithm. Computers and Electronics in Agriculture, 29 (1-2), 115–123. https://doi.org/10.1016/s0168-1699(00)00139-3
34. Zhao, S., Zhang, T., Ma, S., Chen, M. (2022). Dandelion Optimizer: A nature-inspired metaheuristic algorithm for engineering applications. Engineering Applications of Artificial Intelligence, 114, 105075. https://doi.org/10.1016/j.engappai.2022.105075
35. Paliwal, N., Srivastava, L., Pandit, M. (2020). Application of grey wolf optimization algorithm for load frequency control in multi-source single area power system. Evolutionary Intelligence, 15 (1), 563–584. https://doi.org/10.1007/s12065-020-00530-5
36. Dorigo, M., Blum, C. (2005). Ant colony optimization theory: A survey. Theoretical Computer Science, 344 (2-3), 243–278. https://doi.org/10.1016/j.tcs.2005.05.020
37. Poli, R., Kennedy, J., Blackwell, T. (2007). Particle swarm optimization: an overview. Swarm Intelligence, 1 (1), 33–57. https://doi.org/10.1007/s11721-007-0002-0
38. Bansal, J. C., Sharma, H., Jadon, S. S., Clerc, M. (2014). Spider Monkey Optimization algorithm for numerical optimization. Memetic Computing, 6 (1), 31–47. https://doi.org/10.1007/s12293-013-0128-0
39. Yeromina, N., Kurban, V., Mykus, S., Peredrii, O., Voloshchenko, O., Kosenko, V. et al. (2021). The Creation of the Database for Mobile Robots Navigation under the Conditions of Flexible Change of Flight Assignment. International Journal of Emerging Technology and Advanced Engineering, 11 (5), 37–44. https://doi.org/10.46338/ijetae0521_05
40. Maccarone, A. D., Brzorad, J. N., Stone, H. M. (2008). Characteristics and Energetics Of Great Egret and Snowy Egret Foraging Flights. Waterbirds, 31 (4), 541–549. https://doi.org/10.1675/1524-4695-31.4.541
41. Ramaji, I. J., Memari, A. M. (2018). Interpretation of structural analytical models from the coordination view in building information models. Automation in Construction, 90, 117–133. https://doi.org/10.1016/j.autcon.2018.02.025
42. Pérez-González, C. J., Colebrook, M., Roda-García, J. L., Rosa-Remedios, C. B. (2019). Developing a data analytics platform to support decision making in emergency and security management. Expert Systems with Applications, 120, 167–184. https://doi.org/10.1016/j.eswa.2018.11.023

43. Chen, H. (2018). Evaluation of Personalized Service Level for Library Information Management Based on Fuzzy Analytic Hierarchy Process. Procedia Computer Science, 131, 952–958. https://doi.org/10.1016/j.procs.2018.04.233
44. Chan, H. K., Sun, X., Chung, S.-H. (2019). When should fuzzy analytic hierarchy process be used instead of analytic hierarchy process? Decision Support Systems, 125, 113114. https://doi.org/10.1016/j.dss.2019.113114
45. Osman, A. M. S. (2019). A novel big data analytics framework for smart cities. Future Generation Computer Systems, 91, 620–633. https://doi.org/10.1016/j.future.2018.06.046
46. Nechyporuk, O., Sova, O., Shyshatskyi, A., Kravchenko, S., Nalapko, O., Shknai, O. et al. (2023). Development of a method of complex analysis and multidimensional forecasting of the state of intelligence objects. Eastern-European Journal of Enterprise Technologies, 2 (4 (122)), 31–41. https://doi.org/10.15587/1729-4061.2023.276168
47. Merrikh-Bayat, F. (2015). The runner-root algorithm: A metaheuristic for solving unimodal and multimodal optimization problems inspired by runners and roots of plants in nature. Applied Soft Computing, 33, 292–303. https://doi.org/10.1016/j.asoc.2015.04.048
48. Poliarush, O., Krepych, S., Spivak, I. (2023). Hybrid approach for data filtering and machine learning inside content management system. Advanced Information Systems, 7 (4), 70–74. https://doi.org/10.20998/2522-9052.2023.4.09
49. Balochian, S., Baloochian, H. (2019). Social mimic optimization algorithm and engineering applications. Expert Systems with Applications, 134, 178–191. https://doi.org/10.1016/j.eswa.2019.05.035
50. Lenord Melvix, J. S. M. (2014). Greedy Politics Optimization: Metaheuristic inspired by political strategies adopted during state assembly elections. 2014 IEEE International Advance Computing Conference (IACC). IEEE, 1157–1162. https://doi.org/10.1109/iadcc.2014.6779490
51. Moosavian, N., Roodsari, B. K. (2014). Soccer League Competition Algorithm, a New Method for Solving Systems of Nonlinear Equations. International Journal of Intelligence Science, 4 (1), 7–16. https://doi.org/10.4236/ijis.2014.41002
52. Hayyolalam, V., Pourhaji Kazem, A. A. (2020). Black Widow Optimization Algorithm: A novel meta-heuristic approach for solving engineering optimization problems. Engineering Applications of Artificial Intelligence, 87, 103249. https://doi.org/10.1016/j.engappai.2019.103249
53. Abualigah, L., Yousri, D., Abd Elaziz, M., Ewees, A. A., Al-qaness, M. A. A., Gandomi, A. H. (2021). Aquila Optimizer: A novel meta-heuristic optimization algorithm. Computers & Industrial Engineering, 157, 107250. https://doi.org/10.1016/j.cie.2021.107250
54. Hodlevskyi, M., Burlakov, G. (2023). Information technology of quality improvement planning of process subsets of the spice model. Advanced Information Systems, 7 (4), 52–59. https://doi.org/10.20998/2522-9052.2023.4.06
55. Askari, Q., Younas, I., Saeed, M. (2020). Political Optimizer: A novel socio-inspired meta-heuristic for global optimization. Knowledge-Based Systems, 195, 105709. https://doi.org/10.1016/j.knosys.2020.105709
56. Mohamed, A. W., Hadi, A. A., Mohamed, A. K. (2019). Gaining-sharing knowledge based algorithm for solving optimization problems: a novel nature-inspired algorithm. International Journal of Machine Learning and Cybernetics, 11 (7), 1501–1529. https://doi.org/10.1007/s13042-019-01053-x

57. Gödri, I., Kardos, C., Pfeiffer, A., Váncza, J. (2019). Data analytics-based decision support workflow for high-mix low-volume production systems. CIRP Annals, 68 (1), 471–474. https://doi.org/10.1016/j.cirp.2019.04.001
58. Harding, J. L. (2013). Data quality in the integration and analysis of data from multiple sources: some research challenges. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XL-2/W1, 59–63. https://doi.org/10.5194/isprsarchives-xl-2-w1-59-2013
59. Orouskhani, M., Orouskhani, Y., Mansouri, M., Teshnehlab, M. (2013). A Novel Cat Swarm Optimization Algorithm for Unconstrained Optimization Problems. International Journal of Information Technology and Computer Science, 5 (11), 32–41. https://doi.org/10.5815/ijitcs.2013.11.04
60. Karaboga, D., Basturk, B. (2007). A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. Journal of Global Optimization, 39 (3), 459–471. https://doi.org/10.1007/s10898-007-9149-x
61. Voznytsia, A., Sharonova, N., Babenko, V., Ostapchuk, V., Neronov, S., Feoktystov, S. et al. (2025). Development of methods for intelligent assessment of parameters in decision support systems. Eastern-European Journal of Enterprise Technologies, 4 (4 (136)), 73–82. https://doi.org/10.15587/1729-4061.2025.337528
62. Sova, O., Radzivilov, H., Shyshatskyi, A., Shvets, P., Tkachenko, V., Nevhad, S. et al. (2022). Development of a method to improve the reliability of assessing the condition of the monitoring object in special-purpose information systems. Eastern-European Journal of Enterprise Technologies, 2 (3 (116)), 6–14. https://doi.org/10.15587/1729-4061.2022.254122
63. Litvinenko, O., Kashkevich, S., Shyshatskyi, A., Dmytriieva, O., Neronov, S., Plekhova, G. et al.; Shyshatskyi, A. (Ed.) (2024). Information and control systems: modelling and optimizations. Kharkiv: TECHNOLOGY CENTER PC, 180. https://doi.org/10.15587/978-617-8360-04-7
64. Voznytsia, A., Sharonova, N., Babenko, V., Ostapchuk, V., Neronov, S., Feoktystov, S. et al. (2025). Development of methods for intelligent assessment of parameters in decision support systems. Eastern-European Journal of Enterprise Technologies, 4 (4 (136)), 73–82. https://doi.org/10.15587/1729-4061.2025.337528
65. Nalapko, O., Shyshatskyi, A., Ostapchuk, V., Mahdi, Q. A., Zhyvotovskyi, R., Petruk, S. et al. (2021). Development of a method of adaptive control of military radio network parameters. Eastern-European Journal of Enterprise Technologies, 1 (9 (109)), 18–32. https://doi.org/10.15587/1729-4061.2021.225331
66. Kalantaievska, S., Pievtsov, H., Kuvshynov, O., Shyshatskyi, A., Yarosh, S., Gatsenko, S. et al. (2018). Method of integral estimation of channel state in the multiantenna radio communication systems. Eastern-European Journal of Enterprise Technologies, 5 (9 (95)), 60–76. https://doi.org/10.15587/1729-4061.2018.144085
67. Alieinykov, I., Thamer, K. A., Zhuravskyi, Y., Sova, O., Smirnova, N., Zhyvotovskyi, R. et al. (2019). Development of a method of fuzzy evaluation of information and analytical support of strategic management. Eastern-European Journal of Enterprise Technologies, 6 (2 (102)), 16–27. https://doi.org/10.15587/1729-4061.2019.184394

