DOI: 10.15587/978-617-8360-13-9.CH1

Svitlana Kashkevich, Iraida Stanovska, Oleh Shknai,
Oleksandr Lytvynenko, Yevhen Artamonov, Andrii Veretnov

CHAPTER 1

SCIENTIFIC METHOD APPARATUS FOR INTELLECTUAL ASSESSMENT OF THE STATE OF COMPLEX SYSTEMS

ABSTRACT

In this section of the research, a scientific and method apparatus for intelligent assessment of the state of complex systems is proposed. The basis of this research is the theory of artificial intelligence, namely evolving artificial neural networks, basic genetic algorithm procedures, neuro-fuzzy expert systems and bio-inspired algorithms. In the course of the research, the authors proposed:

– the method approach to assessing the state of hierarchical systems using a metaheuristic algorithm;

– the method of analysis and forecasting of the state of multidimensional objects using a metaheuristic algorithm;

– the method of increasing the reliability of the assessment of the object state.

The use of the proposed scientific and method apparatus will allow:

– to reduce the probability of premature convergence of the metaheuristic algorithm;

– to maintain a balance between the speed of convergence of the metaheuristic algorithm and diversification;

– to take into account the type of uncertainty and noisy data of the metaheuristic algorithm;

– to take into account the available computing resources of the state analysis system of the analysis object;

– to take into account the priority of search by swarm agents of the metaheuristic algorithm;

– to carry out the initial display of flock individuals taking into account the uncertainty type;

– to conduct accurate training of individuals of metaheuristic algorithms;

– to conduct a local and global search taking into account the degree of noise of the data on the state of the analysis object;

– to apply as a universal tool for solving the task of analyzing the state of analysis objects due to the hierarchical description of analysis objects;

– to check the reliability of the obtained results;

– to increase the reliability of the assessment of the objects state of analysis due to the construction of object and relational models of their state with different degrees of hierarchy;

– to avoid the local extremum problem.

KEYWORDS

Bio-inspired algorithms, multi-agent systems, combined systems, reliability and efficiency.

1.1 A METHOD APPROACH TO ASSESSING THE STATE OF HIERARCHICAL SYSTEMS USING A METAHEURISTIC ALGORITHM

The process of assessing complex and hierarchical systems is a complex process of determining a set of possible states for a wide range of tasks, including for making management decisions [1–10].

State assessments of complex and hierarchical systems are discontinuous, undifferentiated and multimodal. Considering the above, it is impractical to use classic gradient deterministic algorithms [11–22] to solve this type of problem.

The most common approaches to assessing the state of hierarchical systems are swarm intelligence algorithms (swarm algorithms). The most famous swarm algorithms are particle swarm optimization algorithm, artificial bee colony algorithm, firefly swarm algorithm, ant colony optimization algorithm, wolf swarm optimization algorithm and sparrow swarm algorithm [23–40].

However, most of the basic bio-inspired algorithms mentioned above are unable to maintain a balance between research and use, resulting in unsatisfactory performance for real-world complex optimization tasks.

This encourages the implementation of various strategies to improve the convergence speed and accuracy of the underlying bio-inspired algorithms. Therefore, research devoted to the development of new approaches to assessing the state of complex hierarchical systems is relevant.

An analysis of works [41–71] showed that the common shortcomings of the above-mentioned researches are:

– no possibility of hierarchical processing of various data types;

– the lack of possibility of additional involvement of necessary computing resources of the system;

– a failure to take into account the type of uncertainty and noisy data about the information circulating in the system;

– the lack of deep learning mechanisms of knowledge bases;

– the lack of search priority in a certain direction.

The aim of the research is the development of method approach to assessing the state of hierarchical systems using a metaheuristic algorithm. This will allow to increase the efficiency of assessment of the state of hierarchical systems with a given reliability and the development of subsequent management decisions. This will make it possible to develop software for intelligent decision-making support systems.

To achieve the aim, the following tasks were set:

– to determine the procedures for implementing a method approach to assessing the state of hierarchical systems;

– to lead an example of assessing hierarchical systems while analyzing the operational situation of a group of troops (forces) using the proposed method approach.

In this research, an optimizer based on simulating the behavior of antlions is proposed – a population-based stochastic algorithm that uses antlion agents (ALA) as search agents. The antlion algorithm is based on the imitation of the way antlions dig anthills to hunt ants in the natural environment.

The method approach to assessing the state of hierarchical systems using the metaheuristic algorithm consists of the following sequence of actions:

Step 1. Input of initial data. At this stage, the main parameters of the algorithm are determined, such as:

– the type of task being solved;

– the number of agents in the population;

– the number of variables characterizing the task being solved;

– available computing resources of the system;

– the complexity of the hierarchical system to be assessed;

– the parameters of the improved genetic algorithm (selection parameters, mutations), the number of individuals;

– the type of uncertainty about the hierarchical system (complete uncertainty, partial uncertainty, complete awareness);

– volume and type of research sample;

– volume and type of test sample;

– artificial neural network architecture, etc.

Step 2. Creation of ALA flock. Initialization of the ALA population X_i (i=1, 2, ..., n) takes place. The set of ALA form a population, which is described by the matrix X. The initial population of ALA in this algorithm is generated taking into account the uncertainty about the state of the hierarchical system based on the constraints of the problem under consideration. The members of the ALA population are search agents in the solution space, providing candidate values for the problem variables based on their positions in the search space. Mathematically, each member of the general set is a vector, the number of elements of which is equal to the number of task variables.

ALA is issued taking into account the uncertainty about the state of a complex hierarchical system based on basic system models and circulating data models [2, 19, 21] (1.1):

$$X = \begin{bmatrix} X_1 \\ \vdots \\ X_i \\ \vdots \\ X_N \end{bmatrix}_{N\times m} = \begin{bmatrix} x_{1,1}\times\iota_{1,1} & \cdots & x_{1,d}\times\iota_{1,d} & \cdots & x_{1,m}\times\iota_{1,m} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{i,1}\times\iota_{i,1} & \cdots & x_{i,d}\times\iota_{i,d} & \cdots & x_{i,m}\times\iota_{i,m} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{N,1}\times\iota_{N,1} & \cdots & x_{N,d}\times\iota_{N,d} & \cdots & x_{N,m}\times\iota_{N,m} \end{bmatrix}_{N\times m}. \quad (1.1)$$

where X is the ALA population matrix; X_i is the i-th member of the ALA swarm (solution candidate); $x_{i,d}$ is the d-th dimension in the search space (decision variable); N is the number of ALA; m is the number of decision variables describing the state complex hierarchical system.

Step 3. Numbering of ALA in the flock, i, $i \in [0, S]$. At this stage, each ALA is assigned a serial number. This makes it possible to determine the parameters of finding a solution for each individual in the flock.

Step 4. Determination of the initial ALA speed.

Initial speed v_0 of each ALA is defined by the following expression:

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

The process of updating the ALA population is based on the simulation of two strategies of the exploration phase and the exploitation phase.

Step 5. Validation of each ALA.

