UDC 004.81
DOI: 10.15587/1729-4061.2024.304131

DEVELOPMENT OF A METHOD FOR ASSESSING THE STATE OF DYNAMIC OBJECTS USING A COMBINED SWARM ALGORITHM

Andrii Shyshatskyi
Corresponding author
PhD, Senior Researcher, Associate Professor*
E-mail: ierikon13@gmail.com
Oksana Dmytriieva
Doctor of Economic Sciences, Professor, Head of Department**
Oleksandr Lytvynenko
PhD, Senior Researcher
Research Center
Military Institute of Taras Shevchenko National University of Kyiv
Yuliyi Zdanovskoi str., 81, Kyiv, Ukraine, 03680
Ihor Borysov
PhD, Associate Professor
Deputy Head of the Institute for Scientific Work
Scientific-Research Institute of Military Intelligence
Yuriy Illenka str., 81, Kyiv, Ukraine, 04050
Yuliia Vakulenko
PhD, Associate Professor
Department of Information Systems and Technologies
Poltava State Agrarian University
Skovorody str., 1/3, Poltava, Ukraine, 36003
Temerbay Mukashev
PhD, Professor
Department of Economics and International Business
Karaganda Buketov University
University str., 28, Karaganda, Republic of Kazakhstan, 100028
Oleksandr Mordovtsev
PhD, Associate Professor**
Svitlana Kashkevich
Senior Lecturer*
Anna Lyashenko
Senior Researcher
Scientific Center***
Vira Velychko
Lecturer
Department of Automated Control Systems***
*Department of Computerized Management Systems
National Aviation University
Lubomyra Huzara ave., 1, Kyiv, Ukraine, 03058
**Department of Economics and Entrepreneurship
Kharkiv National Automobile and Highway University
Yaroslava Mudroho str., 25, Kharkiv, Ukraine, 61002
***Military Institute of Telecommunications
and Information Technologies named after Heroes of Kruty
Kyivska str., 45/1, Kyiv, Ukraine, 01011

The object of the study is complex dynamic objects. The subject of the study is the decision-making process in the problems of managing complex dynamic objects. A method of assessing the state of dynamic objects using a combined swarm algorithm is proposed. The research is based on a combined swarm algorithm - for finding a solution to the state of dynamic objects with a hierarchical structure. To train the individuals of the combined swarm algorithm (CSA), evolving artificial neural networks are used, and to select the best in the combined swarm algorithm, an improved genetic algorithm is used. The originality of the method is:

– in taking into account the type of uncertainty and noise of data during the operation of the combined swarm algorithm due to the use of appropriate correction factors;

– in the implementation of adaptive strategies for the search for food sources due to setting appropriate search priorities;

– in taking into account the presence of a predator while choosing food sources by the flock agents of the combined swarm algorithm, which allows excluding unwanted search areas;

– in the additional consideration of the available computing resources of the state analysis system of complex dynamic objects while determining the maximum permissible parameters of the combined swarm algorithm;

– in the possibility of changing the search area and speed of movement by separate individuals of the flock of the combined swarm algorithm;

– in determining the best individuals of the flock of the combined swarm algorithm using an improved genetic algorithm;

– in training knowledge bases, carried out by training the synaptic weights of the artificial neural network, the type and parameters of the membership function, the architecture of individual elements and the architecture of the artificial neural network as a whole.

The method makes it possible to increase the efficiency of data processing at the level of 14–20 % by using additional improved procedures. The proposed method should be used to solve problems of evaluating complex dynamic objects

Keywords: efficiency of decision-making, hierarchical structures, complex and dynamic objects, optimization

Received date 11.03.2024
Accepted date 13.05.2024
Published date 28.06.2024

How to Cite: *Shyshatskyi, A., Dmytriieva, O., Lytvynenko, O., Borysov, I., Vakulenko, Y., Mukashev, T., Mordovtsev, O., Kashkevich, S., Lyashenko, A., Velychko, V. (2024). Development of a method for assessing the state of dynamic objects using a combined swarm algorithm. Eastern-European Journal of Enterprise Technologies, 3 (4 (129)), 44–54. https://doi.org/10.15587/1729-4061.2024.304131*

1. Introduction

Optimization is a complex process of determining multiple solutions for a variety of functions. Many calculation problems today relate specifically to optimization problems [1–3]. While solving optimization problems, solution variables are defined in such a way that complex dynamic objects work at their best point (mode) according to a certain optimization criterion.

In essence, optimization problems of complex dynamic objects are discontinuous, undifferentiated and multimodal. Thus, classical gradient deterministic algorithms [4–6] are not useful for solving optimization problems of complex dynamic objects.

To overcome the drawbacks of classical optimization algorithms for solving optimization problems of complex dynamic objects, a significant number of stochastic optimization algorithms, known as metaheuristic algorithms, were created [7–9].

One type of stochastic optimization algorithms for complex dynamic objects is swarm intelligence algorithms (swarm algorithms) [10, 11]. Swarm intelligence algorithms are based on swarm movement and simulate interactions between the swarm and its environment to improve knowledge of the environment, such as new food sources. The most famous swarm algorithms are the particle swarm optimization algorithm, the artificial bee colony algorithm, the ant colony optimization algorithm, the wolf pack algorithm and the sparrow search algorithm [12–14].

Unfortunately, most of the basic metaheuristic algorithms mentioned above are unable to balance exploration and exploitation, resulting in poor performance for real-world complex optimization problems [15–17].

This encourages the implementation of various strategies to improve the convergence rate and accuracy of basic metaheuristic algorithms. One option to increase the efficiency of decision-making using metaheuristic algorithms is to combine them, that is, to add the basic procedures of one algorithm to another [16–18].