68. Koshlan, A., Salnikova, O., Chekhovska, M., Zhyvotovskyi, R., Prokopenko, Y., Hurskyi, T. et al. (2019). Development of an algorithm for complex processing of geospatial data in the special-purpose geoinformation system in conditions of diversity and uncertainty of data. Eastern-European Journal of Enterprise Technologies, 5 (9 (101)), 35–45. https://doi.org/10.15587/1729-4061.2019.180197
69. Sova, O., Zhuravskyi, Y., Vakulenko, Y., Shyshatskyi, A., Salnikova, O., Nalapko, O. (2022). Development of methodological principles of routing in networks of special communication in conditions of fire storm and radio-electronic suppression. EUREKA: Physics and Engineering, 3, 159–166. https://doi.org/10.21303/2461-4262.2022.002434
70. Koval, M., Sova, O., Orlov, O., Shyshatskyi, A., Artabaiev, Y., Shknai, O. et al. (2022). Improvement of complex resource management of special-purpose communication systems. Eastern-European Journal of Enterprise Technologies, 5 (9 (119)), 34–44. https://doi.org/10.15587/1729-4061.2022.266009
71. Fedoriienko, V., Koshlan, O., Kravchenko, S., Shyshatskyi, A., Vasiukova, N., Trotsko, O. et al. (2021). Development of a methodological approach for processing different types of data in systems of special purpose. Technology Audit and Production Reserves, 6 (2 (62)), 18–24. https://doi.org/10.15587/2706-5448.2021.243950
72. Abed, A. A., Repilo, I., Zhyvotovskyi, R., Shyshatskyi, A., Hohoniants, S., Kravchenko, S. et al. (2021). Improvement of the method of estimation and forecasting of the state of the monitoring object in intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 4 (3 (112)), 43–55. https://doi.org/10.15587/1729-4061.2021.237996
73. Shyshatskyi, A., Zhuravskyi, Y., Plekhova, G., Shostak, I., Feoktystova, O., Dmytriieva, O. et al. (2025). Development of a polymodel resource management complex for intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 5 (4 (137)), 41–63. https://doi.org/10.15587/1729-4061.2025.340387
74. Shyshatskyi, A., Plekhova, G., Odarushchenko, E., Miahkykh, H., Feoktystova, O., Shostak, I. et al. (2025). Development of a polymodel complex of information systems resource management. Eastern-European Journal of Enterprise Technologies, 4 (4 (136)), 58–72. https://doi.org/10.15587/1729-4061.2025.335688

DOI: 10.15587/978-617-8360-23-8.CH6

6 DEVELOPMENT OF A SCIENTIFIC AND METHODOLOGICAL APPARATUS FOR ENSURING THE FUNCTIONAL RELIABILITY OF SPECIAL-PURPOSE INFORMATION SYSTEMS

Oleh Shknai, Illia Dmytriiev, Oleg Sova, Andrii Shyshatskyi, Olesia Zhuk, Bohdan Molodetskyi

ABSTRACT

The object of research is special-purpose information systems (IS). The problem addressed in the study is the improvement of the functional reliability of special-purpose IS. The development of a scientific and methodological apparatus for providing a functional special-purpose IS was carried out. The originality of the research consists of:

– systematic assessment of the state of functional reliability of special-purpose IS using the proposed principles of its provision;

– construction of multidimensional dependencies of the state of functional reliability of the special-purpose IS, which achieves an assessment of the functional reliability of the IS based on an arbitrary number of indicators;

– in the assessment of the functional reliability of special-purpose IS using the joint use of measurement data and fuzzy expert assessments, which solves the problem of dimensionality;

– in the construction of the time dependence of changes in indicators that characterize the state of functional reliability of special-purpose IS, which allows determining the moments of deviation of their values from the nominal ones.

In the assessment of the functional reliability of information services based on the concept of profiles, which achieves the possibility of decentralized influence on the special-purpose IS to increase its functional reliability.

In reducing uncertainty about the state of functional reliability of special-purpose IS, due to the use of an appropriate approach in the method of assessing the functional reliability of information services based on the concept of profiles.

The proposed scientific and methodological apparatus provides an increase in the efficiency of assessing the functional reliability of the IS by an average of 40%, while ensuring high reliability of the obtained results at the level of 92%, which is confirmed by the results of a numerical experiment.

KEYWORDS

Multidimensionality of assessment, complex systems, efficiency, reliability, complex assessment, methodology.

Special-purpose information systems (IS), as a distinct class of complex systems, are becoming increasingly widespread, regardless of their field of use and the tasks they solve [1, 2].

In the conditions of military conflicts of various levels, special-purpose ISs have become widespread to solve a wide range of tasks, such as [1]:

– collecting, processing, and summarizing the information circulating in them;

– storage of various types of data, their archiving, and output;

– solving individual and/or complex calculation tasks for a wide range of users;

– modeling the nature of military conflicts;

– transmission of information between IS elements, etc.

At the same time, considering the specific tasks performed by special-purpose ISs, higher requirements are placed on the hardware and software components compared to ISs that perform the same tasks in the interests of general users.

This is because special-purpose ISs function under a wide range of destabilizing factors, such as:

– the influence of systems and means of radio-electronic countermeasures and cyber warfare;

– fire damage to special-purpose IS elements by various means of damage;

– aggressive influence of climatic and mechanical factors that significantly affect;

– specific methods of using special-purpose IS.

One of the conceptual requirements that significantly distinguishes special-purpose IS is the functional reliability requirements imposed on them.

Taking into account the above, one of the options for increasing the effectiveness of the assessment of the IS state is the development of a scientific and methodological apparatus for ensuring the functional reliability of special-purpose information systems.

The analysis of works [9–74] showed that the common shortcomings of the above-mentioned studies are:

– assessment of the IS functional reliability state is carried out only at a separate level of their functioning, or only at a separate element of the special-purpose IS;

– with a comprehensive approach to assessing the IS functional reliability, as a rule, one or two components of the process of their functioning are considered. This does not allow to fully assess the impact of management decisions on the further functioning of the special-purpose IS;

– the approaches listed above (methods, techniques), provide weak integration into each other (or make it impossible at all), which does not allow them to be combined with each other for a joint assessment of the functioning of the IS state;

– the above approaches for assessing the state of IS functioning use a different mathematical apparatus, which requires appropriate mathematical transformations, which in turn increase computational complexity and reduce the accuracy of assessing the state of IS functional reliability, etc.

The aim of the study is to develop a scientific and methodological apparatus for ensuring their functional reliability. This will make it possible to comprehensively and multidimensionally assess the functional reliability of special-purpose ISs at different levels of their functioning (separate elements of special-purpose ISs) for the development of subsequent management decisions. Also, it will make it possible to develop (improve) the software of modern and promising IS systems by integrating the proposed conceptual foundations into the corresponding software.

To achieve the goal, the following tasks were set:

– to develop a scientific and methodological apparatus for ensuring the functional reliability of special-purpose information systems;

– to evaluate the effectiveness according to the defined criterion of functional reliability of special-purpose IS.

The object of the study is a special-purpose IS. The problem addressed in the study is the improvement of the functional reliability of special-purpose IS. The subject of the study is the process of assessing the functional reliability of special-purpose IS. The hypothesis of the study is the possibility of increasing the functional reliability of special-purpose IS due to the development of a scientific and methodological apparatus for ensuring their functional reliability.

In the course of the study, the following research methods were used:

– a general scientific method of analysis – for decomposing problematic issues of assessing the functional reliability of special-purpose IS when they perform tasks as intended. Also, the general scientific method of analysis is used to determine the advantages and disadvantages of known approaches to assessing the functional reliability of special-purpose IS when they perform tasks as intended;

– general scientific method of synthesis – to substantiate the most appropriate approaches to assessing the functional reliability of special-purpose IS when they perform tasks as intended;

– regression methods – to describe the dynamics of changes in the functional reliability of special-purpose IS. The specified approach was used to determine the regularity of changes in the state of functional reliability of special-purpose IS.

Theory of fuzzy sets – for multidimensional assessment of the functional reliability of special-purpose IS.

The IC of the communication and informatization system of the operational grouping of troops (forces) was adopted as a special-purpose IS for modeling in this study. The operational group of troops (forces) was formed according to the state of martial law (typical state). The mode of operation of the IS system of communication and information systems – defense operation. A computational experiment of the proposed methods was conducted, which is an integral part of the scientific and methodological apparatus in the Microsoft Visual Studio 2022 software environment (USA). The hardware of the research process is AMD Ryzen 5.

6.1 SCIENTIFIC AND METHODOLOGICAL APPARATUS FOR ENSURING FUNCTIONAL RELIABILITY OF SPECIAL-PURPOSE IS

PRINCIPLES AND APPROACHES OF ENSURING THE FUNCTIONAL RELIABILITY OF INFORMATION SYSTEMS

Ensuring the functional reliability of special-purpose IS must be considered from the point of view of managing complex systems. Consideration of the functional reliability of IS as a type of complex system creates a methodological basis for scientifically based adaptation of already known approaches, principles, methods, and models. These approaches have proven themselves well when solving the tasks of managing complex systems in other fields of use, in the field of ensuring the functional reliability of IS.

In this section, a system of principles is formed, which forms the basis of the proposed scientific and methodological apparatus for ensuring the functional reliability of special-purpose IS. These principles are necessary for the formation of theoretical developments, a system of views, and the selection of possible ways of informational support for ensuring the functional reliability of special-purpose IS at the stage of preparing them for application.

In accordance with the substantive characteristics and purpose, it is appropriate to classify all the principles applied in the field of ensuring functional reliability into two main groups:

– system-wide principles;

– principles of ensuring functional reliability of special-purpose IS.

System-wide principles for ensuring functional reliability:

1. The principle of systematicity. This principle is basic in the group of system-wide principles, as it reflects the fundamental provisions of the system approach used in the study. The system for ensuring functional reliability is considered simultaneously as a complex subject-centric system, as well as a separate subsystem of the information support system of special-purpose ISs in their preparation for use.

2. The principle of polymorphism. The principle of polymorphism emphasizes the multiplicity of forms of special-purpose IS elements and the variety of connections between them. The implementation of this principle allows describing the same system using different architectural models reflecting different approaches to ensure functional reliability, including scenario analysis as well as the concept of "barrier thinking".

The antipode of the principle of polymorphism is the principle of isomorphism. It assumes the existence of structural and characteristic similarities between ISs of different origins. The essence of this principle converges to the fact that the uniformity of forms of description of IS of different origins, the similarity of structures and properties of system characteristics, determines the possibility of using universal formalized methods of analysis based on the use of symbolic models. In the context of the tasks of ensuring the functional reliability of IS, this principle justifies the use of an architectural approach and system archetypes as tools for determining functional reliability.

3. The principle of diversity. The essence of this principle is to recognize the existence of a set of forms (morphisms) of errors made by subjects in the development of special-purpose IS, as well as defects caused by these errors. A partial manifestation of the diversity principle is multi-criteria, which implies the need to take into account and optimize different criteria. At the same time, the same criterion can be the basis for making decisions, including regarding the continuation or completion of tests of parts of IS for various functional purposes.

4. The principle of decomposition. As part of ensuring the functional reliability of the IS, this principle means the possibility of step-by-step selection of key factors, time, and spatial characteristics of the occurrence of failures, as well as resources and mechanisms aimed at their prevention or elimination.

5. The principle of integration (composition). This principle provides for the possibility of building a set of models that contribute to the choice of rational strategies for achieving the required level of functional reliability of IS.

6. The principle of equivalent ways of achieving the goal. Taking into account the uncertainty inherent in both the state of the special-purpose IS and its individual characteristics at different stages of the life cycle, it is impossible to implement the principle of unity of R. Colman when building models focused on solving the tasks of ensuring functional reliability.