The relevance of each search ALA is determined in each iteration using the improved genetic algorithm proposed in work [26] and comparison of the obtained values with standardized functions. The fitness value of each ALA in the search swarm (each row in the X matrix) is measured and compared with the fitness of the remaining ALA (the other rows of the X matrix).

Step 6. Preliminary assessment of the ALA search area. In this procedure, the natural language search area is determined precisely by the halo of the existence of the ALA, where the ants live.

Step 7. Classification of ant nests.

The location of the best anthill (thus, the smallest anthill with the least number of ants) is considered to be (FS_{ht}), which is nearby and requires the least amount of energy to find and retrieve it. The largest anthill, with the largest number of ants, will be denoted as FS_{at}.

Other single ants will be denoted as FS_{nt}:

$$FS_{ht} = FS(\text{sorte\_index}(0,8)), \tag{1.3}$$

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

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

Step 8. Determining the number of available computing resources of the system.

At this stage, the amount of computing resources available for calculations is determined. In accordance with the provisions outlined in Step 4, the concept of updating the provisions of the ALA is chosen.

Step 9. Reconnaissance (surrounding the prey).

The position of ALA is directly dependent on the position of their prey (in our case, these are ants). The position of each ant in each dimension is updated using a random walk. This random walk is described by the following mathematical expression:

$$x(t)=\left[0,cumsum\left(2t\left(t_1\right)-1\right),cumsum\left(2t\left(t_2\right)-1\right),\ldots,cumsum\left(2t\left(t_T\right)-1\right)\right], \quad (1.6)$$

where T is the maximum number of iterations; t_i is the t-th iteration; *cumsum* is the cumulative summation; $r(t)$ is a random function calculated as follows:

$$r(t)=\begin{cases}1, & rand \geq 0.5;\\ 0, & rand < 0.5,\end{cases} \quad (1.7)$$

where t is the iteration index; *rand* is a randomly generated number in [0, 1].

The total population of ants on the search plane, on which ALA hunting takes place, is described by the matrix:

$$M_{ant}=\begin{bmatrix}\overrightarrow{Ant_1}\\ \overrightarrow{Ant_2}\\ .\\ .\\ \overrightarrow{Ant_n}\end{bmatrix}, \quad (1.8)$$

where n is the number of ants in the population.

The value of anthills in this research is identified as the value of the decision made in relation to the optimization task, stored in the following vector:

$$M_{oa}=\begin{bmatrix}f\left(\overrightarrow{Ant_1}\right)\\ f\left(\overrightarrow{Ant_2}\right)\\ .\\ .\\ .\\ f\left(\overrightarrow{Ant_n}\right)\end{bmatrix}, \quad (1.9)$$

Each antlion is represented by a pose vector and a target vector as follows:

$$\overrightarrow{Antlion_i}=\left[A_{i,1},A_{i,1},\ldots,A_{i,d}\right], \quad (1.10)$$

where $Antlion_i$ is the i-th ant lion; $A_{i,d}$ is the position of the i-th ant in the d-th dimension.

$$M_{Antlion}=\begin{bmatrix}\overrightarrow{Antlion_1}\\ \overrightarrow{Antlion_2}\\ .\\ .\\ \overrightarrow{Antlion_n}\end{bmatrix}, \quad (1.11)$$

$$M_{oal} = \begin{bmatrix} f\left(\overrightarrow{Antlion_1}\right) \\ f\left(\overrightarrow{Antlion_2}\right) \\ \cdot \\ \cdot \\ \cdot \\ f\left(\overrightarrow{Antlion_n}\right) \end{bmatrix}, \tag{1.12}$$

where n is the number of ants in the population.

Step 10. Verification of hitting the global optimum. At this stage, the condition for the algorithm to reach the global optimum is checked according to the specified criterion for assessing the state of complex hierarchical systems.

Step 11. Global restart procedure.

The restart procedure can effectively improve the ability of the algorithm to go beyond the current optimum and improve the exploratory ability of the algorithm. If the optimal population of the algorithm remains unchanged after T iterations, the population is likely to fall into a local optimum. Thus, the candidate solution will be initialized randomly to accelerate the departure from the global optimum.

Step 12. Hunting phase (exploitation).

To determine the priority of the anthill, the anthill with the highest value of ant pheromone (with more ants) is chosen for the ALA attack:

$$P_{ij}^{k} = \begin{cases} \dfrac{\left(\tau_{ij}\right)^{\alpha}\left(\eta_{ij}\right)^{\beta}}{\sum\limits_{h \notin tabu_k}\left(\tau_{ih}\right)^{\alpha}\left(\eta_{ih}\right)^{\beta}}, & j \notin tabu_k; \\ 0, & \text{otherwise}, \end{cases} \tag{1.13}$$

where τ_{ij} and η_{ij} is the intensity of pheromones and the cost of the route between anthills i and j, respectively. Relative value τ_{ij} and η_{ij} is determined by the parameters α and β, respectively. $tabu_k$ is a list of unavailable routes (visited nodes) for ALA k.

Step 13. Checking the stop criterion. The algorithm terminates when the maximum number of iterations is completed. Otherwise, the behavior of generating new places and checking conditions is repeated.

Step 14. Learning ALA knowledge bases.

In this research, the learning method based on evolving artificial neural networks developed in the research [2] is used to train the knowledge bases of each ALA. The method is used to change the nature of movement of each ALA, for more accurate analysis results in the future.

The end of algorithm.

A method approach to assessing the state of hierarchical systems using a metaheuristic algorithm is proposed. To determine the effectiveness of the proposed methodical approach, modeling of its work was carried out to solve the task of determining the composition of the operational grouping of troops (forces) and the elements of its operational construction in order to determine the expediency of regrouping troops (forces).

The effectiveness of the method approach is compared with swarm optimization algorithms, using a set of CEC2019 test functions listed in the **Table 1.1**. The efficiency assessment criterion is the speed of decision making (msec) with a given assessment reliability (0.9).

As it can be seen from the **Table 1.1**, increasing the efficiency of assessing the state of hierarchical systems is achieved at the level of 22–25 % due to the use of additional procedures.

It can be seen that the method approach is able to converge to the true value for most unimodal functions with the fastest convergence speed and the highest accuracy, while the convergence results of the ant swarm algorithm are far from satisfactory.