Considering the above, the actual direction of scientific research is the development of new (improvement of existing) metaheuristic algorithms in order to increase the efficiency of analyzing complex dynamic objects with a given reliability.

2. Literature review and problem statement

The work [9] presents a cognitive modeling algorithm. The main advantages of cognitive tools are determined. The shortcomings of this approach include the lack of consideration of the type of uncertainty about the state of the analysis object.

The work [10] disclosed the essence of cognitive modeling and scenario planning. A system of complementary principles of building and implementing scenarios is proposed, different approaches to building scenarios are highlighted, the procedure for modeling scenarios based on fuzzy cognitive maps is described. The approach proposed by the authors does not take into account the type of uncertainty about the state of the analysis object and the noise of the initial data.

The work [11] carried out an analysis of the main approaches to cognitive modeling. Cognitive analysis allows you to: investigate problems with fuzzy factors and relationships; take into account changes in the external environment and use objectively formed trends in the development of the situation to your advantage. At the same time, the issue of describing complex and dynamic processes remains unexplored in this work.

The work [12] presents a method of analyzing large data sets. The specified method is focused on finding hidden information in large data sets. The method includes the operations of generating analytical baselines, reducing variables, detecting sparse features and specifying rules. The disadvantages of this method include the impossibility of taking into account different decision evaluation strategies, the lack of consideration of the type of uncertainty of the input data.

The work [13] presents a mechanism of transformation of information models of construction objects to their equivalent structural models. This mechanism is designed to automate the necessary conversion, modification and addition operations during such information exchange. The disadvantages of the mentioned approach include the impossibility of assessing the adequacy and reliability of the information transformation process and making appropriate correction of the obtained models.

The work [14] developed an analytical web platform to study the geographical and temporal distribution of incidents. The web platform contains several information panels with statistically significant results by territory. The disadvantages of the specified analytical platform include the impossibility of assessing the adequacy and reliability of the information transformation process and high computational complexity. Also, one of the shortcomings of the mentioned research is that the search for a solution is not unidirectional.

The work [15] developed a method of fuzzy hierarchical assessment of library service quality. The specified method allows you to evaluate the quality of libraries based on a set of input parameters. The disadvantages of the specified method include the impossibility of assessing the adequacy and reliability of the assessment and, accordingly, determining the assessment error.

The work [16] carried out an analysis of 30 algorithms for processing large data sets. Their advantages and disadvantages are shown. It was found that the analysis of large data sets should be carried out in layers, take place in real time and have the opportunity for self-learning. The disadvantages of these methods include their high computational complexity and the impossibility of checking the adequacy of the obtained estimates.

The work [17] presents an approach for evaluating input data for decision support systems. The essence of the proposed approach consists in clustering the basic set of input data, analyzing them, after which the system is trained based on the analysis. The disadvantages of this approach are the gradual accumulation of assessment and training errors due to the lack of an opportunity to assess the adequacy of decisions made.

The work [18] presents an approach to processing data from various sources of information. This approach allows you to process data from various sources. The disadvantages of this approach include the low accuracy of the obtained estimate and the impossibility of verifying the reliability of the obtained estimate.

The work [19] carried out a comparative analysis of existing decision support technologies, namely: the analytic hierarchy process, neural networks, fuzzy set theory, genetic algorithms and neuro-fuzzy modeling. The advantages and disadvantages of these approaches are indicated. The scope of their application is defined. It is shown that the analytic hierarchy process works well under the condition of complete initial information, but due to the need for experts to compare alternatives and choose evaluation criteria, it has a high share of subjectivity. For forecasting problems under risk and uncertainty, the use of fuzzy set theory and neural networks is justified.

The work [20] developed a method of structural and objective analysis of the development of weakly structured systems. An approach to the study of conflict situations caused by contradictions in the interests of subjects that affect the development of the studied system and methods of solving poorly structured problems based on the formation of scenarios for the development of the situation. At the same time, the problem is defined as the non-compliance of the existing system state with the required one, which is set by the management subject. The disadvantages of the proposed method include the problem of the local optimum and the inability to conduct a parallel search.

The work [21] presents a cognitive approach to simulation modeling of complex systems. The advantages of the specified approach, which allows you to describe the hierarchical components of the system, are shown. The shortcomings of the proposed approach include the lack of consideration of the computing resources of the system.

The work [22] indicated that the most popular evolutionary bio-inspired algorithms are the so-called «swarm» procedures (Particle Swarm Optimization – PSO). Among them, there are cat swarm optimization algorithms (CSO), which are very promising both in terms of speed and ease of implementation. These algorithms have proven their effectiveness in solving a number of rather complex problems and have already undergone a number of modifications. Among the modifications, procedures based on harmonic search, fractional derivatives, adaptation of search parameters and, finally, «crazy cats» can be noted. At the same time, these procedures are not without some drawbacks that worsen the properties of the global extremum search process.

An analysis of the works [9–22] showed that the common shortcomings of the above studies are:

– the lack of possibility of forming a hierarchical system of indicators for assessing the state of complex dynamic objects;

– the lack of consideration of computing resources of the system that evaluates the state of complex dynamic objects;

– the lack of mechanisms for adjusting the system of indicators for assessing the state of complex dynamic objects;

– the lack of consideration of the type of uncertainty and noise of data on the state of complex dynamic objects, which creates corresponding errors while assessing their real state;

– the lack of deep learning mechanisms for knowledge bases;

– high computational complexity;

– the lack of consideration of computing (hardware) resources available in the system;

– the lack of search priority in a certain direction.

3. The aim and objectives of the study

The aim of the study is to develop a method for assessing the state of dynamic objects using a combined swarm algorithm. This will increase the efficiency of assessing the state of dynamic objects with a given reliability and developing subsequent management decisions. This will make it possible to develop software for intelligent decision support systems.