It is assumed that there are many alternative ways to achieve an acceptable level of functional reliability, which are implemented both within the framework of reactive and proactive approaches. The choice of a specific method should be based on the assessment of effectiveness and efficiency in the specific conditions of project implementation.

7. The principle of system readiness. In the context of ensuring functional reliability, this principle consists of the fact that an incident (failure) leads to negative consequences only when a set of conditions necessary for its occurrence occurs simultaneously. This, in turn, can initiate the implementation of the so-called "domino" principle. Therefore, the main principle of failure prevention is to exclude the conditions under which this can happen and become a trigger of a chain reaction of failures.

8. The principle of objectivity. According to this principle, only those that rely on proven knowledge and reliable empirical data should be selected from a set of possible estimates. The a priori uncertainty of the information cannot be compensated by assumptions that have no actual basis, since such a replacement leads to distortion of the results and reduces the reliability of conclusions about the state of the special-purpose IS.

9. The principle of multidimensionality. This principle provides for the functioning of special-purpose IS in several dimensions, such as time, energy, frequency, information, and others. That is why the process of functioning of special-purpose IS should be considered from the standpoint of multidimensionality and reduced to their comprehensive assessment.

Below are approaches to ensuring the functional reliability of special-purpose IS, each of which is based on the principles of creating and operating large and complex systems:

1. The system approach, which acts as a methodological basis of the research, is focused on the analysis of the processes of ensuring the functional reliability of the IS from the standpoint of system-wide principles**.** Within the framework of this approach, it becomes a priority to identify the functional importance of individual subsystems in the formation of the properties of special-purpose IS as a whole, as well as to identify the features of the interaction of special-purpose IS and the external environment.

2. A dynamic approach that involves taking into account changes in the composition and content of IS requirements under the influence of the external environment, including the transformation of end-user needs, as well as changes in the volume and availability of IS resources. Within this approach, the IS is considered as a dynamic object, which state is subject to the influence of both internal and external factors.

3. A structural approach aimed at identifying regularities in the creation of a system for ensuring the functional reliability of IS, which allows establishing the relationship between the structure and its properties.

4. The cybernetic approach, implemented as a methodological framework for research and management, considers the reliability assurance system as a manageable dynamic object. Management efficiency in this context is ensured on the basis of the use of feedback mechanisms, allowing the adjustment of IS behavior on the basis of information on the internal state of the facility and the state of the external environment.

5. A situational approach focused on making project and management decisions under conditions of uncertainty and dynamic changes in the operational situation. The situational approach is based on the analysis of up-to-date information and the use of accumulated experience in the conditions of changes in IS resources.

6. A resource-targeted approach that focuses on the justified and purposeful distribution of resources between activities aimed at ensuring the functional reliability of the IS. It involves setting priorities and developing strategies that ensure the achievement of given targets with the limited resources of the special-purpose IS.

7. An informational approach focused on the creation of complex ISs that implement complex processing of various types of information. Within this methodology, the focus of research is on information flows related to the study of the needs of end users of special-purpose IS.

8. A value-oriented approach focuses on creating a product that meets the needs and expectations of users as much as possible. The main idea of the approach is to take maximum account of the initial stages of creating all the features of using IS to ensure that the functional reliability of IS meets the requirements set by users as much as possible.

9. The reactive approach focuses on the analysis of test results and retrospective data reflecting the experience of IS operation, to identify errors, establish the causes of their occurrence, and patterns in the appearance of failures.

10. A proactive approach involves identifying both present and future problems related to reliability assurance, the main objective of failure prevention. However, it is necessary to understand that it is impossible to prevent all failures before their occurrence, so it is necessary to maintain a balance between proactive and reactive approaches.

11. The barrier approach focuses on the development of multi-layered, echeloned protection systems against aggressive external influences. None of the individual barriers provides full protection against the negative effects caused by the potential hazard. However, integration into a single system of barriers allows to increase in the level of protection due to the manifestation of a systemic effect. The conceptual basis of the "barrier thinking approach" (barrier thinking) is the philosophy of "defense-in-depth" [20].

Let's consider *the main criteria for ensuring the functional reliability of special-purpose IS*:

1. Connectivity. Special-purpose ISs include sets of different types of agents, the interaction between which forms connections of different strengths, which determine the level of correspondence and mutual influence between agents.

2. Autonomy. Agents function without centralized management, having a certain autonomy within the established rules adopted for special-purpose IS. The high autonomy of the agents implies a higher complexity of the system.

3. Emergence. The behavior of special-purpose IS is formed from local interactions of self-organizing agents that lead to the emergence of global structures. These structures, in turn, begin to influence the behavior of the agents themselves in the form of negative (weakening) or positive (reinforcing) feedback.

4. Unbalance. The special-purpose IS functions in conditions of constant external and internal disturbances. Such changes can lead to fluctuations associated with a cyclic transition from one equilibrium state to another, with the strength of the bonds between the elements affecting the stability of the special-purpose IS. Understanding the strength of connections between special-purpose IS elements allows to identify "bottlenecks" and manage the behavior of special-purpose IS in the future.

5. Nonlinearity. The final behavior of a special-purpose IS is not a simple sum of the behavior of individual components (agents). Minor disturbances can cause large-scale consequences — manifestation "butterfly effect". The emergence of autocatalytic processes is often caused by minor and sometimes random events.

6. Self-organization. Special-purpose IS can independently rearrange structure and behavior in response to changes, restoring stability and preventing degradation. Such adaptability can generate collective decisions and new forms of functioning — manifestations of "emergent mind".

7. Evolution. If to consider the external environment of a special-purpose IS as a set of all systems interacting with it, it becomes clear that complex systems are open: they not only adapt to environmental conditions, but also actively transform them. Such interaction has, as a rule, an irreversible nature — decisions made in conditions of self-organization cannot be reproduced again, since the initial conditions are already lost. In other words, developing in parallel and asynchronously, complex systems and their environments affect each other; that is, they co-evolve.

The given criteria, principles, and approaches to ensuring the functional reliability of special-purpose IS are components of the methodology for ensuring the functional reliability of special-purpose IS at the stage of preparation for application.

METHODOLOGY FOR ENSURING THE FUNCTIONAL RELIABILITY OF SPECIAL-PURPOSE ISS AT THE STAGE OF PLANNING THEIR APPLICATION

The conceptual framework of this study relies on the following system of views:

1. Ensuring the functional reliability of the special-purpose IS as part of the intelligent decision support systems of network-centric management of distributed complex systems belongs to the class of management tasks in the conditions of dynamic fuzzy management goals.

2. ISs are a type of complex subject-centric system. This fact confirms the possibility of scientifically based adaptation of known approaches to the model description of failures that occur when managing complex systems of various nature in the field of ensuring the functional reliability of special-purpose IS. A model description of problematic situations is the basis for informational support for making rational decisions regarding their settlement.

3. A critical factor in functional reliability is the failure of IS elements. Sources of failures are destructive factors that affect special-purpose IS, as well as technical failures of software and hardware.

4. The basis for ensuring functional reliability is the complex use of information obtained from various sources: the subsystem of technical analysis of IS, the results of processing retrospective data on the experience of using special-purpose IS in similar conditions.

5. Functional reliability is determined by the number of failures that occurred at different stages of the life cycle, starting with the awareness of the presence of a problematic situation, for the settlement

of which the IS use is necessary. Moments of failure and moments of their occurrence are distributed in space and time.

As part of the methodology for ensuring the functional reliability of special-purpose ISs at the planning stage of their application, the following is proposed in this study:

– the method of constructing multidimensional dependencies based on the joint use of measurement data and fuzzy expert evaluations;

– the method of assessing the functional reliability of information services based on the concept of profiles;

– the method of multidimensional assessment of the functional reliability of special-purpose IS.

A METHOD OF CONSTRUCTING MULTIDIMENSIONAL DEPENDENCIES BASED ON THE JOINT USE OF MEASUREMENT DATA AND FUZZY EXPERT EVALUATIONS

The use of information on the causes and places of occurrence of failures in special-purpose IS is currently gaining particular importance due to the high cost associated with insufficient functional reliability of the IS [2, 10].

One of the aspects of the conceptual foundations of ensuring the functional reliability of special-purpose IS is highlighted as follows:

– the basis of ensuring functional reliability is the complex use of information obtained from various sources: structural analysis of the architectures of systems for ensuring functional reliability and the internal structure of the IS, the results of processing retrospective data (including metric characteristics) related to the manifestation of failures of various nature, expert evaluations of subjects involved in the creation and application of special-purpose IS, forms the basis of the systematic approach of ensuring functional reliability at the application stage.

The classical approach to solving this task is based on the use of empirical (regression) data characterizing the relationship between the parameters of the conditions and processes of the implementation of IS creation projects and the parameters characterizing the properties of the obtained IS. Traditional approaches to the construction of regression dependencies [7] assume the presence of a table of values of independent and dependent random variables that are jointly observed.

Regression dependencies are one of the main tools for constructing empirical descriptive models of inertialess objects.

To date, the theoretical apparatus for building regression models has been developed, on the basis of which software-implemented tools have been developed. At the same time, a special place in the tasks of regression analysis is occupied by linear models. This is because they create the basis of studying the main properties of objects under conditions of a small number of measurement data, which is characteristic when studying the experience of using special-purpose IS due to the uniqueness of this process [4, 12, 13].

The information basis for the construction of regression dependencies is a table of jointly observed values of independent and dependent random variables. When solving practical tasks, one has to face a situation where the formation of such tables meets a number of difficulties. Under such conditions, the lack of measurement data has to be compensated by expert evaluations of various kinds: as expected values; in the form of an interval of possible values; in the form of a collection of expected values, an interval of possible values of a random variable, as well as personal (subjective) opinions of experts.

Due to these circumstances, it is of interest to develop a method of converting measurement data and fuzzy expert estimates, given in various forms to the form of a table of jointly observed values, which is the basis for the construction of multidimensional regression dependencies.

An approach to the construction of multidimensional regression dependencies is proposed in the case when the output data corresponding to different components of vectors of independent and dependent variables are presented either in the form of measurement results or in the form of various expert evaluations.

The classical approach to the construction of regression dependencies can be matched with the scheme

$$A^{(0)}:\left\{\vec{x},\vec{y}\right\}_1^N \to \vec{y}=\varphi_0\left(\vec{x},\vec{\theta}\right), \tag{6.1}$$

where $\left\{\vec{x},\vec{y}\right\}_1^N$ – a set of jointly observed values of the components of independent vectors $\vec{x}$ and dependent $\vec{y}$ magnitudes;

$\varphi_0\left(\vec{x},\vec{\theta}\right)$ – functional dependence given in parametric form;

N – number of pairs of values $\vec{x}$ and $\vec{y}$.