The advantages of the proposed method approach are due to the following:

– the initial position of the ALA is carried out taking into account the type of uncertainty (Step 2) due to the use of appropriate correction coefficients for the degree of awareness of the placement of anthills (in our case, priority search directions), in comparison with works [9, 14, 21];

– the initial speed of each ALA is taken into account (Step 4), which allows to determine the search priority of each ALA in the specified search direction, in comparison with works [9–15];

– the suitability of ALA hunting sites is determined, which reduces the time for assessing the state of the hierarchical system (Step 6), in comparison with works [14, 16, 17];

– the degree of data noise is taken into account in the process of updating the ALA position (Steps 9–12), thereby reducing the time for assessing the state of hierarchical systems, compared to works [9–15];

– the use of the procedure of global restart of the algorithm, which achieves the ability of the algorithm to go beyond the current optimum and improve the research ability of the algorithm (Step 11), which reduces the time for assessing the state of hierarchical systems, compared to works [9–15];

– the universality of solving the task of assessing the state of hierarchical ALA systems due to the hierarchical nature of their description (Steps 1–14, **Table 1.1**), in comparison with works [9, 12–18];

– the possibility of simultaneously searching for a solution in different directions (Steps 1–14, **Table 1.1**);

– the adequacy of the obtained results (Steps 1–14), in comparison with works [9–23];

– the possibility of clarifying the selection of an anthill at the hunting stage (Step 12) due to the ranking of anthills by the level of ant pheromone, in comparison with works [9, 12–18];

– an improved possibility of selecting the best ALA in comparison with random selection due to the use of an improved genetic algorithm (Step 5), in comparison with works [9–15]. This allows to improve the reliability of assessment of the state of hierarchical systems;

– the ability to avoid the local extremum problem (Steps 1–14);

– the possibility of in-depth learning of ALA knowledge bases (Step 14), in comparison with works [9–23].

Table 1.1 Comparison of the proposed method approach with other swarm algorithms for a defined set of test functions

Type of test functions CEC2019	Value	Particle swarm algorithm	Ant colony algo-rithm	Black widow algorithm	Algorithm of a flock of gray wolves	Bee swarm algorithm	Canonical algorithm of ant lions	The proposed method approach
F1	Better	6.2501	4.4884	4.103	4.4136	6.2606	5.0994	**2.7698**
	Average	8.1507	6.2966	5.309	5.1521	6.2606	8.1594	**2.7698**
	Standard	7.33	6.01	6.2	6.4	7.838	7.7192	**3.3712**
F2	Better	545.9192	5.6911	4.8556	4.2197	4.0557	4.2739	**3.2141**
	Average	2689.105	6.91	4.9935	55.3157	4.8087	4.9449	**4.6568**
	Standard	1741.300	0.94	0.02708	106.434	0.33374	0.17328	**0.64381**
F3	Better	2.3979	2.4361	1.9805	1.1634	1.4104	2.9411	**1.4173**
	Average	7.2501	4.4884	4.103	4.4136	6.2606	5.0994	**2.7698**
	Standard	1.9616	1.2868	0.766	2.2871	2.6603	1.0752	**0.62451**
F4	Better	9.2209	10.0576	38.8009	12.248	4.1414	29.1901	**3.9849**
	Average	27.9446	25.5342	57.3153	24,843	28.4397	44.8217	**16.6629**
	Standard	11.0734	8.7901	6.9365	10.7428	15.7463	8.8591	**11.1345**
F5	Better	1.5511	1.4833	29.405	1.1339	1.0497	2.1918	**1.0074**
	Average	11.428	1.7597	72.0211	9.5542	1.115	3.7237	**1.0641**
	Standard	13.5064	0.18663	19.4782	9.0179	0.061305	1.1233	**0.050987**
F6	Better	1.9552	3.1214	8.5015	1.7038	1.3443	6.2846	**1.0041**
	Average	6.7367	5.9982	10.5902	5.2258	3.9784	9.1707	**3.0444**
	Standard	2.4593	1.2484	0.77804	1.7237	1.5453	1.3384	**1.268**
F7	Better	308.3668	249.343	1208.30	309.912	83.4932	399.241	**126.6386**
	Average	1102.474	832.301	1623.67	810.994	894.644	1206.1076	**508.5085**
	Standard	355.712	265.703	130.086	376.011	339.8851	290.3894	**230.6677**
F8	Better	805.925	811.990	837.627	809.783	804.9748	816.6299	**802.0457**
	Average	825.7583	824.501	848.282	824.769	825.9996	832.6092	**815.3867**
	Standard	10.1005	7.4753	6.0262	9.2526	8.9454	8.4370	**9.9757**
F9	Better	1.1616	1.1532	1.5475	1.2025	1.1683	1.3514	**1.0359**
	Average	1.3579	1.4924	1.9341	1.3228	1.3786	1.6591	**1.1378**
	Standard	0.0999	0.1697	0.1336	0.0928	0.1442	0.1545	**0.0522**
F10	Better	1436.99	1274.08	1641.75	1325.81	1204.19	1818.69	**1139.99**
	Average	1937.94	1861.89	2362.49	1828.64	1819.19	2217.23	**1505.39**
	Standard	349.35	275.94	194.21	344.89	267.68	216.92	**229.69**

The disadvantages of the proposed method approach should include:

– the loss of informativeness while processing various types of data due to the construction of the membership function;

– lower accuracy of processing one type of data due to gradient search;

– the loss of credibility of the obtained solutions while searching for a solution in several directions at the same time;

– lower accuracy of assessment compared to other assessment approaches.

The specified method approach will allow:

– to assess the state of complex hierarchical systems;

– to determine effective measures to improve the efficiency of assessing the state of complex hierarchical systems while maintaining the given reliability;

– to reduce the use of computing resources of decision-making support systems.

The limitations of the research are the need to have an initial database on the state of hierarchical systems, the need to take into account the time delay for collection and proving information from intelligence sources.

It is advisable to use the proposed method approach to solve the problems of assessing the state of complex hierarchical systems in conditions of uncertainty and risks characterized by a high degree of complexity.

This research is a further development of researches aimed at developing method principles for increasing the efficiency of processing various types of data, which were published earlier [2, 4–6, 21–23].

The directions of further research should be aimed at reducing computing costs when processing various types of data in special purpose systems.

1.2 THE DEVELOPMENT OF A METHOD FOR ANALYZING AND FORECASTING THE STATE OF MULTIDIMENSIONAL OBJECTS USING A METAHEURISTIC ALGORITHM

The Butterfly Swarm Algorithm (BSA) is based on the behavior of a swarm of butterflies in search of food. As a rule, butterflies can determine the source of the aroma accurately and distinguish between different aromas. Butterflies move from their place to other places with more nectar. Butterflies produce scent as they move to share their current location and personal information with other butterflies.

The inspiration and behavior of butterflies can in this research be formulated as an optimization method, where butterflies represent search agents and produced aromas characterize the value of optimization.

In BSA, butterfly agents (BA) can generate a scent/fitness value with some strength to distinguish it from other scents. This behavior can assist other BA in updating their position in the search space. Once BA that finds the best nectar in the search space produces a scent, all neighboring BA move to the best location for the BA. This kind of mechanism update is called a global search in BSA.

On the other hand, BA will move randomly in the search space if the scents of other BA are detected, known as local search.

The method of analyzing and forecasting the state of multidimensional objects using the metaheuristic algorithm consists of the following sequence of actions:

Step 1. Input of initial data. At this stage, the main parameters of the algorithm are determined, such as:

– the type of task being solved;

– the number of BA in the population;

– the number of variables characterizing the task of analysis and forecasting of multidimensional objects to be solved;

– available computing resources of the system of analysis and forecasting of multidimensional objects;

– the complexity of multidimensional objects to be assessed;

– the parameters of the improved genetic algorithm (selection parameters, mutations), the number of individuals to be selected;

– the type of uncertainty about the state of multidimensional objects (complete uncertainty, partial uncertainty and complete awareness);

– the volume and type of training sample for artificial neural networks;

– the volume and type of test sample for artificial neural networks;

– artificial neural network architecture, etc.