To achieve the aim, the following objectives were set:

– to determine the algorithm for implementing the method;

– to verify the method while analyzing the operational situation of a group of troops (forces).

4. Materials and methods

The problem solved in the study is to increase the efficiency of decision-making in the problems of assessing the state of dynamic objects while ensuring the given reliability regardless of its hierarchy. The object of the study is complex dynamic objects with a hierarchical structure. The subject of the study is the process of decision-making in management problems using an improved hawk optimization algorithm, an improved coot optimization algorithm, an improved genetic algorithm and evolving artificial neural networks.

The advantage of metaheuristic optimization algorithms allows for gradient search, with constraint elements in certain directions and the necessary search speed, which is an undeniable advantage in the analysis of hierarchical and non-uniform dynamic objects [22–27].

During the work of metaheuristic algorithms, there is a need to select the best individuals that meet the optimal criteria in this study – an improved genetic algorithm was chosen to solve this partial problem in the study. This improved algorithm has all necessary procedures for this, such as selection, mutation, etc.

In order to reduce the number of errors that accumulate during the work of metaheuristic algorithms, it is necessary to adjust the knowledge bases (in the natural language of optimization, this is the accumulation of experience). For this, the method of training artificial evolving neural networks is used. This approach is due to the fact that it allows you to create the necessary number of connections for a specific task.

The hypothesis of the study is the possibility of increasing the efficiency of decision-making with a given assessment reliability using a combined swarm algorithm.

The proposed method was simulated in the MathCad 14 software environment (USA). The problem solved during the simulation was to assess the elements of the operational situation of a group of troops (forces). The hardware of the research process is AMD Ryzen 5.

An operational group of troops (forces) was considered as an object of assessment. The operational group of troops (forces) formed on the basis of an operational command with a typical composition of forces and means according to the wartime state and with a range of responsibilities under current regulations.

The research is based on a combined swarm algorithm – for finding a solution regarding the state of dynamic objects with a hierarchical structure. To train the individuals of the combined swarm algorithm, evolving artificial neural networks are used, and to select the best ones in the combined swarm algorithm, an improved genetic algorithm is used.

5. Development of a method for assessing the state of dynamic objects using a combined swarm algorithm

5. 1. Algorithm of the method for assessing the state of dynamic objects using the combined swarm algorithm

The proposed approach is a population-based swarm algorithm, which assumes that each of the animals (coots and hawks) forms a flock. The proposed approach is able to provide appropriate solutions for optimization problems in a multiple iterative process based on the possibilities of finding its members (agents of the combined flock) in the problem-solving space.

Each member of the combined flock, based on its position in space, determines the values for the solution variables of the problem. Thus, each agent of the combined flock, as a member of the population, is a candidate for solving the problem, which is modeled mathematically using a vector.

The method of assessing the state of dynamic objects using the combined swarm algorithm consists of the following sequence of actions:

Step 1. Input of initial data. At this stage, the available initial data on the dynamic object to be evaluated are entered.

Step 2. Setting the flock agents of the combined algorithm on the search plane.

The agents of the combined flock together form the population of the combined swarm algorithm, which can be modeled mathematically using a matrix according to the equation (at a separate point in time). Agents of the combined swarm algorithm are set on the search plane, taking into account the uncertainty about the dynamic object with a hierarchical structure to be evaluated, and the basic model of its state is initialized [2, 19, 21] (1).

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

The position of the flock agents of the combined swarm algorithm in the problem space is initialized at the beginning of the combined swarm algorithm run using equation (2):

$$x_{i,d}=lb_d+r\cdot\left(ub_d-lb_d\right). \quad (2)$$

Here, X is the population matrix of the flock agents of the combined swarm algorithm, X_i is the i-th flock member of the combined swarm algorithm, $x_{i,d}$ is the d-th dimension in the solution search space, N is the number of flock agents of the combined swarm algorithm, m is the number of solution variables, r is a random number in the interval [0, 1], lb_d, ub_d are the lower and upper bounds of the d-th solution variables.

Since the position of each flock agent of the combined swarm algorithm in the problem-solving space represents a solution to the problem, the value of the objective function can be estimated according to the position of each agent. Accordingly, the set of estimated values for the objective function can be represented by equation (3):

$$F=\begin{bmatrix} F_1 \\ . \\ . \\ . \\ F_i \\ . \\ . \\ . \\ F_N \end{bmatrix}_{N\times 1} = \begin{bmatrix} F(X_1) \\ . \\ . \\ . \\ F(X_i) \\ . \\ . \\ . \\ F(X_N) \end{bmatrix}_{N\times 1}, \quad (3)$$

where F is the vector of the estimated objective function for solving the problem of assessing the state of dynamic objects, F_i is the estimated objective function based on the i-th flock member of the combined swarm algorithm.

The best value obtained for the objective function corresponds to the best flock member (the best possible solution) and the worst value obtained for the objective function corresponds to the worst flock member (the worst possible solution). Since the position of the flock agents of the combined swarm algorithm in the problem-solving space is updated at each iteration, the best flock member must also be updated based on a comparison of the updated values for the objective function. At the end of the algorithm implementation, the position of the best flock agent of the combined algorithm, obtained during the iterations of the algorithm, is presented as a solution to the problem of finding a solution to the state of the dynamic object.

Step 3. Numbering the flock agents of the combined swarm algorithm in the flock, i, $i\in[0, S]$. At this stage, each flock agent of the combined swarm algorithm is assigned a serial number.

Step 4. Determining the initial velocity of the flock agents of the combined swarm algorithm.