When solving practical tasks, due to the complexity of organizing the collection of initial data, formation $\left\{\vec{x},\vec{y}\right\}_1^N$ serious difficulties arise.

Work [19] describes an approach to the construction of one-dimensional non-parametric strict functional dependencies based on solving the inverse problem of constructing the distribution law of a random argument function.

The scheme for solving the reverse problem has the form

$$A^{(1)}:\left\{F\left(x\right),F\left(y\right)\right\} \to y=\varphi_1\left(x\right), \tag{6.2}$$

where $F(x)$, $F(y)$ – estimates of one-dimensional laws of distribution of independent and dependent random variables, determined based on sample data processing

$$A^{(2)}:\left\{x\right\}_1^N \to x=F\left(x\right), A^{(2)}:\left\{y\right\}_1^M \to y=F\left(y\right). \tag{6.3}$$

A feature of scheme (6.2) is that the properties of the sample data $\left\{x\right\}_1^N$, $\left\{y\right\}_1^M$ (scope, accuracy of registration) can be different. A limitation of scheme (6.2) is the need to justify the very fact of the presence of strict dependencies, for example, based on the physical content of the task.

Expert estimates are presented as estimates of the expected value of a random variable $M[z]$ (in other words, estimates of mathematical expectation) and/or the interval of possible values of a random variable Z (The following options for presenting intervals are possible: $z\in\left[a_z,b_z\right]; z\in\left[a_z,\infty\right)$).

The works [2, 4, 7] consider the following separate tasks of constructing estimates of the laws of distribution of random variables based on expert estimates:

If only the limits of the interval of possible values of a random variable are known, i.e $z \in [a_z, b_z]$, that's it

$$A^{(3)} : [a_z, b_z] \to F^{(3)}(z). \tag{6.4}$$

In this case, the optimal assessment $F^{(3)}(z)$ there is a uniform distribution law, if mathematical expectations are known $M[z]$, and the interval of possible values of a random variable is presented in the form $[a_z, \infty)$, then the construction of the assessment is reduced to solving the task

$$A^{(4)} : \{M[z], [a_z, \infty)\} \to F^{(4)}(z). \tag{6.5}$$

Optimal assessment $F^{(4)}(z)$ in this case, there is an indicative distribution law

$$F^{(4)}(z) = 1 - e^{-\lambda z}. \tag{6.6}$$

The parameter of the distribution law is determined by the ratio

$$\lambda = \left(M[z] - a_z\right)^{-1}.$$

If known $M[z]$ and the limits of the interval $[a_z, b_z]$, then the assessment of the distribution law is a result of solving the task

$$A^{(5)} : \{M[z], [a_z, b_z]\} \to F^{(5)}(z). \tag{6.7}$$

The assessment shall be presented as

$$F^{(5)}(z) = \int_{a_z}^{z} e^{\mu_0 + \mu_1 \tau} d\tau, \tag{6.8}$$

and the parameters of the model μ0, μ1 are a result of solving a system of equations

$$\begin{cases} \int_{a_z}^{b_z} e^{\mu_0 + \mu_1 \tau} d\tau = 1 \\ \int_{a_z}^{b_z} \tau e^{\mu_0 + \mu_1 \tau} d\tau = M[z] \end{cases}. \tag{6.9}$$

In a separate case, if it is additionally known that $F^{(5)}(a_z) = 0$, $F^{(5)}(b_z) = 1$, $F^{(5)}(z)$ is a triangular distribution.

The transformation of expert estimates into the form of laws of distribution of continuous random variables allows expanding scheme (6.2) in the case of the construction of one-dimensional non-parametric regression dependencies based on the joint use of measurement data and expert estimates.

At the same time, the methods of constructing strict functional dependencies based on solving the inverse problem of constructing the law of distribution of the function of a random argument, including those based on the joint use of measurement data and expert evaluations, are focused on the study of one-dimensional dependencies and are not adapted to the study of the behavior of multidimensional systems.

Next, an approach is described that allows the adaptation of the construction methods of strict one-dimensional functional dependencies to investigate the behavior of multivariate objects.

Static models used to describe multivariate inertialess multiconnected objects include multivariate regression dependencies

$$y_j = \varphi_j\left(x_{j1},\ldots,x_{jk_j}\right),\ j=\overline{1;N}, \tag{6.10}$$

where y_j – vector component value Y dependent variable;

x_{jk_j}, $j=\overline{1;N}$ – components of the vector of independent variables;

$\varphi_j(\bullet)$ – relationship-establishing dependency (parametric or non-parametric) j-th components of the vector of the dependent variable on the values of the components of the vector of the independent variable.

Construction of the ratio of the type (6.10) depending on the form of representation of the components of the vector of the independent variable x_{jk_j} and the dependent variable y_j (measuring data, or expert evaluations) boils down to using one of the transformations (6.4), (6.6), (6.8) to construct one-dimensional laws of distribution of random variables, followed by an assessment of strict non-parametric functional dependencies taking into account (6.2).

The presence of one-dimensional functional dependencies $y_j = f_{ij}\left(x_j\right)$ allows to reduce the formation of a table of jointly observed values of independent and dependent variables to the sequential implementation of the following steps.

Action 1. Independent ones are generated q-th $\left(q=\overline{1;Q}\right)$ the value of random variables $\xi_j^{(q)}$, which correspond to the estimates of distribution laws $\hat{F}\left(x_j\right)$ based on random number sensors by inverse transformation method [11]

$$\xi_j^{(q)} = \hat{F}^{-1}\left(x_j\right),\ i=\overline{1;M} \tag{6.11}$$

Action 2. By values $\xi_j^{(q)}$, using constructed one-dimensional parametric dependencies $f_{ij}\left(x_j\right)$ calculated values y_j according to the rules

$$y_i^q = \sum\nolimits_i f_{ij}\left(\xi_j^{(q)}\right). \tag{6.12}$$

Action 3. Meaning $\left\{\xi_j^{(q)}, y_i^{(q)}\right\}$ entered as a row in the table of values.

In the future, the formed table is processed by known methods of multidimensional regression analysis.

Thus, a method of constructing multivariate regression dependencies is proposed, which allows for representing the IS in the form of a multi-connected object by a set of one-dimensional non-parametric regression dependencies. The basis of the method is the transformation of the original data, which are initially presented in different forms, to the form of the law of distribution of a continuous random variable. The method allows for investigating the behavior of multi-connected objects in the case where some components are metric characteristics, and some are represented by expert evaluations.

A distinctive feature of the proposed approach is that a multi-connected object is described by a collection of one-dimensional strict nonparametric regression dependencies. A limitation of the approach is the need for a priori logical justification of the existence of a relationship between independent and dependent variables.

A METHOD OF ASSESSING THE FUNCTIONAL RELIABILITY OF INFORMATION SERVICES BASED ON THE CONCEPT OF PROFILES

Functional reliability is determined by the ability of the IS to provide the user with complete information in a timely manner [11, 20].

When implementing network-centric management and creating a single information and management space, it is advisable to consider information services rather than functions.

One of the factors that should be taken into account when building a profile of information services is the consideration of the set of IS operation modes (full-time, the cause of non-full-time; non-full-time). In addition, different information needs of service consumers arise in different modes, which causes them to change their assessments of the properties of the same services in different modes.

The profile of an information service means a complete set of alternatives (for example, a set of alternative categories of users, functions, etc.), for each of which occurrence is probable.

Formally, the profile is a tree, each edge of which is matched by a weight characteristic that characterizes the probability of crossing it.

Since it is assumed that the transition of one and only one of the edges from each vertex takes place necessarily, and which is not known in advance, each of the nodes is assigned the correspondence of formula (6.13)

$$\sum_{j} P_{i,j}^{(l)} = 1, \tag{6.13}$$

where l – layer number;

i – node number in the layer;

j – the number of edges coming out of i-th nodes.

Building a service profile allows to quantify the degree of coverage of tasks by special-purpose IS information services. Yes, for i-th the task of the task layer, the degree of coverage is determined by the following formula

$$SS_i^{(S)} = \frac{{}^2S_i^{(R)}}{{}^2S_i^{(R)} + {}^2S_i^{(PR)}}, \quad (6.14)$$

where ${}^2S_i^{(R)}$ – the number of implemented information services;
${}^2S_i^{(PR)}$ – the number of information services expected to be implemented.

In a similar way, it is possible to evaluate all modes of operation. One of the most important characteristics that must be taken into account when allocating resources, in particular temporary ones, to the development of new information services and the improvement of existing ones, is the degree of coverage of tasks by information services.

Different variants of service profiles can be matched with a system model of the form

$$<E, Pr, Rel>, \quad (6.15)$$

where E – the set of possible events (simple paths in the graph);

Pr – the set of probability values (implicit profiles) corresponding to each of the events (implicit profile is defined as the product of the weights of the arcs entering the simple path);

Rel – the set of reliability assessments of information services.

When constructing formal models, there is a need to convert subjective user estimates to quantitative estimates.

To evaluate reliability characteristics based on subjective user evaluations, it is proposed to use an approach based on the membership function of the linguistic scale.

The linguistic scale in terms of which users belonging to the same *m*-th target group evaluate the reliability of the information service has the form: {*low; medium; high*}.

To everyone *l*-th the value of the linguistic scale is matched by the membership function μ*l*, defined on the interval $\mu_l \in [0;1]$. Provisions of maximum membership *l*-th the value of a linguistic scale is defined as its reference value *rl* on axis *y*.

Each user expresses its opinion about the reliability of the information service and its degree of confidence in it.

It is postulated that the evaluations provided by users reflect their true independent opinion about the information service in the special-purpose IS.

The main issue of the task of quantifying expert evaluations is the selection of membership functions. Proposed solution: membership functions of a triangular type with the same boundaries that coincide with the boundaries of the axis $y \in [0;1]$.

The cumulative reliability score based on subjective user scores is determined based on the ratio

$$D^{(m)} = \frac{\sum_{k=1}^{N_m} \mu_k^{(l)} * r_l^{(k)}}{\sum_{k=1}^{N_m} \mu_k^{(l)}}, \quad (6.16)$$

where N_m – the number of users belonging to *m*-th target group;

$r_l^{(k)}$ – the reference value corresponding to l-th the value of the linguistic scale determined k-th user;

$\mu_k^{(l)}$ – the degree of confidence k-th user in the assessment given.

The main uncertainties complicating practical use (6.17) are:

– approaches to choosing values r_l;

– approaches to choosing the form of the membership function μ_l;

– approaches to defining boundaries $\left\{r_l^{(up)}, r_l^{(down)}\right\}$.