Step 2. Creation of a BA flock. Initialization of the primary (initial) BA population X_i ($i=1, 2, \ldots, n$) takes place. All BAs form a population (flock), which is determined by the matrix X. The initial population (flock) of BA in this research is generated taking into account the uncertainty about the state of multidimensional objects based on the constraints of the problem under consideration (analysis and/or prediction):

$$X=\begin{bmatrix} X_1 \\ \vdots \\ X_i \\ \vdots \\ X_N \end{bmatrix}_{N\times m} = \begin{bmatrix} x_{1,1}\times\iota_{1,1} & \cdots & x_{1,d}\times\iota_{1,d} & \cdots & x_{1,m}\times\iota_{1,m} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{i,1}\times\iota_{i,1} & \cdots & x_{i,d}\times\iota_{i,d} & \cdots & x_{i,m}\times\iota_{i,m} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{N,1}\times\iota_{N,1} & \cdots & x_{N,d}\times\iota_{N,d} & \cdots & x_{N,m}\times\iota_{N,m} \end{bmatrix}_{N\times m}. \quad (1.14)$$

where X is the matrix describing the BA population on the problem solving plane; X_i is the i-th BA that is a solution candidate; $x_{i,d}$ is the d-th dimension of the multidimensional object in the solution search space; N is the number of BA in the population (flocks); m is the number of decision variables describing the state of a multidimensional object.

Step 3. Assigning a serial number to each BA in the flock, i, $i \in [0, S]$. This step allows to determine the parameters of finding a solution for each BA in the population.

Step 4. Setting the initial speed of each BA.

Initial speed for an individual BA v_0 is described by the following mathematical expression:

$$v_i = \left(v_1, v_2, \ldots, v_S\right), v_i = v_0. \quad (1.15)$$

The population update of each BA in this research is determined in order to determine the balance between the two major procedures of exploration and exploitation. These procedures define the global and local search process. The process of updating the BA population is based on the simulation of two strategies of the exploration phase and the exploitation phase.

Step 5. Selection of BA for solving tasks.

The suitability for solving tasks of each BA is determined in each iteration using the improved genetic algorithm proposed in work [26] and comparing the obtained values with standardized functions. The fitness of the BA in the search flock (row in the X matrix) is measured and compared with the fitness of the rest of the BA (the other rows of the X matrix).

Step 6. Preliminary assessment of the BA search area. In this procedure, the search area in natural language is determined precisely by the halo of BA existence, where butterflies live.

Step 7. Classification of nectar sources for BA.

The location of the best cluster of nectar sources for BA is considered to be (FS_{ht}), which is nearby and requires the least amount of energy to find and collect it. Let's denote the most distant clusters of nectar sources as FS_{at}.

Other single sources of nectar will be denoted as FS_{nt}:

$$FS_{ht} = FS(\text{sorte\_index}(0,7)), \quad (1.16)$$

$$FS_{at}(1:3) = FS(\text{sorte\_index}(1:4)), \quad (1.17)$$

$$FS_{nt}(1:NP\text{-}4) = FS(\text{sorte\_index}(2:NP)). \quad (1.18)$$

Step 8. Determining the intensity of the aroma of nectar sources.

The aroma intensity for BA is mathematically modeled as follows:

$$pf_i = cI^a, \quad (1.19)$$

where pf_i is the aroma strength from the i-th BA; I is the stimulus intensity; c is the sensory modality; a is an indicator of the degree of dependence on the modality.

Step 9. Determination of the number of available computing resources of the system.

At this stage, the amount of computing resources available for calculations is determined. In accordance with the provisions outlined in Step 4, the concept of updating the BA provision is chosen.

Step 10. Exploration of nectar sources (global search).

The location of each BA is represented as a vector of certain problem values. This BA location can be updated by trying to find a better location using the following formula:

$$x_i^{t+1} = x_i^t + F_i^{t+1}, \tag{1.20}$$

where x_i^t is the current position of BA i in iteration t; x_i^{t+1} is the following position of the i-th BA; F_i^{t+1} is the fragrance x_i used to update its position during iterations.

In the global search, the i-th BA moves to the strongest BA g^*, which can be represented as:

$$F_i^{t+1} = \left(r^2 \times g^* - x_i^t\right) \times pf_i, \tag{1.21}$$

where r is a random number in the range [0, 1].

Step 11. Verification of hitting the global optimum. At this stage, the condition of the algorithm hitting the global optimum is checked according to the defined criterion for assessing the state of the multidimensional object.

Step 12. Global restart procedure.

The restart procedure can effectively improve the ability of the algorithm to go beyond the current optimum and improve the exploratory ability of the algorithm. If the optimal population of the algorithm remains unchanged after T iterations, the population is likely to fall into a local optimum. Thus, the candidate solution will be initialized randomly to accelerate the departure from the global optimum.

Step 13. Nectar extraction phase (exploitation).

In the local search, the movement of BA updates can be formulated as follows:

$$F_i^{t+1} = \left(r^2 \times x_j^t - x_k^t\right) \times pf_i, \tag{1.23}$$

where x_j^t and x_k^t is the position of the j-th and k-th BA in the search area. A new parameter, called the switching probability p, is used in BSA to switch the behavior of the algorithm between local and global search to obtain the best balance between exploration and exploitation.

Step 14. Checking the stop criterion. The algorithm terminates if the maximum number of iterations is completed. Otherwise, the behavior of generating new places and checking conditions is repeated.

Step 15. Training of BA knowledge bases.

In this research, the learning method based on evolving artificial neural networks developed in the research [2] is used to learn the knowledge bases of each BA. The method is used to change the nature of movement of each BA, for more accurate analysis results in the future.

The end of algorithm.

This section analyzes the BA behavior in exploration and exploitation due to its significant impact on the solution of analysis and forecasting tasks. The success of metaheuristics lies in their ability to achieve the best balance between exploration and exploitation. These two terms conflict in their search behavior. At the research stage, the algorithm has a high ability to explore and move through unexplored areas of the search space, while at the exploitation stage, the algorithm processes focus on deep search in known areas of the search space.

To determine the effectiveness of the proposed method for analyzing and forecasting the state of multidimensional objects using a metaheuristic algorithm, a simulation of its work was carried out to solve the task of determining the composition of an operational grouping of troops (forces) and elements of its operational construction in order to determine the expediency of regrouping troops (forces).

The effectiveness of the method is compared with metaheuristic algorithms, using a set of CEC2017 test functions listed in the **Table 1.2**. The criterion for assessing efficiency is the speed of decision making (msec) with the given reliability of the evaluation (0.9).