The initial velocity v_0 of each flock agent of the combined swarm algorithm is determined by the following expression:

$$v_i=\left(v_1,v_2...v_S\right),\ v_i=v_0. \quad (4)$$

In the planning of the proposed approach, the position of the flock members of the combined swarm algorithm in the problem-solving space is updated based on the simulation of the hunting (foraging) strategy of hawk agents and coot agents in the wild. Accordingly, in each iteration, the position of the population members of the combined swarm algorithm is updated in two stages:

– exploration based on modeling the movement of agents of the combined swarm algorithm to the food source;

– exploitation based on simulating the behavior of the flock agents of the combined swarm algorithm that feed.

Step 5. Preliminary evaluation of the search area of the flock agents of the combined swarm algorithm. In this procedure, the natural language search area is determined precisely by the halo of the existence of the flock agents of the combined swarm algorithm. Given that food sources of flock agents of the combined swarm algorithm are diverse: for hawk agents, it is food of animal origin, and for coot agents, it is food of plant origin. Therefore, it is advisable to sort the fitness of food sources.

Step 6. Classification of food sources for flock agents of the combined swarm algorithm.

Locations of the best food source (minimum fitness) are (FS_{ht}):

– for hawk agents – carrion (carcasses of dead animals), which require the least amount of energy to find and obtain them;

– for coot agents – plant food (watercress), which is nearby and requires the least amount of energy to find and obtain it.

Delicate food of animal origin (for hawk agents) and other plant food (for coot agents) are denoted as FS_{at}.

Other non-priority food sources (food that is necessary for the survival of individuals) are designated as FS_{nt}.

$$FS_{ht}=FS(\text{sorte\_index}(1)), \quad (5)$$

$$FS_{at}(1{:}3)=FS(\text{sorte\_index}(2{:}4)), \quad (6)$$

$$FS_{nt}(1{:}NP\text{-}4)=FS(\text{sorte\_index}(5{:}NP)). \quad (7)$$

Procedures 1–6 are common to all flock agents of the combined swarm algorithm. Below, we consider the specific procedures for each of the flock components of the combined swarm algorithm.

Step 7. Selection of individuals for each of the flocks of the combined swarm algorithm. At this stage, the best individuals are selected using the improved genetic algorithm proposed by the authors in [21].

Step 8. Complex search procedure by each swarm algorithm.

Step 8. 1. Assessing the state of a dynamic object based on the hawk optimization algorithm.

The hawk optimization algorithm is characterized by three main components: the exploration phase, the exploitation phase and the transition phase from exploration to exploitation:

8. 1. 1. Intelligence phase of the hawk agent swarm algorithm.

During this phase, hawk agents sit in certain places according to expression (1). They detect prey with their acute vision and then choose between two strategies with equal probability to carry out hunting. The formula for updating the position of hawk agents at this stage is as follows:

$$X(t+1)=\begin{cases} X_{rand}(t)-r_1\cdot\left|X_{rand}(t)-2r_2X(t)\right|,\ q\geq 0.5,\\ \left(X_{prey}(t)-X_m(t)\right)-\\ -r_3\left(LB+r_4\left(UB-LB\right)\right),\ q<0.5, \end{cases} \tag{8}$$

$X(t)$, $X(t+1)$ is the position vector of hawk agents in the current and next iterations; t is the current number of iterations; X_{prey} is the position of the prey, which is considered the optimal solution; $X_{rand}(t)$ is the position vector of a random individual in the current population; r_1, r_2, r_3, r_4, q are random numbers between [0, 1]; LB, UB are the lower and upper bounds of variables; $X_m(t)$ is the average position of all hawk agents in the population, which is calculated as follows:

$$X_m(t)=\frac{1}{N}\sum_{i=1}^{N}X_i(t), \tag{9}$$

N is the population size. $X_i(t)$ is the vector of the current position of the i-th hawk.

8. 1. 2. Transition of hawk agents from intelligence to exploration.

In the hawk agent search procedure, the energy of the prey distance E from the hawk agent causes the algorithm to switch between the phases of global exploration and local exploitation. The energy of the prey gradually decreases during the escape process, which can be simulated as in equation (10):

$$E=2E_0\left(\frac{T-t}{T}\right), \tag{10}$$

E_0 is a random number between [–1, 1] representing the initial state of the prey's slipping energy; T is the maximum number of iterations. When $|E|\leq 1$, the hawk agent will continue to search for prey locations in the target area. In the case of $|E|<1$, the hawk will start hunting the prey found in the previous stage and move to the exploitation stage.

8. 1. 3. Exploration phase of hawk agents.

During the exploitation phase, there are four possible strategies, including soft siege, hard siege, soft siege with gradual rapid dives and hard siege with gradual rapid dives. The strategies mimic the process of a hawk attacking its prey. r represents the probability of whether the victim can escape the danger before a hawk attack, which is a random number between [0,1]. $r<0.5$ means that the prey successfully passed through a dangerous situation, $r\geq 0.5$ means a case of unsuccessful escape. Different combinations of r-value and escape energy E correspond to different predation strategies. When $|E|<0.5$, a hard siege is performed. Otherwise, a soft siege is conducted.

8. 1. 3. 1. Soft siege of hawk agents.

Soft siege of hawk agents is performed when $r\geq 0.5$ and $|E|<0.5$. At this stage, the positions of the hawk agents are updated as follows (expressions (11)–(13)):

$$X(t+1)=\Delta X(t)-E\left|JX_{prey}(t)-X(t)\right|, \tag{11}$$

$$\Delta X(t)=X_{prey}(t)-X(t), \tag{12}$$

$$J=2(1-r_5), \tag{13}$$

$\Delta X(t)$ is the distance between the hawk agent position and the victim; r_5 is a random number between [0, 1]; J is the random intensity of the prey jump.