To eliminate the specified uncertainties, it is proposed to use the following approach. The basis of the definition r_l to take into account the share of users owned m-th target group and selected l-th the value of the linguistic scale. Under such an approach as r_1 the value is $\frac{n_1^{(m)}}{N_m}$, where $n_1^{(m)}$ – number of users m-th target groups that have selected the value of the linguistic scale "low".

As r_2 the value is: $r_2 = \frac{n_1^{(m)}}{N_m} + \frac{n_2^{(m)}}{2N_m}$, where $n_2^{(m)}$ – the number of users of the target group who selected the value of the linguistic scale "average".

Meaning r_3 determined by the ratio: $r_3 = \frac{n_1^{(m)}}{N_m} + \frac{n_2^{(m)}}{N_m}$. Magnitude r_2 is the middle of the sub-interval between values r_1 and r_3.

Regarding the uncertainties related to the choice of the form of the membership function, as well as the lower and upper bounds $\left\{r_l^{(up)}, r_l^{(down)}\right\}$, the following should be noted.

If the value r_1 corresponds to the maximum value of the linguistic scale "low", then to the left r_1 the degree of satisfaction cannot increase, i.e μ_1 at $y \in \left[0; r_1\right]$ takes the value 1. Right-hand than a point r_1 the expert's confidence in the selected assessment drops. Given that any information about the rate of value reduction μ_1 as possible to move away from the right of the point r_1 absent, following the principle of entropy maximization [5], it is appropriate to consider this rate constant.

In the absence of information imposing restrictions on the scope of the definition μ_1, as such an area, it is advisable to take the entire interval $y \in [r_1; 1]$.

Consequently, the developed approach allows obtaining metric estimates of the reliability of information services taking into account the combined use of both measurement data and subjective user estimates.

It should be noted that the proposed approach to the quantification of expert evaluations is largely formalized, which allows it to be implemented as a tool in the software of modern special-purpose IS.

THE METHOD OF MULTIDIMENSIONAL ASSESSMENT OF THE FUNCTIONAL RELIABILITY OF SPECIAL-PURPOSE IS

Let them be known: a set of solutions $D = \left\{d_j\right\}, \left(j = \overline{1,m}\right)$, which corresponds to the result of the multidimensional assessment of the functional reliability of the special-purpose IS y; a set of input indicators $X = \left\{x_i\right\}, \left(i = \overline{1,n}\right)$; ranges of quantitative change of each input indicator characterizing the functional reliability of the special-purpose IS $x_i \in \left[\underline{x}_i, \overline{x}_i\right], i=\overline{1,n}$; membership functions that allow to present indicators $x_i, i=\overline{1,n}$ in the form of fuzzy sets knowledge matrix, which can be graphically displayed in the form of **Fig. 6.1**.

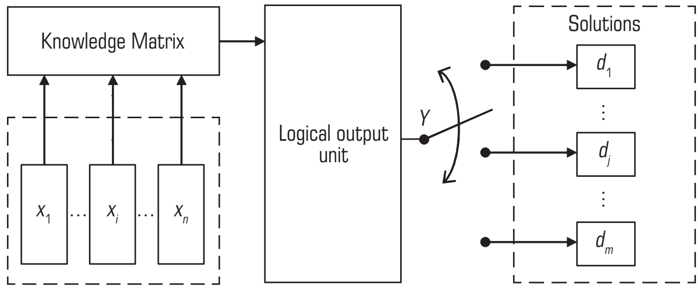

Fig. 6.1 Graphical interpretation of the process of holding a multidimensional assessment of the IS functional assessment

From the analysis of the functioning of the special-purpose IS and indicators for assessing the functional reliability of the special-purpose IS in different operating conditions, the directions of assessment are determined: the similarity of the indicators of the functional reliability of the special-purpose IS, and their changes during the performance of the special-purpose IS tasks.

Let's describe the process of multidimensional assessment of the functional reliability of special-purpose IS

$$D(k)=f\begin{bmatrix} Y_1(k-1), Y_2(k-1),\ldots,Y_n(k-1), Q(k-1), R(k-1), \\ Z_1(k-1),\ldots,Z_n(k-1) \end{bmatrix}, \tag{6.17}$$

where $Y_1(k-1)$ – a vector characterizing the functional reliability of a special-purpose IS based on the first indicator of the assessment on k-1 simulation steps;

$Y_2(k-1)$ – a vector characterizing the functional reliability of the special-purpose IS according to the second indicator on k-1 simulation steps;

$Y_n(k-1)$ – a vector characterizing the functional reliability of special-purpose IS by n-th assessment indicator on k-1 simulation steps;

$Q(k-1)$ – a vector characterizing the time-frequency indicators of the functioning of the subsystem for assessing the functional reliability of special-purpose IS;

$R(k-1)$ – a vector characterizing the information indicators of the functioning of the subsystem for assessing the functional reliability of special-purpose IS;

$Z_1(k-1),\ldots,Z_n(k-1)$ – vectors characterizing modes of operation of special-purpose IS.

For indicators that have a quantitative dimension, the range of change is divided into four quanta. This will provide the ability to transform a continuous universal set $U=\left[\underline{u},\overline{u}\right]$ into a discrete five-element set [2]

$$U=\left\{u_1,u_2,\ldots,u_5\right\}, \tag{6.18}$$

where $u_1=\underline{u}$, $u_2=\underline{u}+\Delta_1$, $u_3=u_2+\Delta_2$, $u_4=u_3+\Delta_3$, $u_5=\overline{u}$, and $\Delta_1+\Delta_2+\Delta_3+\Delta_4=\overline{u}-\underline{u}$, $\overline{u}\left(\underline{u}\right)$ – the upper (lower) limit of the indicator change range. Then all matrices of even comparisons have dimension. The choice of four quanta is determined by the possibility of approximating nonlinear curves at five points [3, 19].

Each of the introduced terms is a fuzzy set given by the corresponding membership function. In the general case, input variables x_1, x_2, …, x_n may be given by a number, a linguistic term, or by the thermometer principle [1, 16].

A multidimensional assessment of the functional reliability of special-purpose IS using expert information is carried out using fuzzy logical equations [4], which are a knowledge matrix and a system of logical statements. These equations make it possible to calculate the values of the membership functions of various identification results at fixed values of the input indicators.

Let $\mu^{\alpha_i^{jp}}(x_i)$ – indicator membership function $x_i\in\left[\underline{x},\overline{x}\right]$ fuzzy term α_i^{jp}, $i=\overline{1,n}$, $j=\overline{1,m}$, $p=\overline{1,l_i}$; $\mu^{d_j}\left(x_1,x_2,\ \ldots,x_n\right)$ – the function of belonging to the vector of input variables $X=\left(x_1,x_2,\ \ldots,x_n\right)$ and the value of the initial assessment $y=d_j$, $j=\overline{1,m}$.

The relationship between these functions is determined by a fuzzy knowledge base and can be represented as the following logical equations

$$\begin{aligned}&\mu^{d_j}\left(x_1,x_2,\ \ldots,x_n\right)=\mu^{a_1^{j1}}\left(x_1\right)\wedge\mu^{a_2^{j1}}\left(x_2\right)\wedge\ldots\wedge\mu^{a_n^{j1}}\left(x_n\right)\vee\\&\vee\mu^{a_1^{j2}}\left(x_1\right)\wedge\mu^{a_2^{j2}}\left(x_2\right)\wedge\ldots\wedge\mu^{a_n^{j2}}\left(x_n\right)\ldots\\&\ldots\mu^{a_1^{jl_j}}\left(x_1\right)\wedge\mu^{a_2^{jl_j}}\left(x_2\right)\wedge\ldots\wedge\mu^{a_n^{jl_j}}\left(x_n\right),\ j=\overline{1,m}.\end{aligned} \tag{6.19}$$

The equations are obtained from a fuzzy knowledge base by replacing variables (linguistic terms) with their membership functions, and operations I and OR – with operations $\wedge$ and $\vee$.

Let's briefly write the system (6.20) as follows

$$\mu^{d_j}\left(x_i\right)=\bigvee_{p=1}^{l_j}\left[\bigwedge_{i=1}^{n}\mu^{\alpha_i^{jp}}\left(x_i\right)\right],\ j=\overline{1,m}. \tag{6.20}$$

Fuzzy logical equations are an analogue of Zadeh's introduced procedure of fuzzy logical inference [4, 5], which is carried out using the operation "fuzzy (min-max) composition", in which operations $\wedge$ and $\vee$ min and max operations correspond.

6.2 EVALUATION OF THE EFFECTIVENESS OF THE FUNCTIONAL RELIABILITY OF THE SPECIAL-PURPOSE IS ACCORDING TO THE DEFINED CRITERION

To determine the effectiveness of the scientific and methodological apparatus for ensuring the functional reliability of special-purpose information systems, modeling of the specified provisions was carried out when solving the task of assessing the state of functional reliability of the special-purpose IS of the group of troops (forces) under the initial conditions.

Separate parts of the computational experiment, using the proposed scientific and methodological apparatus, are given in **Tables 6.1** and **6.2**. The general computational experiment is laid out on 126 sheets; in this section, only its separate part is presented.

Table 6.1 Results of calculations of the functional reliability of special-purpose IS using membership functions

№ rules	Results of calculations of decision membership functions according to the rules											
	1	2	3	4	5	6	7	8	9	10	11	12
1	2	3	4	5	6	7	8	9	10	11	12	13
1	0.00065	0.043	0.042	0.037	0.064	0.03	0.0068	0.0061	0.0087	0.045	0.066	0.045
2	0.06	0.0387	0.1148	0	0.1265	0.1534	0.1487	0.0112	0.0765	0.1381	0.166	0.1612
3	0.063	0.0443	0.05	0.0259	0.0122	0.0576	0.0341	0.067	0.0466	0.052	0.064	0.044
4	0.088	0.074	0.153	0.068	0.004	0.1	0.0018	0.169	0.0052	0.053	0.046	0.163
5	0.174	0.0147	0.083	0.083	0.076	0.002	0.102	0.083	0.162	0.116	0.09	0.105
6	0.028	0.057	0.019	0.036	0.047	0.038	0.025	0.028	0.0029	0.005	0.036	0.063
7	0.061	0.067	0.056	0.045	0.012	0.014	0.0007	0.012	0.022	0.056	0.069	0.00216
8	0.197	0.219	0.211	0.232	0.197	0.203	0.057	0.07	0.119	0.13	0.138	0.0054
9	0	0.122	0.124	0.157	0.243	0.003	0.262	0.208	0	0.165	0.084	0.151
10	0.146	0.079	0.142	0.076	0.005	0.121	0.107	0.121	0.114	0.091	0.049	0.139
11	0.165	0.139	0.065	0.044	0.07	0.1	0.083	0.163	0.061	0.165	0.133	0.086
12	0.026	0.039	0.001	0.006	0.043	0.021	0.036	0.013	0.014	0.034	0.02	0.03
13	0.035	0.006	0.037	0.04	0.021	0.038	0.004	0.0005	0.033	0.017	0.021	0.017
14	0.0054	0.003	0.033	0.021	0.007	0.028	0.029	0.0076	0.05	0.033	0.017	0.038
15	0.049	0.009	0.012	0.021	0.033	0.03	0.044	0.023	0.024	0.034	0.018	0.041
16	0.03	0.042	0.027	0.019	0.014	0.047	0.029	0.011	0.036	0.023	0.05	0.033
17	0.021	0.0005	0.031	0.028	0.032	0.047	0.031	0.02	0.024	0.012	0.02	0.032
18	0.03	0.008	0.016	0.044	0.02	0.036	0.016	0.048	0.05	0.014	0.035	0.0086