Table 1.2 Comparison of the proposed method with other metaheuristic algorithms for a defined set of test functions

The type of test functions	**Metrics**	**Particle swarm algorithm**	**Ant colony algorithm**	**Black widow algorithm**	**Algorithm of a flock of gray wolves**	**Bee swarm algorithm**	**Canonical butterfly swarm algorithm**	**The proposed method**
1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
CEC 2017-F1	Average	3.61E+09	7.21 E+07	3.54E+09	3.88E+09	7.98E+09	5.38E+09	**2.36E+05**
	Standard	3.15E+09	1.21E+08	1.70E+09	2.37E+09	5.03E+09	3.18E+09	**8.39E+04**
CEC 2017-F2	Average	8.46E+31	4.75E+29	3.91E+31	1.44E+35	3.21E+34	7.36E+34	**1.97E+33**
	Standard	4.49E+32	2.54E+30	1.53E+32	5.49E+35	1.75E+35	3.95E+35	**1.08E+34**
CEC 2OI7-F3	Average	1.57E+05	1.11E+05	5.58E+04	7.19E+04	6.82E+04	6.87E+04	**2.12E+04**
	Standard	5.23E+04	3.72E+04	1.02E+04	1.46E+04	2.04E+04	1.46E+04	**1.05E+04**
CEC 2017-F4	Average	8.33E+02	6.40E+02	1.02E+03	7.45E+02	1.22E+03	9.81E+02	**6.99E+02**
	Standard	1.86E+02	5.06E+01	2.56E+02	1.57E+02	7.74E+02	4.35E+02	**2.34E+02**
CEC 2017-F5	Average	7.35E+02	7.07E+02	7.19E+02	6.50E+02	6.72E+02	6.36E+02	**6.37E+02**
	Standard	2.49E+01	3.09E+01	3.47E+01	4.17E+01	3.66E+01	4.20E+01	**2.09E+01**
CEC 2017-F6	Average	6.57E+02	6.56E+02	6.38E+02	6.16E+02	6.32E+02	6.23E+02	**6.08E+02**
	Standard	1.1 IE+01	8.62E+00	1.03E+01	5.22E+(X)	1.01E+01	8.22E+00	**6.15E+(X)**
CEC 2017-F7	Average	1.26E+03	1.14E+03	1.31E+03	9.24E+02	9.83E+02	9.26E+02	**8.78E+02**
	Standard	7.14E+01	7.87E+01	1.30E+02	8.40E+01	5.10E+01	5.80E+01	**3.01E+01**
CEC 2017-F8	Average	9.65E+02	9.63E+02	1.00E+03	9.30E+02	9.48E+02	9.18E+02	**9.00E+02**
	Standard	1.82E+01	2.78E+01	3.37E+01	4.64E+01	2.80E+01	2.67E+01	**1.97E+01**

Continuation of Table 1.2

1	2	3	4	5	6	7	8	9
CEC 2017-F9	Average	8.82E+03	6.16E+03	6.71E+03	3.55E+03	5.40E+03	4.06E+03	**3.18E+03**
	Standard	1.80E+03	9.63E+02	1.33E+03	1.36E+03	1.53E+03	1.54E+03	**7.41E+02**
CEC 2017-F10	Average	7.54E+03	5.19E+03	8.27E+03	4.90E+03	4.64E+03	4.74E+03	**3.52E+03**
	Standard	4.70E+02	7.04E+02	3.05E+02	1.46E+03	1.15E+03	1.59E+03	**6.20E+02**
CEC 2017-F11	Average	3.48E+03	2.53E+03	2.20E+03	6.68E+03	8.37E+03	1.19E+04	**131E+03**
	Standard	1.35E+03	1.10E+03	9.02E+02	4.18E+03	4.34E+03	6.41E+03	**1.88E+02**
CEC 2017-F12	Average	1.80E+09	7.55E+07	1.62E+09	1.16E+09	5.92E+08	5.87E+07	**6.77E+07**
	Standard	9.48E+08	3.90E+07	9.40E+08	6.26E+08	1.96E+09	8.65E+07	**5.88E+07**
CEC 2017-F13	Average	7.05E+07	2.18E+04	1.15E+08	8.35E+07	4.58E+04	8.45E+04	**7.63E+04**
	Standard	1.39E+08	2.43E+04	1.36E+08	2.11E+08	4.80E+04	6.59E+04	**7.05E+04**
CEC 2017-F14	Average	3.06E+06	3.07E+05	1.10E+06	1.04E+06	4.25E+05	5.37E+05	**3.55E+05**
	Standard	4.63E+06	3.06E+05	1.07E+06	1.07E+06	5.26E+05	5.94E+05	**3.42E+05**
CEC 2017-F15	Average	8.43E+03	2.81E+05	1.05E+06	2.28E+05	2.96E+04	4.87E+04	**5.04E+04**
	Standard	3.08E+03	1.57E+05	1.10E+06	1.29E+05	2.27E+04	4.26E+04	**5.56E+04**
CEC 2017-F16	Average	3.60E+03	3.56E+03	4.12E+03	3.86E+03	3.02E+03	3.05E+03	**3.19E+03**
	Standard	4.40E+02	1.90E+02	8.26E+02	5.61E+02	4.96E+02	4.26E+02	**4.50E+02**
CEC 2017-F17	Average	2.82E+03	2.58E+03	2.90E+03	3.04E+03	2.79E+03	2.34E+03	**2.67E+03**
	Standard	2.71E+02	1.65E+02	3.62E+02	3.13E+02	3.04E+02	1.97E+02	**3.08E+02**
CEC 2017-F18	Average	4.28E+06	4.65E+06	3.70E+07	1.15E+07	1.61E+06	1.84E+06	**1.83E+06**
	Standard	3.92E+06	2.52E+06	4.64E+07	8.25E+06	1.44E+06	1.40E+06	**1.56E+06**
CEC 2017-F19	Average	9.03E+05	3.77E+03	2.12E+07	7.52E+06	3.31E+05	1.30E+05	**4.71E+05**
	Standard	1.08E+06	2.05E+03	1.84E+07	5.42E+06	9.21E+05	2.12E+05	**5.42E+05**
CEC 2017-F20	Average	3.26E+03	2.99E+03	2.94E+03	2.96E+03	3.00E+03	2.85E+03	**2.79E+03**
	Standard	2.52E+02	1.10E+02	2.03E+02	1.99E+02	2.47E+02	2.11E+02	**1.86E+02**
CEC 2017-F21	Average	2.20E+03	2.20E+03	2.20E+03	2.20E+03	2.20E+03	2.20E+03	**2.20E+03**
	Standard	1.60E+00	3.25E-06	2.98E-02	1.15E+00	1.49E-01	5.43E-03	**1.26E-03**

As it can be seen from the **Table 1.2**, increasing the efficiency of assessing the state of hierarchical systems is achieved at the level of 14–16 % due to the use of additional procedures.

The method converges to the true value for most unimodal functions with the fastest convergence speed and the highest accuracy, while the convergence results of the black widow algorithm are far from satisfactory.