8. 1. 3. 2. Hard siege of hawk agents.

The hawk agent will take a hard siege when $r\geq 0.5$ and $|E|<0.5$. The mathematical expression describing this behavior of hawk agents is given below:

$$X(t+1)=X_{prey}(t)-E\,|\,\Delta X(t)\,|. \tag{14}$$

8. 1. 3. 3. Soft siege of hawk agents with gradual rapid dives.

When $r<0.5$ and $|E|\geq 0.5$, the hawk agents will take a soft siege with gradual rapid dives. Levy flight is integrated into the procedure of the hawk agent flock algorithm, and the mathematical model is described as follows (expressions (15)–(17)):

$$Y=X_{prey}(t)-E\left|JX_{prey}(t)-X(t)\right|, \tag{15}$$

$$Z=Y+S\times LF(D), \tag{16}$$

$$X(t+1)\begin{cases} Y,\ \text{if}\ \ F(Y)<F\left(X(t)\right),\\ Z,\ \text{if}\ \ F(Z)<F\left(X(t)\right), \end{cases} \tag{17}$$

D is the dimension of the problem to be solved; S is a random vector of size $1\times D$; $F(\cdot)$ is the objective function.

Only the best position between Y and Z is chosen by the following position, $LF(\cdot)$ is the Levy flight function, which is calculated as follows:

$$LF(x)=0.01\times\frac{u\times\sigma}{|v|^{\frac{1}{\beta}}},\ \sigma=\left(\frac{\Gamma(1+\beta)\times\sin\left(\frac{\pi\beta}{2}\right)}{\Gamma(1+\beta)\times\beta\times 2^{\left(\frac{\beta-1}{2}\right)}}\right)^{1/\beta}, \tag{18}$$

u and v are two random numbers between [0, 1], β is a constant with a fixed value of 1.5. $\Gamma(\cdot)$ is a gamma function.

8. 1. 3. 4. Hard siege with gradual rapid dives.

When $r<0.5$ and $|E|<0.5$, Harris's hawk will perform a hard siege to approach the prey and then make a surprise attack. The mathematical model of this behavior is modeled as follows (relation (19)–(21)):

$$Y=X_{prey}(t)-E\left|JX_{prey}(t)-X_m(t)\right|, \tag{19}$$

$$Z=Y+S\times LF(D), \tag{20}$$

$$X(t+1)\begin{cases}Y, \text{ if } F(Y)<F(X(t)),\\ Z, \text{ if } F(Z)<F(X(t)),\end{cases} \tag{21}$$

$X_m(t)$ is calculated using equation (9). Moreover, only the best position between Y and Z is selected as the next position.

Step 8. 2. Assessing the state of a dynamic object based on the coot bird algorithm.

In the coot bird agent algorithm, four different irregular and regular behavior strategies are implemented: random movement, chain movement, position adjustment based on group leaders and leader movement.

According to the coot's habits, the original population is divided into two parts: leader coot and follower coot. If N is the population size, the number of coot leaders is calculated as a percentage of the total population equal to L, and the other members (N–L) are considered coot followers. It is noted that all leaders are randomly selected from the population. Then the mentioned four movements are performed.

8. 2. 1. Random movement of coot agents.

At this stage, the random position of the coot agent Q is determined by equation (22). Coot agent followers move to this random position to explore different parts of the search domain:

$$Q=\text{rand}(1,D)*(UB\ -\ LB)+LB, \tag{22}$$

where D is the dimension of the problem, LB, UB are the lower and upper bounds of the variables. The random motion gives the algorithm better global search efficiency and enhances the algorithm's ability to exit the local optimum. The new coot position is updated as follows:

$$X_i(t+1)=X_i(t)+A\times r_6\times(Q-X_i(t)), \tag{23}$$

$X_i(t+1)$ is the position of the i-th coot agent follower in the next iteration t, r_6 is a random number in the range [0, 1] and the parameter A is calculated according to equation (24):

$$A=1-\frac{t}{T}, \tag{24}$$

where t is the number of current iterations and T is the maximum number of iterations.

8. 2. 2. Chain movement of coot agents.

The average position of two individuals is used to perform chain movements by coot agents. The new position of the coot follower is calculated as follows:

$$Xi(t+1)=\frac{1}{2}\times(X_{i-1}(t)+X_i(t)), \tag{25}$$

where $X_{i-1}(t)$ is the position of the (i–1)-th follower in the current iteration t.

8. 2. 3. Setting the positions of coot agents based on the position of group leaders.

As a rule, the whole group is led by one of the group leaders in front, and all remaining coots must change their position based on the leaders and move towards them. However, a serious problem to be solved is that each coot must update its position according to the leader. Equation (19) is designed to select the leader as follows:

$$k=1+(i\ MOD\ L), \tag{26}$$

where i is the index of the current follower, L is the number of leaders and k is the index number of the coot agent leader.

The next position of the coot agent follower based on the selected leader k is calculated using equation (26):

$$\begin{aligned}&X_i(t+1)=LeaderX_k(t)+2\times r_7\times\\&\times\cos(2R\pi)\times(LeaderX_k(t)-X_i(t)),\end{aligned} \tag{27}$$

where $LeaderX_k(t)$ is the position of the chosen leader of the flock of coot agents, r_7 is a random number in the interval [0, 1], and R is a random number in the interval [–1, 1].

8. 2. 4. Leadership movement of coot agents.