Continuation of Table 6.1

1	2	3	4	5	6	7	8	9	10	11	12	13
19	0.026	0.039	0.038	0.014	0.003	0.002	0.031	0.011	0.031	0.0076	0.034	0.013
20	0.007	0.046	0.049	0.033	0.015	0.007	0.049	0.023	0.05	0.016	0.03	0.034
21	0.042	0.026	0.026	0.025	0.037	0.029	0.027	0.021	0.015	0.01	0.041	0.00758
22	0.126	0.027	0.017	0.315	0.033	0.096	0.206	0.305	0.093	0.146	0.116	0.00332
23	0.391	0.462	0.616	0.443	0.077	0.231	0.0064	0.077	0.616	0.109	0.237	0.61
24	0.132	0.005	0.04	0.002	0.035	0.139	0.063	0.0088	0.112	0.118	0.109	0.037
25	0.14	0.125	0.044	0.139	0.13	0.074	0.107	0.125	0.1	0.054	0.021	0.158
26	0.041	0.047	0.02	0.026	0.008	0.016	0.025	0.019	0.043	0.031	0.04	0.049
27	0.022	0.014	0.041	0.037	0.034	0.046	0.013	0.027	0.022	0.011	0.042	0.012
28	0.038	0.008	0.015	0.011	0.018	0	0.017	0.033	0.018	0.042	0.043	0.023
29	0.037	0	0.039	0.015	0.035	0.004	0.021	0.017	0.039	0.031	0.004	0.05
30	0.007	0.028	0.011	0.031	0.012	0.048	0.021	0.026	0.032	0.036	0.033	0.026
31	0.032	0.011	0.007	0.018	0.033	0.036	0.04	0.011	0.038	0.024	0.018	0.045
32	0.041	0.02	0.05	0.027	0.008	0.017	0.05	0.024	0.031	0.045	0.034	0.022
33	0.022	0.019	0.039	0.049	0.043	0.000	0.045	0.029	0.0025	0.016	0.013	0.037
34	0.042	0.048	0.011	0.02	0.013	0.042	0.006	0.0035	0.014	0.0056	0.049	0.049
35	0.05	0.032	0.032	0.037	0.027	0.014	0.005	0.046	0.038	0.02	0.037	0.039
36	0.081	0.044	0.049	0.102	0.016	0.146	0.053	0.114	0.133	0.054	0.054	0.086
37	0.139	0.153	0.025	0.172	0.014	0.142	0.025	0.114	0.063	0.04	0.091	0.135
38	0.019	0.044	0.012	0.004	0.03	0.047	0.008	0.024	0.05	0.033	0.008	0.0015
39	0.023	0.034	0.041	0.003	0.015	0.015	0.05	0.048	0.018	0.036	0.035	0.027
40	0.034	0.063	0.056	0.023	0.085	0.045	0.025	0.0073	0.012	0.113	0.078	0.036
41	0.045	0.016	0.023	0.027	0.032	0.006	0.027	0.011	0.036	0.045	0.038	0.041
42	0.018	0.013	0.019	0.038	0.05	0.021	0.023	0.03	0.028	0.024	0.015	0.045
43	0.0005	0.031	0.033	0.028	0.047	0.023	0.0005	0.035	0.0066	0.034	0.044	0.031
Defense operation	0.166	0.151	0.177	0.011	0.146	0.162	0.152	0.173	0.164	0.147	0.161	0.168

Table 6.2 Comparative results of the process of assessing the functional reliability of the special-purpose IS of the state of the grouping of troops (forces)

Evaluation limits	With use scientific and methodological apparatus	Without use scientific and methodological apparatus
Promptness of the process of assessing the functional reliability of the special-purpose IS of the group of troops (forces)		
Better case	59–502 s	82–801.3 s
Worse case	255.1–2501.5 s	402.8–4007 s
The reliability of the obtained solutions for assessing the functional reliability of the special-purpose IS of the group of troops (forces)		
Better case	0.94–1.0	0.76–0.82
Worse case	0.9–1.0	0.66–0.77

From the analysis of **Table 6.2**, it can be concluded that the proposed scientific and methodological apparatus provides an increase in efficiency by an average of 40%, while ensuring high reliability of the obtained results at the level of 92%.

6.3 DISCUSSION OF THE RESULTS OF THE DEVELOPMENT OF A SCIENTIFIC AND METHODOLOGICAL APPARATUS FOR ENSURING THE FUNCTIONAL RELIABILITY OF SPECIAL-PURPOSE IS

The advantages of the proposed scientific and methodological apparatus for ensuring functional reliability are as follows:

– to systematically conduct a multi-level assessment of the state of functional reliability of special-purpose IS using the proposed principles of its provision. This will make it possible to comprehensively assess the functional reliability of the special-purpose IS, both as a separate element of it and as a whole, in comparison with works [2, 5];

– to consider from different sides the problem of ensuring the functional reliability of special-purpose IS, with the help of the proposed methodology, which expands the system of views on its provision, compared to works [4, 7];

– to build multidimensional dependencies of the state of functional reliability of special-purpose IS (expressions (6.1)–(6.12)), which will allow to evaluate the functional reliability of IS based on an arbitrary number of indicators, compared to works [9, 13];

– to assess the functional reliability of special-purpose ISs by sharing measurement data and fuzzy expert evaluations (expressions (6.1)–(6.12)), which will solve the dimension problem, compared to works [8, 12];

– to build a time dependence of the change in indicators that characterize the state of functional reliability of special-purpose IS (expressions (6.1)–(6.12)), which allows for determining the moments of deviation of their values from the nominal, in comparison with works [11, 14];

– to evaluate the functional reliability of information services based on the concept of profiles (expressions (6.13)–(6.16)), which achieves the possibility of decentralized influence on special-purpose IS to increase its functional reliability, compared to works [12, 16];

– reduce uncertainty about the state of functional reliability of special-purpose IS, by using an appropriate approach in the method of assessing the functional reliability of information services based on the concept of profiles, compared to works [10, 14];

– to conduct a comprehensive and multidimensional assessment of the state of functional reliability of the special-purpose IS due to the use of the method of multidimensional assessment of the functional reliability of the special-purpose IS (expressions (6.17)–(6.20)), compared to works [9, 17].

The disadvantage of the proposed scientific and methodological apparatus should include the need to coordinate the opinions of experts about the state of functional reliability of special-purpose IS.

The proposed scientific and methodological apparatus allows:

– conduct modeling of the process of assessing the functional reliability of special-purpose IS;

– determine effective measures to increase the efficiency of assessing the state of functional reliability of special-purpose IS;

– comprehensively assess the functional reliability of special-purpose IS.

The limitations of the study are the need to take into account the delay time for collecting and proving information from special-purpose IS sensors (sensors).

The proposed scientific and methodological apparatus should be used as software for automated troop control systems of the "Dzvin-AS", "Oreanda-PS" type, as well as integrated information systems of the "Delta" type.

CONCLUSIONS

The study proposes a scientific and methodological apparatus for ensuring the functional reliability of special-purpose information systems. The novelty of the proposed scientific and methodological apparatus consists in:

– systematic assessment of the state of functional reliability of special-purpose IS using the proposed principles of its provision. This will make it possible to comprehensively assess the functional reliability of the special-purpose IS, both as a separate element of it and as a whole;

– an extended consideration on various sides of the problem of ensuring the functional reliability of the special-purpose IS, with the help of the proposed methodology, which expands the system of views on its provision;

– construction of multidimensional dependencies of the state of functional reliability of the special-purpose IS, which achieves an assessment of the functional reliability of the IS based on an arbitrary number of indicators;

– the assessment of the functional reliability of special-purpose IS using the joint use of measurement data and fuzzy expert assessments, which solves the problem of dimensionality;

– the construction of the time dependence of changes in indicators that characterize the state of functional reliability of special-purpose IS, which allows determining the moments of deviation of their values from the nominal ones;

– the assessment of the functional reliability of information services based on the concept of profiles, which achieves the possibility of decentralized influence on the special-purpose IS to increase its functional reliability;

– reducing uncertainty about the state of functional reliability of special-purpose IS, due to the use of an appropriate approach in the method of assessing the functional reliability of information services based on the concept of profiles;

– conducting a comprehensive and multidimensional assessment of the state of functional reliability of a special-purpose IS, due to the use of the method of multidimensional assessment of the functional reliability of a special-purpose IS.

The proposed scientific and methodological apparatus provides an increase in the efficiency of assessing the IS functional reliability by an average of 40%, while ensuring high reliability of the obtained results at the level of 92%, which is confirmed by the results of a numerical experiment.

CONFLICT OF INTEREST

The authors declare that they have no conflict of interest in relation to this research, whether financial, personal, authorship or otherwise, that could affect the research and its results presented in this paper.

USE OF ARTIFICIAL INTELLIGENCE

The authors confirm that they did not use artificial intelligence technologies in creating the submitted work.