The advantages of the proposed method are due to the following:

– the initial display of BA on the plane of multidimensional objects is carried out taking into account the type of uncertainty (Step 2) due to the use of appropriate correction coefficients for the degree of awareness of the location of nectar sources (in our case, priority search directions), in comparison with works [9, 14, 21];

– adjusting the initial speed of the BA (Step 4) allows to determine the priority of the search, in comparison with works [9–15];

– the suitability of BA nectar collection sites is determined, which reduces the time for assessing the state of multidimensional objects (Step 6), in comparison with works [14, 16, 17];

– the presence of the possibility of global restart of the algorithm, which enables the algorithm to go beyond the current optimum and improve the algorithm's research ability (Step 11), which reduces the time needed to assess the state of multidimensional objects, compared to works [9–15];

– the universality of solving the task of assessing the state of hierarchical BA systems due to the hierarchical nature of their description (Steps 1–15, **Table 1.2**), in comparison with works [9, 12–18];

– the possibility of simultaneously searching for a solution in different directions (Steps 1–15, **Table 1.2**);

– the adequacy of the obtained results (Steps 1–15), in comparison with works [9–23];

– the possibility of clarification at the stage of collecting nectar clusters (Step 12) due to the ranking of nectar sources by the level of stimulus intensity, in comparison with works [9, 12–18];

– an improved possibility of selecting the best BA in comparison with traditional selection due to the use of an improved genetic algorithm (Step 5), in comparison with works [9–15];

– the ability to avoid the local extremum problem (Steps 1–15);

– the possibility of in-depth learning of BA knowledge bases (Step 15), in comparison with works [9–23].

The disadvantages of the proposed method of approach should include:

– lower accuracy of finding solutions in one direction due to gradient search;

– the loss of credibility of the obtained solutions while searching for a solution in several directions at the same time;

– lower accuracy of assessment compared to other assessment approaches.

This method will allow:

– to assess the condition of multidimensional objects;

– to determine effective measures to improve the efficiency of assessment of the state of multidimensional objects while maintaining the given reliability;

– to reduce the use of computing resources of decision-making support systems.

The limitations of the research are the need to have an initial database on the state of multidimensional objects, the need to take into account the delay time for collection and proving information from intelligence sources.

1.3 DEVELOPMENT OF A METHOD OF INCREASING THE RELIABILITY OF THE ASSESSMENT OF THE OBJECT STATE

The Rete II method [5, 10, 11] was chosen as the basis for the development of the method of increasing the reliability of the assessment of the object state. Despite the advantages of the Rete II method, including such as increased performance of processing various types of data, the presence of a reverse inference algorithm, the main disadvantages of this method are:

– the work only with clear products;

– low speed of setting up databases;

– compared to other methods, the reliability of the assessment is low, which does not allow it to be used in the processing of various types of data that require a high degree of reliability of decision making.

The essence of the proposed method is the following:

– construction of an object model for the assessment of the object state;

– adjusting the neuro-fuzzy knowledge base by several bio-inspired algorithms and combining the results of parallel work of bio-inspired algorithms using the Merge function;

– construction of a relational model of object state assessment.

Due to this set of procedures, an increase in the reliability of the assessment of the object state is achieved.

The algorithm for the implementation of the proposed method of increasing the reliability of the assessment of the object state consists of the following sequence of actions:

Step 1. Input of initial data.

At this stage, the initial data about the object to be assessed is entered. The starting point about the degree of uncertainty about the object state is determined. The initial data for the work of bio-inspired algorithms are determined, the reliability parameters of the research object are introduced.

Step 2. Formation of the object model.

A formal object model using neuro-fuzzy expert systems has the following form:

$$\left\{P_o\right\} = \left\{\mathrm{Rule}_o\right\}, \tag{1.24}$$

where Rule is a set of rules characterizing the object model of the specified analysis object.

Each Rule is described as follows:

$$\mathrm{Rule}_o = < C \to S >, \tag{1.25}$$

where C is the condition of each rule of the object model, S is the consequence for each rule of the object model.

For this type of object model, it is necessary to correctly present the grammatical structure of the rules with various types of nested conditions. For this purpose, it is proposed to use a recursive mechanism for describing the nodes and terminal vertices of the rule condition tree. In expression (1.25), C is described as:

$$C = < C_l, R, C_r >, \tag{1.26}$$

where C_l is the left node of the condition of each rule of the object model; R is the relationship between the nodes of each rule of the object model; C_r is the right node of the condition of each rule of the object model. Then, let's consider the given parameters:

$$C_l = FC_l \,\|\, \mathrm{Null} \,\|\, C, \tag{1.27}$$

$$C_r = FC_r \,\|\, \mathrm{Null} \,\|\, C, \tag{1.28}$$

where FC_l is the left terminal triple of the condition of each rule of the object model; FC_r is the right final triple of the condition of each rule of the object model. The expressions (1.27) and (1.28) allow to describe the conditions of each rule of the object model with a different degree of hierarchy:

$$FC_l = < L, Z, W >, \tag{1.29}$$

$$FC_r = < L, Z, W >, \tag{1.30}$$

where L is a linguistic variable of the object model; Z is a condition sign $Z = \{<, >, <=, >=, =, != \}$; W is the condition value of the object model, which is determined as follows (1.31):

$$W = L \,\|\, V, \tag{1.31}$$

where L is a linguistic variable of the object model; V is some fixed value (1.32):

$$V = T_i \,\|\, \mathrm{const}, \tag{1.32}$$

where T_i is the value of the fuzzy variable from the term-sets of the linguistic variable, const is a constant. The given set of mathematical expressions (1.24)–(1.31) allows the use of not only linguistic variables, but also classical variables.

Similarly to parameter C of the object model, parameter S is defined as a consequence of the rule of the object model:

$$S = < S_l, R, S_r >, \tag{1.33}$$

where S_l is the left node of the consequence of the rule of the object model; R is the relationship between the nodes of the consequence of the rule; S_r is the right node of the consequence of the rule of the object model:

$$S_l = FS_l \| \text{Null} \| S, \tag{1.34}$$

$$S_r = FS_r \| \text{Null} \| S, \tag{1.35}$$

where FS_l is the left terminal triplet of the consequence of the rule of the object model; FS_r is the right final triple of the consequence of the rule of the object model. Formulas (1.34) and (1.35) allow to describe consequences with different degrees of hierarchy:

$$FS_l = < L, \text{Op}, W >, \tag{1.36}$$

$$FS_r = < L, \text{Op}, W >, \tag{1.37}$$

where L is a linguistic variable of the object model; Op is the type of operation, $\text{Op} = \{:=\}$; W is the value of the consequence of the object model.

Step 3. Setting up a neuro-fuzzy knowledge base and combining the results of their work.

At this stage, the results of parallel work of bio-inspired algorithms are combined using the Merge function.

The function element-by-element compares two binary vectors from the outputs of two bio-inspired algorithms is the bat swarm algorithm and the frog swarm algorithm.

Under the condition that the value of the element at the same position coincides, the given value will be written to this position in the resulting vector. Otherwise, a random number is generated from the interval from 0 to 1 [14–20].