The group must be focused on the optimal territory, so in some cases leaders have to leave the current optimal position in search of a better one. The formula for updating the leader's position is as follows:

$$LeaderX_i(t+1)=\begin{cases}B\times r_8\times\cos(2R\pi)\times\\ \times(gBest(t)-LeaderX_i(t))+\\ +gBest(t), r_9<0.5,\\ B\times r_8\times\cos(2R\pi)\times\\ \times(gBest(t)-LeaderX_i(t))-\\ -gBest(t), r_9\geq 0.5.\end{cases} \tag{28}$$

In equation (28), $gBest$ means the current optimal position; r_8 and r_9 are random numbers in the interval [0, 1], R is a random number in the interval [1, 1]; r_8 is designed to generate stochastic movement to help the algorithm eliminate local optimal solutions; $\cos(2R\pi)$ is designed to find the best individual with different radii to get the top position. The value of B is calculated using equation (29):

$$B=2-t\times\left(\frac{1}{T}\right), \tag{29}$$

where t is the number of current iterations and T means the maximum.

Step 9. Combining individual swarm procedures into a mixed one.

The combination of individual swarm procedures into a mixed one is carried out using the procedure of ensemble mutation of separate individuals, described by the following mathematical dependencies (30)–(32):

$$V_{i1}=\begin{cases}X_{R1}+F_1\times(X_{R2}-X_{R3}), r_{10}<C_1,\\ X_i, r_{10}\geq C_1,\end{cases} \tag{30}$$

$$V_{i2}=\begin{cases}X_{R4}+F_2\times(X_{R5}-X_{R6})+\\ +F_2\times(X_{R7}-X_{R8}), r_{11}<C_2,\\ X_i, r_{11}\geq C_2,\end{cases} \tag{31}$$

$$V_{i3}=\begin{cases}X_i+F_3\times(X_{R9}-X_i)+\\ +F_3\times(X_{R10}-X_{R11}), r_{12}<C_3,\\ X_i, r_{12}\geq C_3,\end{cases} \tag{32}$$

where V_{i1}, V_{i2} and V_{i3} are the generated mutant positions of the i-th flock agent of the combined swarm algorithm, $R_1\sim R_{11}$ are integer indicators in the range [1, N], F_1, F_2 and F_3 are scaling factors; $r_{10}\sim r_{12}$ are random numbers in the range [0, 1], C_1, C_2 and C_3 are crossover rates of the agents of the combined swarm algorithm.

After generating candidate mutant positions V_{i1}, V_{i2} and V_{i3}, the best position V_i with the lowest fitness value is selected to compare with the fitness of the original position X_i and then the best position will be saved as a new X_i to participate in the next iteration calculation. These processes can be described using equation (33):

$$X_i = \begin{cases} V_i, \text{if } F(V_i) < F(X_i), \\ X_i, \text{otherwise}, \end{cases} \quad (33)$$

where $F(\cdot)$ is the cost function.

Step 10. Checking the presence of a predator by agents of the combined swarm algorithm. At this stage, agents from the flock of the combined swarm algorithm check for the presence of predators. If there are predators, go to Step 11. If there are no predators, go to Step 12.

Step 11. Escape and struggle with predators by the flock agents of the combined swarm algorithm. The escape and anti-predator strategy causes the agents of the combined swarm algorithm to change their position near the position they are at. Simulating this natural behavior of the combined swarm algorithm agents improves the effectiveness of using combined swarm algorithm agents in local search in the problem-solving space around potential solutions. In the initial iterations of the algorithm, priority is given to a global search to identify the optimal region in the search space, the radius of this environment is considered variable. First, the highest value is set and then it becomes smaller during the iterations of the algorithm. For this reason, local lower/upper bounds were used in this stage of the combined swarm algorithm agents to create a variable radius with the iteration of the algorithm. To simulate this phenomenon in the combined swarm algorithm, a plot is assumed around each flock agent of the combined algorithm, which first randomly generates a new position in this neighborhood using (34), (35). Then, if the value of the objective function improves, this new position replaces the previous position according to (36):

$$x_{i,j}^{P_3} = x_{i,j} + \left(lb_{local,j}^t + \left(ub_{local,j}^t - rand \cdot lb_{local,j}^t\right)\right), \quad (34)$$

$$Local\ bounds: \begin{cases} lb_{local,j}^t = \dfrac{lb_j}{t}, \\ ub_{local,j}^t = \dfrac{ub_j}{t}, \end{cases} \quad (35)$$

$$X_i = \begin{cases} X_i^{P_3}, F_i^{P_3} \langle F_i; \\ X_i, \text{else}, \end{cases} \quad (36)$$

where $X_i^{P_3}$ is the new generated position of the i-th flock agent of the combined swarm algorithm, $x_{i,j}^{P_3}$ is the j-th size of the flock agent of the combined swarm algorithm, $F_i^{P_3}$ is the value of the objective function, t is the iterative circuit, lb_j and ub_j are the lower and upper bounds of the j-th variable. $lb_{local,j}^t$ and $ub_{local,j}^t$ are local lower and local upper bounds, admissible for the j-th variable, respectively, for simulating local search in the neighborhood of candidate solutions.

Step 12. Checking the stop criterion. The algorithm terminates when the maximum number of iterations is completed. Otherwise, the behavior of generating new locations and checking conditions is repeated.

Step 13. Training the knowledge bases of the flock agents of the combined swarm algorithm.

In the mentioned research, the learning method based on evolving artificial neural networks developed in [2, 28–30] is used to train the knowledge bases of each flock agent of the combined swarm algorithm. The method is used to change the movement nature of each flock agent of the combined swarm algorithm, for more accurate analysis results in the future.

Step 14. Determining the amount of necessary computing resources of the intelligent decision support system.