REFERENCES

1. Dudnyk, V., Sinenko, Y., Matsyk, M., Demchenko, Y., Zhyvotovskyi, R., Repilo, I. et al. (2020). Development of a method for training artificial neural networks for intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 3 (2 (105)), 37–47. https://doi.org/10.15587/1729-4061.2020.203301
2. Sova, O., Shyshatskyi, A., Salnikova, O., Zhuk, O., Trotsko, O., Hrokholskyi, Y. (2021). Development of a method for assessment and forecasting of the radio electronic environment. EUREKA: Physics and Engineering, 4, 30–40. https://doi.org/10.21303/2461-4262.2021.001940
3. Pievtsov, H., Turinskyi, O., Zhyvotovskyi, R., Sova, O., Zvieriev, O., Lanetskii, B., Shyshatskyi, A. (2020). Development of an advanced method of finding solutions for neuro-fuzzy expert systems

of analysis of the radioelectronic situation. EUREKA: Physics and Engineering, 4, 78–89. https://doi.org/10.21303/2461-4262.2020.001353

4. Zuiev, P., Zhyvotovskyi, R., Zvieriev, O., Hatsenko, S., Kuprii, V., Nakonechnyi, O. et al. (2020). Development of complex methodology of processing heterogeneous data in intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 4 (9 (106)), 14–23. https://doi.org/10.15587/1729-4061.2020.208554
5. Kuchuk, N., Mohammed, A. S., Shyshatskyi, A., Nalapko, O. (2019). The Method of Improving the Efficiency of Routes Selection in Networks of Connection with the Possibility of Self-Organization. International Journal of Advanced Trends in Computer Science and Engineering, 8 (1.2), 1–6.
6. Shyshatskyi, A., Zvieriev, O., Salnikova, O., Demchenko, Ye., Trotsko, O. and Neroznak, Ye. et al. (2020). Complex Methods of Processing Different Data in Intellectual Systems for Decision Support System. International Journal of Advanced Trends in Computer Science and Engineering, 9 (4), 5583–5590. https://doi.org/10.30534/ijatcse/2020/206942020
7. Pozna, C., Precup, R.-E., Horvath, E., Petriu, E. M. (2022). Hybrid Particle Filter–Particle Swarm Optimization Algorithm and Application to Fuzzy Controlled Servo Systems. IEEE Transactions on Fuzzy Systems, 30 (10), 4286–4297. https://doi.org/10.1109/tfuzz.2022.3146986
8. Yang, X.-S., Deb, S. (2013). Cuckoo search: recent advances and applications. Neural Computing and Applications, 24 (1), 169–174. https://doi.org/10.1007/s00521-013-1367-1
9. Mirjalili, S. (2015). The Ant Lion Optimizer. Advances in Engineering Software, 83, 80–98. https://doi.org/10.1016/j.advengsoft.2015.01.010
10. Yu, J. J. Q., Li, V. O. K. (2015). A social spider algorithm for global optimization. Applied Soft Computing, 30, 614–627. https://doi.org/10.1016/j.asoc.2015.02.014
11. Mirjalili, S., Mirjalili, S. M., Lewis, A. (2014). Grey Wolf Optimizer. Advances in Engineering Software, 69, 46–61. https://doi.org/10.1016/j.advengsoft.2013.12.007
12. Koval, V., Nechyporuk, O., Shyshatskyi, A., Nalapko, O., Shknai, O., Zhyvylo, Y. et al. (2023). Improvement of the optimization method based on the cat pack algorithm. Eastern-European Journal of Enterprise Technologies, 1 (9 (121)), 41–48. https://doi.org/10.15587/1729-4061.2023.273786
13. Gupta, E., Saxena, A. (2015). Robust generation control strategy based on Grey Wolf Optimizer. Journal of Electrical Systems, 11, 174–188.
14. Chaman-Motlagh, A. (2015). Superdefect Photonic Crystal Filter Optimization Using Grey Wolf Optimizer. IEEE Photonics Technology Letters, 27 (22), 2355–2358. https://doi.org/10.1109/lpt.2015.2464332
15. Nuaekaew, K., Artrit, P., Pholdee, N., Bureerat, S. (2017). Optimal reactive power dispatch problem using a two-archive multi-objective grey wolf optimizer. Expert Systems with Applications, 87, 79–89. https://doi.org/10.1016/j.eswa.2017.06.009
16. Koval, M., Sova, O., Orlov, O., Shyshatskyi, A., Artabaiev, Y., Shknai, O. et al. (2022). Improvement of complex resource management of special-purpose communication systems. Eastern-European Journal of Enterprise Technologies, 5 (9 (119)), 34–44. https://doi.org/10.15587/1729-4061.2022.266009

17. Ali, M., El-Hameed, M. A., Farahat, M. A. (2017). Effective parameters' identification for polymer electrolyte membrane fuel cell models using grey wolf optimizer. Renewable Energy, 111, 455–462. https://doi.org/10.1016/j.renene.2017.04.036
18. Zhang, S., Zhou, Y. (2017). Template matching using grey wolf optimizer with lateral inhibition. Optik, 130, 1229–1243. https://doi.org/10.1016/j.ijleo.2016.11.173
19. Khouni, S. E., Menacer, T. (2023). Nizar optimization algorithm: a novel metaheuristic algorithm for global optimization and engineering applications. The Journal of Supercomputing, 80 (3), 3229–3281. https://doi.org/10.1007/s11227-023-05579-4
20. Saremi, S., Mirjalili, S., Lewis, A. (2017). Grasshopper Optimisation Algorithm: Theory and application. Advances in Engineering Software, 105, 30–47. https://doi.org/10.1016/j.advengsoft.2017.01.004
21. Braik, M. S. (2021). Chameleon Swarm Algorithm: A bio-inspired optimizer for solving engineering design problems. Expert Systems with Applications, 174, 114685. https://doi.org/10.1016/j.eswa.2021.114685
22. Yapici, H., Cetinkaya, N. (2019). A new meta-heuristic optimizer: Pathfinder algorithm. Applied Soft Computing, 78, 545–568. 23. Duan, H., Qiao, P. (2014). Pigeon-inspired optimization: a new swarm intelligence optimizer for air robot path planning. International Journal of Intelligent Computing and Cybernetics, 7 (1), 24–37. https://doi.org/10.1108/ijicc-02-2014-0005
24. Yang, X.-S.; Durand-Lose, J., Jonoska, N. (Eds.) (2012). Flower Pollination Algorithm for Global Optimization. Unconventional Computation and Natural Computation, 240–249. https://doi.org/10.1007/978-3-642-32894-7_27
25. Gomes, G. F., da Cunha, S. S., Ancelotti, A. C. (2018). A sunflower optimization (SFO) algorithm applied to damage identification on laminated composite plates. Engineering with Computers, 35 (2), 619–626. https://doi.org/10.1007/s00366-018-0620-8
26. Mehrabian, A. R., Lucas, C. (2006). A novel numerical optimization algorithm inspired from weed colonization. Ecological Informatics, 1 (4), 355–366. https://doi.org/10.1016/j.ecoinf.2006.07.003
27. Qi, X., Zhu, Y., Chen, H., Zhang, D., Niu, B.; Huang, D. S., Jo, K. H., Zhou, Y. Q., Han, K. (Eds.) (2013). An Idea Based on Plant Root Growth for Numerical Optimization. Intelligent Computing Theories and Technology. Berlin, Heidelberg: Springer, 571–578. https://doi.org/10.1007/978-3-642-39482-9_66
28. Bezuhlyi, V., Oliynyk, V., Romanenko, I., Zhuk, O., Kuzavkov, V., Borysov, O. et al. (2021). Development of object state estimation method in intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 5 (3 (113)), 54–64. https://doi.org/10.15587/1729-4061.2021.239854
29. Mahdi, Q. A., Shyshatskyi, A., Prokopenko, Y., Ivakhnenko, T., Kupriyenko, D., Golian, V. et al. (2021). Development of estimation and forecasting method in intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 3 (9 (111)), 51–62. https://doi.org/10.15587/1729-4061.2021.232718
30. Sova, O., Radzivilov, H., Shyshatskyi, A., Shevchenko, D., Molodetskyi, B., Stryhun, V. et al. (2022). Development of the method of increasing the efficiency of information transfer in the special purpose networks. Eastern-European Journal of Enterprise Technologies, 3 (4 (117)), 6–14. https://doi.org/10.15587/1729-4061.2022.259727

31. Zhang, H., Zhu, Y., Chen, H. (2013). Root growth model: a novel approach to numerical function optimization and simulation of plant root system. Soft Computing, 18 (3), 521–537. https://doi.org/10.1007/s00500-013-1073-z
32. Labbi, Y., Attous, D. B., Gabbar, H. A., Mahdad, B., Zidan, A. (2016). A new rooted tree optimization algorithm for economic dispatch with valve-point effect. International Journal of Electrical Power & Energy Systems, 79, 298–311. https://doi.org/10.1016/j.ijepes.2016.01.028
33. Murase, H. (2000). Finite element inverse analysis using a photosynthetic algorithm. Computers and Electronics in Agriculture, 29 (1-2), 115–123. https://doi.org/10.1016/s0168-1699(00)00139-3
34. Zhao, S., Zhang, T., Ma, S., Chen, M. (2022). Dandelion Optimizer: A nature-inspired metaheuristic algorithm for engineering applications. Engineering Applications of Artificial Intelligence, 114, 105075. https://doi.org/10.1016/j.engappai.2022.105075
35. Paliwal, N., Srivastava, L., Pandit, M. (2020). Application of grey wolf optimization algorithm for load frequency control in multi-source single area power system. Evolutionary Intelligence, 15 (1), 563–584. https://doi.org/10.1007/s12065-020-00530-5
36. Dorigo, M., Blum, C. (2005). Ant colony optimization theory: A survey. Theoretical Computer Science, 344 (2-3), 243–278. https://doi.org/10.1016/j.tcs.2005.05.020
37. Poli, R., Kennedy, J., Blackwell, T. (2007). Particle swarm optimization: an overview. Swarm Intelligence, 1 (1), 33–57. https://doi.org/10.1007/s11721-007-0002-0
38. Bansal, J. C., Sharma, H., Jadon, S. S., Clerc, M. (2014). Spider Monkey Optimization algorithm for numerical optimization. Memetic Computing, 6 (1), 31–47. https://doi.org/10.1007/s12293-013-0128-0
39. Yeromina, N., Kurban, V., Mykus, S., Peredrii, O., Voloshchenko, O., Kosenko, V. et al. (2021). The Creation of the Database for Mobile Robots Navigation under the Conditions of Flexible Change of Flight Assignment. International Journal of Emerging Technology and Advanced Engineering, 11 (5), 37–44. https://doi.org/10.46338/ijetae0521_05
40. Maccarone, A. D., Brzorad, J. N., Stone, H. M. (2008). Characteristics and Energetics Of Great Egret and Snowy Egret Foraging Flights. Waterbirds, 31 (4), 541–549. https://doi.org/10.1675/1524-4695-31.4.541
41. Ramaji, I. J., Memari, A. M. (2018). Interpretation of structural analytical models from the coordination view in building information models. Automation in Construction, 90, 117–133. https://doi.org/10.1016/j.autcon.2018.02.025
42. Pérez-González, C. J., Colebrook, M., Roda-García, J. L., Rosa-Remedios, C. B. (2019). Developing a data analytics platform to support decision making in emergency and security management. Expert Systems with Applications, 120, 167–184. https://doi.org/10.1016/j.eswa.2018.11.023
43. Chen, H. (2018). Evaluation of Personalized Service Level for Library Information Management Based on Fuzzy Analytic Hierarchy Process. Procedia Computer Science, 131, 952–958. https://doi.org/10.1016/j.procs.2018.04.233
44. Chan, H. K., Sun, X., Chung, S.-H. (2019). When should fuzzy analytic hierarchy process be used instead of analytic hierarchy process? Decision Support Systems, 125, 113114. https://doi.org/10.1016/j.dss.2019.113114