If the value is less than or equal to 0.5, then the corresponding position of the new vector is written by the element from the worst vector. Otherwise, an element from a better vector will be displayed at this location.

Thus, the merge function can be given as follows:

$$merge(S_w, S_b) = \begin{cases} s_i^* = s_{wi} = s_{bi}, & \text{if } s_{wi} = s_{bi}; \\ s_i^* = s_{wi}, & \text{if } s_{wi} \neq s_{bi} \text{ and } rand \leq 0.5; \\ s_i^* = s_{bi}, & \text{if } s_{wi} \neq s_{bi} \text{ and } rand > 0.5, \end{cases} \tag{1.38}$$

where *rand* is a random, uniformly distributed number, $rand \in [0;1]$.

Step 4. Construction of a relational model of object state assessment.

Construction of a relational model of object state assessment in this research is based on the Gray relational analysis (GRA) method. This approach is one of the approaches to multi-criteria analysis, which is used to assess alternatives based on a number of different criteria. This method is used to measure the level of relationship between existing alternatives by calculating the Gray's correlation coefficient (Gray's relational coefficient). The stages of completion by the GRA method are as follows.

Step 4.1. Normalization of relational model data.

Normalization is used to transform data into a single scale that allows better comparison of different variables. The normalization equation in GRA is as follows:

$$X_{norm} = \frac{X_i - X_{\min}}{X_{\max} - X_{\min}}. \tag{1.39}$$

Step 4.2. Formation of the matrix of relational analysis of Gray.

After data normalization, the result of the normalization matrix is the relational analysis of the Gray matrix, namely:

$$G = \begin{bmatrix} x_{11} & \cdots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{m1} & \cdots & x_{mn} \end{bmatrix}, \tag{1.40}$$

where G is the result of the data normalization matrix; m is an existing alternative; n is the existing criterion; x_{ij} is a normalization in the measurement of alternatives.

Step 4.3. Multiplying the GRA matrix by weights.

The next step is to determine the relative weight for each variable. This weight reflects the level of importance of each variable in the GRA analysis. In addition, the GRA method is to give each criterion a weighting that relates to the level of importance of the criterion. Below is the formula for calculations:

$$V_{ij} = g_{i,j} w_j. \tag{1.41}$$

Thus, the following results of the weighted normalization matrix can be formed:

$$V = \begin{bmatrix} v_{1,1} & \cdots & v_{1,n} \\ \vdots & \ddots & \vdots \\ v_{m,1} & \cdots & x_{m,n} \end{bmatrix}. \tag{1.42}$$

Step 4.4. Calculating the value of Gray relational analysis.

In this step, the Gray ratio value is calculated for each variable based on the Gray ratio matrix and the relative weights that were determined using the following equation:

$$GRG_i = \frac{1}{n} \sum_{j=1}^{n} V_{ij}, \tag{1.43}$$

where GRG_i is the value of the Gray ratio (GRG) of the i-th variable to the reference variable.

The end of the algorithm.

To determine the effectiveness of the proposed method of increasing the reliability of the assessment of the object state, a simulation of its work was carried out to solve the task of determining the composition of the operational grouping of troops (forces) and the elements of its operational construction in order to determine the feasibility of regrouping troops (forces).

The following linguistic variables were used to solve the problem:

1. Types of radio emitting devices (RED): The range of permissible values: radio communication devices, radio relay devices, satellite communication devices, air monitoring devices (radar detection devices); devices of radio-electronic countermeasures:

RED="Types of radio emitting devices"={"brigade tactical group", "operational grouping of troops (forces)", "strategic grouping of troops (forces)"}.

2. The types of organizational and staff formations: Range of permissible values: 0÷1:

OSF="Types of organizational and staff formations"={"brigade tactical groups", "operational groupings of troops (forces)", "strategic groupings of troops (forces)"}.

3. The types of control points: Range of permissible values: 0÷1:

CP="Types of control points"={"control points of brigade tactical groups", "control points of operational groups of troops (forces)", "control points of strategic groups of troops (forces)"}.

4. The availability of fortifications: The range of permissible values: 0÷1:

F="Availability of fortifications"={"Fortifications of the first tier", "Fortifications of the first and second tiers", "Fortifications of the first to third tiers"}.

5. The availability of reserves: Range of permissible values: 0÷1:

AR="Availability of reserves"={"reserve brigade tactical group", "two reserve brigade tactical groups", "reserve operational group"}.

6. The operation type: The range of permissible values: 0÷1:

TO="Operation type"={" defensive", "offensive", "counter-offensive"}.

7. The activity of actions in the specified direction: The range of permissible values: 0÷1:

AA="Activity of actions in the specified direction"={"low", "medium", "high"}.

8. The uncertainty of operational situation: The range of permissible values: complete uncertainty ÷ complete awareness:

$$UN = \text{"Uncertainty of operational situation"} = \{\text{"Complete uncertainty", "partial uncertainty", "full awareness"}\}.$$

To simplify further writing, let's denote the vague variables "zero" is "*Z*", "low" is "*L*", "normal" is "*N*", "high" is "*H*".

The membership functions given in the example for the analysis of the operational situation are presented in the specified form according to the formula:

$$1)\ (CP=\text{"H"})\ and\ (OSF=\text{"H"})\ and\ (UN=\text{"H"})\ and\ (AR=\text{"L"}) \rightarrow (REZ=\text{"H"}),$$

$$...$$

$$81)\ (CP=\text{"L"})\ and\ (OSF=\text{"L"})\ and\ (UN=\text{"L"})\ and\ (AR=\text{"H"}) \rightarrow (REZ=\text{"L"}),$$

$$82)\ (F=\text{"L"})\ and\ (AA=\text{"L"})\ and\ (UN=\text{"H"})\ and\ (AR=\text{"H"}) \rightarrow (REZ=\text{"N"}),$$

$$...$$

$$108)\ (SC=\text{"L"})\ and\ (OSF=\text{"L"})\ and\ (UN=\text{"H"})\ and\ (CP=\text{"L"}) \rightarrow (REZ = \text{"N"}).$$

In this example, only a separate part of the rule base of the neuro-fuzzy expert system is given. In the main base of rules there are rules not only with connections of conditions with the help of *T*-norms, but also with the help of *T*-conorms and with negations of conditions.

In the worst case, to find a solution, the system should check all the rules contained in the rule base. Thus, it is necessary to check 617 conditions and calculate 315 *T*-norm operations.

The reliability score for the rule bases (RB*i*) is given in **Table 1.3**.

The classic Rete II, Treat and Leaps method and the proposed method [10–12] were used to compare the reliability of the assessment.

This table clearly shows that the application of the modified Rete II method is justified for rule bases containing a large number of rules and a relatively small number of linguistic variables. In this case, the modified Rete method allows the reliability of information processing to be almost twice that of a fuzzy expert system, and by 20–25 % compared to the classic Rete method.

The research of the developed method showed that the specified method provides an average of 20 % higher reliability of obtaining an estimate (**Table 1.3**).