In order to prevent calculations from looping on Steps 1–13 of this method and increase the efficiency of calculations, the system load is additionally determined. When the specified threshold of computational complexity is exceeded, the amount of software and hardware resources that must be additionally involved is determined using the method proposed in [31].

End of the algorithm.

5. 2. Example of applying the proposed method in the analysis of the operational group of troops (forces)

To determine the effectiveness of the proposed method, a simulation of its work was carried out to solve the problem of determining the composition of the operational group of troops (forces) and elements of its operational structure.

Initial data for determining the composition of the operational group of troops (forces) and elements of its operational structure using the method:

– the number of sources of information about the state of the monitoring object – 3 (radio monitoring means, earth remote sensing tools and unmanned aerial vehicles). To simplify the modeling, the same number of each tool was taken – 4 tools each;

– the number of informational features for determining the state of the monitoring object – 12. Such parameters include: affiliation, type of organizational and staff formation, priority, minimum width along the front, maximum width along the front. The number of personnel, minimum depth along the flank, maximum depth along the flank, the number of samples of weapons and military equipment (WME), the number of types of WME samples and the number of communication devices), type of operational structure are also taken into account;

– options of organizational and staff formations – company, battalion, brigade.

Operating parameters of swarm algorithms:

– the number of iterations – 100;

– the number of individuals in the flock – 50 (for the improved method 25/25);

– the range of the feature space – [–150, 150].

Parameters of the advanced genetic algorithm:

– selection – roulette wheel (proportional);

– crossover – probability=0.8;

– mutation – Gaussian probability=0.05.

The efficiency of the method of assessing the state of dynamic objects using the combined swarm algorithm is compared with the swarm optimization algorithms listed in Tables 1–3. The comparison was made with unimodal and multimodal functions.

As can be seen from Tables 1–3, the increase in decision-making efficiency is achieved at the level of 14–20 % by using additional procedures.

It can be seen that the method of assessing the state of dynamic objects using the combined swarm algorithm is able to converge to the true value for most unimodal functions with the fastest convergence rate and the highest accuracy, while the convergence results of the particle swarm algorithm are far from satisfactory.

Table 1

Efficiency of optimization algorithms in solving the problem of determining the composition of an operational group of troops (forces) and elements of its operational structure

Name of the algorithm	T_s	Optimal variables		L	Optimal value
		T_h	R		
Walrus optimization algorithm	0.7280271	0.3845792	40.312284	200	5,882.8955
Particle swarm optimization algorithm	0.7480269	0.3845797	40.312282	200	5,882.9013
Flying squirrel search optimization algorithm	0.7690308	0.384581	40.312476	199.99732	5,882.9077
Artificial bee colony algorithm	1.1950157	0.64038	60.549321	48.031984	7,759.8234
Ant colony optimization algorithm	0.7780271	0.3845792	40.312284	200	5,882.9013
Proposed method	0.7794994	0.385819	40.386517	200	5,909.3749
Monkey search algorithm	0.911517	0.4510723	46.230782	133.83941	6,270.8621
Bat algorithm	0.8344267	0.4164052	43.217775	163.90679	6,003.8497
Locust swarm optimization algorithm	0.7784599	0.3858127	40.320627	199.96442	5,890.2105
Genetic algorithm	1.5622593	0.4813024	47.695987	124.64823	10,807.366
Cat swarm optimization algorithm	1.1300127	1.1576349	44.110061	190.7876	11,984.417
Invasive weed optimization algorithm	1.55006	0.6231249	63.139483	49.78495	9,998.6395
Firefly algorithm	1.406417	0.7832762	58.253368	73.964478	10,920.286

Table 2

Comparative analysis of the efficiency of optimization algorithms in solving the problem of determining the composition of an operational group of troops (forces) and elements of its operational structure

Name of the algorithm	Average	Best	Worst	Standard	Median	Rank
Walrus optimization algorithm	5,882.8955	5,882.8955	5,882.8955	1.87E-12	5,882.8955	1
Particle swarm optimization algorithm	5,891.226	5,882.9013	5,965.0365	22.218932	5,882.9017	3
Flying squirrel search optimization algorithm	6,219.5386	5,882.9077	7,046.3206	352.35848	6,047.6955	5
Artificial bee colony algorithm	12,409,586	7,759.8234	19,991.769	3127.065	11,403.338	9
Ant colony optimization algorithm	5,882.9013	5,882.9013	5,882.9013	3.68E-06	5,882.9013	2
Proposed method	6,271.132	5,909.3749	6,948.3792	333.1584	6,143.6153	6
Monkey search algorithm	7,998.6372	6,270.8621	12,805.388	1,681.8974	7,579.6333	8
Bat algorithm	6,518.1019	6,003.8497	7,050.4059	320.31898	6,572.19	7
Locust swarm optimization algorithm	6,012.3675	5,890.2105	6,670.9945	239.38549	5,898.5494	4
Genetic algorithm	28,273.334	10,807.366	60,311.64	13,795.65	24,975.491	12
Cat swarm optimization algorithm	20,643.589	11,984.417	32,105.445	6711.6675	19,830.394	10
Invasive weed optimization algorithm	29,687.575	9,998.6395	50,712.307	12,915.318	32,709.339	13
Firefly algorithm	25,427.766	10,920.286	45,530.922	10,828.815	22,551.255	11

Table 3

Comparative analysis of the efficiency of optimization algorithms in determining the composition of the operational group of troops (forces)