45. Osman, A. M. S. (2019). A novel big data analytics framework for smart cities. Future Generation Computer Systems, 91, 620–633. https://doi.org/10.1016/j.future.2018.06.046
46. Nechyporuk, O., Sova, O., Shyshatskyi, A., Kravchenko, S., Nalapko, O., Shknai, O. et al. (2023). Development of a method of complex analysis and multidimensional forecasting of the state of intelligence objects. Eastern-European Journal of Enterprise Technologies, 2 (4 (122)), 31–41. https://doi.org/10.15587/1729-4061.2023.276168
47. Merrikh-Bayat, F. (2015). The runner-root algorithm: A metaheuristic for solving unimodal and multimodal optimization problems inspired by runners and roots of plants in nature. Applied Soft Computing, 33, 292–303. https://doi.org/10.1016/j.asoc.2015.04.048
48. Poliarush, O., Krepych, S., Spivak, I. (2023). Hybrid approach for data filtering and machine learning inside content management system. Advanced Information Systems, 7 (4), 70–74. https://doi.org/10.20998/2522-9052.2023.4.09
49. Balochian, S., Baloochian, H. (2019). Social mimic optimization algorithm and engineering applications. Expert Systems with Applications, 134, 178–191. https://doi.org/10.1016/j.eswa.2019.05.035
50. Lenord Melvix, J. S. M. (2014). Greedy Politics Optimization: Metaheuristic inspired by political strategies adopted during state assembly elections. 2014 IEEE International Advance Computing Conference (IACC). IEEE, 1157–1162. https://doi.org/10.1109/iadcc.2014.6779490
51. Moosavian, N., Roodsari, B. K. (2014). Soccer League Competition Algorithm, a New Method for Solving Systems of Nonlinear Equations. International Journal of Intelligence Science, 4 (1), 7–16. https://doi.org/10.4236/ijis.2014.41002
52. Hayyolalam, V., Pourhaji Kazem, A. A. (2020). Black Widow Optimization Algorithm: A novel meta-heuristic approach for solving engineering optimization problems. Engineering Applications of Artificial Intelligence, 87, 103249. https://doi.org/10.1016/j.engappai.2019.103249
53. Abualigah, L., Yousri, D., Abd Elaziz, M., Ewees, A. A., Al-qaness, M. A. A., Gandomi, A. H. (2021). Aquila Optimizer: A novel meta-heuristic optimization algorithm. Computers & Industrial Engineering, 157, 107250. https://doi.org/10.1016/j.cie.2021.107250
54. Hodlevskyi, M., Burlakov, G. (2023). Information technology of quality improvement planning of process subsets of the spice model. Advanced Information Systems, 7 (4), 52–59. https://doi.org/10.20998/2522-9052.2023.4.06
55. Askari, Q., Younas, I., Saeed, M. (2020). Political Optimizer: A novel socio-inspired meta-heuristic for global optimization. Knowledge-Based Systems, 195, 105709. https://doi.org/10.1016/j.knosys.2020.105709
56. Mohamed, A. W., Hadi, A. A., Mohamed, A. K. (2019). Gaining-sharing knowledge based algorithm for solving optimization problems: a novel nature-inspired algorithm. International Journal of Machine Learning and Cybernetics, 11 (7), 1501–1529. https://doi.org/10.1007/s13042-019-01053-x
57. Gödri, I., Kardos, C., Pfeiffer, A., Váncza, J. (2019). Data analytics-based decision support workflow for high-mix low-volume production systems. CIRP Annals, 68 (1), 471–474. https://doi.org/10.1016/j.cirp.2019.04.001
58. Harding, J. L. (2013). Data quality in the integration and analysis of data from multiple sources: some research challenges. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XL-2/W1, 59–63. https://doi.org/10.5194/isprsarchives-xl-2-w1-59-2013

59. Orouskhani, M., Orouskhani, Y., Mansouri, M., Teshnehlab, M. (2013). A Novel Cat Swarm Optimization Algorithm for Unconstrained Optimization Problems. International Journal of Information Technology and Computer Science, 5 (11), 32–41. https://doi.org/10.5815/ijitcs.2013.11.04
60. Karaboga, D., Basturk, B. (2007). A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. Journal of Global Optimization, 39 (3), 459–471. https://doi.org/10.1007/s10898-007-9149-x
61. Voznytsia, A., Sharonova, N., Babenko, V., Ostapchuk, V., Neronov, S., Feoktystov, S. et al. (2025). Development of methods for intelligent assessment of parameters in decision support systems. Eastern-European Journal of Enterprise Technologies, 4 (4 (136)), 73–82. https://doi.org/10.15587/1729-4061.2025.337528
62. Sova, O., Radzivilov, H., Shyshatskyi, A., Shvets, P., Tkachenko, V., Nevhad, S. et al. (2022). Development of a method to improve the reliability of assessing the condition of the monitoring object in special-purpose information systems. Eastern-European Journal of Enterprise Technologies, 2 (3 (116)), 6–14. https://doi.org/10.15587/1729-4061.2022.254122
63. Litvinenko, O., Kashkevich, S., Shyshatskyi, A., Dmytriieva, O., Neronov, S., Plekhova, G. et al.; Shyshatskyi, A. (Ed.) (2024). Information and control systems: modelling and optimizations. Kharkiv: TECHNOLOGY CENTER PC, 180. https://doi.org/10.15587/978-617-8360-04-7
64. Voznytsia, A., Sharonova, N., Babenko, V., Ostapchuk, V., Neronov, S., Feoktystov, S. et al. (2025). Development of methods for intelligent assessment of parameters in decision support systems. Eastern-European Journal of Enterprise Technologies, 4 (4 (136)), 73–82. https://doi.org/10.15587/1729-4061.2025.337528
65. Nalapko, O., Shyshatskyi, A., Ostapchuk, V., Mahdi, Q. A., Zhyvotovskyi, R., Petruk, S. et al. (2021). Development of a method of adaptive control of military radio network parameters. Eastern-European Journal of Enterprise Technologies, 1 (9 (109)), 18–32. https://doi.org/10.15587/1729-4061.2021.225331
66. Kalantaievska, S., Pievtsov, H., Kuvshynov, O., Shyshatskyi, A., Yarosh, S., Gatsenko, S. et al. (2018). Method of integral estimation of channel state in the multiantenna radio communication systems. Eastern-European Journal of Enterprise Technologies, 5 (9 (95)), 60–76. https://doi.org/10.15587/1729-4061.2018.144085
67. Alieinykov, I., Thamer, K. A., Zhuravskyi, Y., Sova, O., Smirnova, N., Zhyvotovskyi, R. et al. (2019). Development of a method of fuzzy evaluation of information and analytical support of strategic management. Eastern-European Journal of Enterprise Technologies, 6 (2 (102)), 16–27. https://doi.org/10.15587/1729-4061.2019.184394
68. Koshlan, A., Salnikova, O., Chekhovska, M., Zhyvotovskyi, R., Prokopenko, Y., Hurskyi, T. et al. (2019). Development of an algorithm for complex processing of geospatial data in the special-purpose geoinformation system in conditions of diversity and uncertainty of data. Eastern-European Journal of Enterprise Technologies, 5 (9 (101)), 35–45. https://doi.org/10.15587/1729-4061.2019.180197
69. Sova, O., Zhuravskyi, Y., Vakulenko, Y., Shyshatskyi, A., Salnikova, O., Nalapko, O. (2022). Development of methodological principles of routing in networks of special communication in conditions of fire

storm and radio-electronic suppression. EUREKA: Physics and Engineering, 3, 159–166. https://doi.org/10.21303/2461-4262.2022.002434

70. Koval, M., Sova, O., Orlov, O., Shyshatskyi, A., Artabaiev, Y., Shknai, O. et al. (2022). Improvement of complex resource management of special-purpose communication systems. Eastern-European Journal of Enterprise Technologies, 5 (9 (119)), 34–44. https://doi.org/10.15587/1729-4061.2022.266009
71. Fedoriienko, V., Koshlan, O., Kravchenko, S., Shyshatskyi, A., Vasiukova, N., Trotsko, O. et al. (2021). Development of a methodological approach for processing different types of data in systems of special purpose. Technology Audit and Production Reserves, 6 (2 (62)), 18–24. https://doi.org/10.15587/2706-5448.2021.243950
72. Abed, A. A., Repilo, I., Zhyvotovskyi, R., Shyshatskyi, A., Hohoniants, S., Kravchenko, S. et al. (2021). Improvement of the method of estimation and forecasting of the state of the monitoring object in intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 4 (3 (112)), 43–55. https://doi.org/10.15587/1729-4061.2021.237996
73. Shyshatskyi, A., Zhuravskyi, Y., Plekhova, G., Shostak, I., Feoktystova, O., Dmytriieva, O. et al. (2025). Development of a polymodel resource management complex for intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 5 (4 (137)), 41–63. https://doi.org/10.15587/1729-4061.2025.340387
74. Shyshatskyi, A., Plekhova, G., Odarushchenko, E., Miahkykh, H., Feoktystova, O., Shostak, I. et al. (2025). Development of a polymodel complex of information systems resource management. Eastern-European Journal of Enterprise Technologies, 4 (4 (136)), 58–72. https://doi.org/10.15587/1729-4061.2025.335688

Edited by
Yurii Zhuravskyi

INTELLIGENT DECISION SUPPORT SYSTEMS:
METHODS FOR OPTIMIZING AND SUPPORTING MANAGEMENT DECISIONS

Yurii Zhuravskyi, Dmytro Fedorchuk, Oleksandr Perehuda, Mykola Romanchuk, Roman Stavisiuk, Dmitro Stupak, Serhii Neronov, Ganna Plekhova, Olena Feoktystova, Igor Shostak, Anastasiia Voznytsia, Andrii Shyshatskyi, Danylo Pliekhov, Oleksii Nalapko, Yuliia Vakulenko, Andrii Lebedynskyi, Oksana Dmytriieva, Ivan Starynskyi, Oleksandr Zhuk, Pavlo Zhuk, Andrii Veretnov, Olena Shaposhnikova, Yaroslav Melnyk, Oleh Shknai, Illia Dmytriiev, Oleg Sova, Olesia Zhuk, Bohdan Molodetskyi

Monograph

Technical editor I. Prudius
Desktop publishing T. Serhiienko

TECHNOLOGY CENTER PC®
Published in January 2026
Enlisting the subject of publishing No. 4452 – 10.12.2012
Address: Shatylova dacha str., 4, Kharkiv, Ukraine, 61165