The advantages of the proposed method are the following:

– the possibility of increasing the reliability of the assessment of the object state due to the parallel use of two bio-inspired algorithms, in comparison with works [9, 12, 19];

– taking into account the degree of awareness of the object state, due to the application of correction coefficients for the degree of awareness, in comparison with works [9–15];

– the construction of both object and relational models, which makes it possible to increase the reliability of the assessment of the objects state, in comparison with works [9–15];

– the possibility of combining the results of the work of bio-inspired algorithms, which makes it possible to mutually check the correctness of the work of each of the algorithms in comparison with works [14, 16, 17];

– universality of solving the task of assessing the condition of objects with different degrees due to the hierarchical nature of their description (Steps 1–4, **Table 1.3**), in comparison with works [9, 12–18];

– the possibility of simultaneously searching for a solution in different directions (Steps 1–4, **Table 1.3**);

– the adequacy of the obtained results (Steps 1–4), in comparison with works [9–18].

Table 1.3 The value of reliability estimates

	n	m_{av}	k	t_{av}	s	$\Xi_{Rete\ II}$	Ξ_{Treat}	Ξ_{Leaps}	$\Xi_{mod\ Rete\ II}$
RB1	20	9	12	5	6	**0.7**	**0.68**	**0.77**	**0.89**
RB2	200	9	12	5	6	**0.76**	**0.67**	**0.75**	**0.85**
RB3	400	9	12	5	6	**0.65**	**0.67**	**0.77**	**0.88**
RB4	600	9	12	5	6	**0.66**	**0.69**	**0.8**	**0.87**
RB5	20	9	12	5	6	**0.69**	**0.7**	**0.76**	**0.87**
RB6	200	9	12	5	6	**0.68**	**0.71**	**0.72**	**0.88**
RB7	400	9	12	5	6	**0.69**	**0.67**	**0.74**	**0.89**
RB8	600	9	12	5	6	**0.7**	**0.66**	**0.77**	**0.95**
RB9	20	9	12	5	6	**0.66**	**0.7**	**0.75**	**0.92**
RB10	200	9	12	5	6	**0.67**	**0.72**	**0.78**	**0.93**
RB11	400	9	12	5	6	**0.64**	**0.71**	**0.73**	**0.97**
RB12	600	9	12	5	6	**0.67**	**0.73**	**0.76**	**0.98**
RB13	20	9	12	5	6	**0.6**	**0.69**	**0.74**	**0.92**
RB14	200	9	12	5	6	**0.74**	**0.73**	**0.75**	**0.94**
RB15	400	9	12	5	6	**0.69**	**0.7**	**0.76**	**0.96**
RB16	600	9	12	5	6	**0.7**	**0.69**	**0.78**	**0.97**

The disadvantages of the proposed approach method should include: increased computational complexity due to the construction of two types of models – object and relational, as well as the operation of two bio-inspired algorithms.

This method will allow:

– to carry out an assessment of the objects state with a given degree of reliability;

– to determine effective measures to increase the reliability of the assessment of the objects state while maintaining the specified efficiency.

The limitations of the research are the need to have an initial database on the objects state, the need to take into account the delay time for collection and proving information from intelligence sources.

The proposed method should be used to solve the problems of assessing the state of multidimensional objects in conditions of uncertainty and risks, which are characterized by high requirements for the reliability of the obtained data.

This research is a further development of researches aimed at developing method principles for increasing the efficiency of processing various types of data, which were published earlier [2, 4–6, 20].

The directions of further research should be aimed at reducing computing costs while processing various types of data in special purpose systems.

CONCLUSIONS

1. The procedures for implementing a method approach to assessing the state of complex hierarchical systems have been defined, thanks to additional and improved procedures that allow:

– to take into account the type of uncertainty and noise;

– to implement adaptive strategies for finding sources of ALA hunting;

– to determine the hunting strategy, taking into account the available computing resources of the system and the priority of the assessment;

– to take into account the available computing resources of the decision-making support system;

– to change the search area by individual ALA;

– to change the speed of the ALA in the specified search direction;

– to carry out the initial issuance of ALA, taking into account the type of uncertainty;

– to conduct a local and global search taking into account the degree of uncertainty and noisy data;

– to conduct training of knowledge bases, which is carried out by training the synaptic weights of the artificial neural network, the type and parameters of the membership function, as well as the architecture of individual elements and the architecture of the artificial neural network as a whole;

– to perform classification of ant nests according to the priority of assessment;

– to adjust the route of the ALA attack by ranking ant nests according to the level of ant pheromones;

– to avoid the problem of local extremum.

2. An example of the use of the proposed method approach has been provided, using the example of solving the task of determining the composition of an operational group of troops (forces) and elements of its operational construction. The specified example showed an increase in the effectiveness of the assessment of the state of hierarchical systems at the level of 22–25 % due to the use of additional improved procedures.

3. The procedures for the analysis and forecasting of the state of multidimensional objects using a metaheuristic algorithm are defined, which allows:

– to take into account available information on the state of multidimensional objects that determine awareness of their state;

– to implement various strategies for finding sources of nectar for BA;

– to determine the BA nectar extraction strategy taking into account the available computing resources of the system and the priority of the assessment;

– to change the search area by individual BA;

– to change the speed of BA movement in the specified search direction;

– to carry out the initial exposure of BA taking into account the type of uncertainty;

– to conduct a local and global search taking into account the degree of uncertainty;

– to conduct training of knowledge bases, which is carried out by training the synaptic weights of the artificial neural network, the type and parameters of the membership function, as well as the architecture of individual elements and the architecture of the artificial neural network as a whole;

– to classify nectar clusters according to the priority of assessment;

– to adjust the BA nectar collection route due to the ranking of nectar accumulations by the level stimulus;

– to avoid the problem of local extremum.

4. An example of the use of the proposed method was provided, on the example of solving the task of determining the composition of an operational group of troops (forces) and elements of its operational construction. The specified example showed an increase in the effectiveness of the operational efficiency of the state assessment of multidimensional objects at the level of 14–16 % due to the use of additional improved procedures.

5. The procedures of the method of increasing the reliability of the assessment of the object state have been determined, which allows:

– to increase the reliability of the assessment of the object state due to the parallel use of two bio-inspired algorithms;

– to take into account the degree of awareness of the object state due to the application of correction coefficients for the degree of awareness;

– to build both object and relational models, which allows to increase the reliability of the assessment of the objects state;

– to combine the results of the work of bio-inspired algorithms, which makes it possible to mutually verify the correctness of the work of each of the algorithms;

– to obtain the universal solution to the task of assessing the objects state with different degrees due to the hierarchical nature of their description;

– to carry out a simultaneous search for a solution in different directions.

6. An example of the use of the proposed method was provided on the example of solving the task of determining the composition of an operational group of troops (forces) and elements of its

operational construction. The specified example showed an increase in the reliability of the state assessment of objects on average by 20 % due to the use of additional improved procedures.

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 Computation and Natural Computation, 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, 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. 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
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. Waterbird, 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