No. F	Value	Grey wolf optimization algorithm	Walrus optimization algorithm	Hawk optimization algorithm	Bat algorithm	Coot optimization algorithm	Proposed method
1	2	3	4	5	6	7	8
F_1	Average	2.22E–27	7.92E–72	5.75E–98	6.49E–06	4.99E–07	1.20E–22
	Standard	5.43E–27	4.22E–71	2.87E–97	1.28E–05	6.71E–07	6.59E–22
F_2	Average	8.23E–17	4.46E–51	1.51E–50	3.39E–05	1.26E–05	2.01E–14
	Standard	6.13E–17	1.30E–50	5.19E–50	4.15E–05	2.54E–05	6.47E–14
F_3	Average	1.43E–05	4.25E+04	3.50E–72	9.74E+01	1.29E–01	3.96E–26
	Standard	3.47E–05	1.23E+04	1.79E–71	1.57E+02	2.52E–01	1.92E–26
F_4	Average	1.08E–06	5.26E+01	3.14E–49	2.87E–01	5.03E–02	2.10E–13
	Standard	1.26E–06	2.68E+01	1.47E–48	3.48E–01	8.79E–02	9.37E–13
F_5	Average	2.70E+01	2.80E+01	8.34E–03	2.89E+01	2.84E+01	4.50E+01
	Standard	8.74E–01	5.51E–01	1.36E–02	1.26E–01	4.05E–01	3.20E+01
F_6	Average	7.66E–01	4.69E–01	1.02E–04	3.72E+00	2.56E+00	1.96E–01
	Standard	3.53E–01	2.86E–01	1.16E–04	4.53E–01	4.64E–01	1.22E–01
F_7	Average	2.19E–03	2.54E–03	1.28E–04	2.09E–03	6.91E–03	5.75E–03
	Standard	1.22E–03	2.76E–03	1.01E–04	2.80E–03	5.71E–03	5.22E–03
F_8	Average	–6.02E+03	–1.08E+04	–1.24E+04	–5.74E+03	–5.91E+03	–7.40E+03
	Standard	1.02E+03	1.73E+03	5.03E+02	7.42E+01	5.13E+02	7.23E+02
F_9	Average	1.90E+00	1.75E+00	0.00E+00	1.66E+01	1.12E+01	1.73E–04
	Standard	2.74E+00	7.09E+00	0.00E+00	2.01 E+01	1.64E+01	9.48E–04

Continuation of Table 3

1	2	3	4	5	6	7	8
F_{10}	Average	1.00E–13	4.56E–15	8.88E–16	2.00E+01	2.00E+01	2.76E–09
	Standard	1.47E–14	2.38E–15	0.00E+00	1.20E–03	1.20E–03	1.40E–08
F_{11}	Average	4.27E–03	6.89E–03	0.00E+00	2.57E–02	2.59E–02	1.26E–16
	Standard	1.06E–02	2.77E–02	0.00E+00	6.67 E–02	4.07E–02	2.75E–16
F_{12}	Average	4.27E–02	2.20E–02	1.02E–05	5.41E–01	2.41E–01	3.74E–01
	Standard	2.01E–02	1.08 E–02	1.58E–05	2.24E–01	1.61E–01	8.27E–01
F_{13}	Average	6.82E–01	5.70E–01	9.94E–05	2.72E+00	1.89E+00	4.10E–01
	Standard	2.49E–01	2.52E–01	1.54E–04	1.27E–01	2.51E–01	5.26E–01
F_{14}	Average	4.29E+00	3.48E+00	1.72E+00	1.32E+00	1.23E+00	9.98E–01
	Standard	4.20E+00	3.66E+00	1.97E+00	1.78E+00	5.64E–01	6.60E–16
F_{15}	Average	3.14E–03	8.30E–04	3.74E–04	1.30E–03	1.09E–03	1.32E–03
	Standard	6.88E–03	5.59E–04	1.71E–04	5.13E–05	3.35E–04	3.60E–03
F_{16}	Average	–1.03E+00	–1.03E+00	–1.03E+00	–1.03E+00	–1.03E+00	–1.03E+00
	Standard	2.62E–08	3.12E–09	8.85E–10	1.46E–05	3.45E–06	3.23E–12
F_{17}	Average	3.98E–01	3.98E–01	3.98E–01	5.54E–01	3.98E–01	3.98E–01
	Standard	3.11E–06	3.57E–05	2.30E–05	8.47E–01	4.95E–04	8.84E–07

6. Discussion of the results of developing a method for assessing the state of dynamic objects using a combined swarm algorithm

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

– the initial setting of the flock agents of the combined swarm algorithm is carried out taking into account the type of uncertainty (Step 2), compared to [9, 14, 21], by using the corresponding correction factors, which are known before the method starts working;

– the initial velocity of each flock agent of the combined swarm algorithm is taken into account (Step 4), compared to [9–15], which is set according to the shape of the hierarchical dynamic object, for the purpose of uniform analysis over time;

– the fitness of the search location for the flock agents of the combined swarm algorithm is determined, which reduces the solution search time (Step 5), compared to [14, 16, 17]. This is achieved by using the ranking of priority search areas, using the appropriate attractiveness coefficients of food located in certain search areas;

– taking into account the presence of a predator during foraging by the flock agents of the combined swarm algorithm, which allows avoiding local optima (Steps 9, 10), compared to [12, 13, 15–18]. This advantage is achieved by using appropriate predator presence coefficients (undesirable search area), with appropriate escape procedures;

– perform mutual verification of the adequacy of the work of each flock of the combined swarm algorithm (Step 8), compared to [12, 13, 15–20]. The advantage is achieved by using the ensemble selection procedure, which allows using several types of multi-agent systems;

– accelerated selection of individuals for each flock of the combined swarm algorithm due to the use of an improved genetic algorithm (Step 7), compared to [9, 12–18];

– the universality of solving the problem of analyzing the state of dynamic objects by flock agents of the combined swarm algorithm due to the hierarchical nature of their description (Steps 1–14), compared to [9, 12–18];

– the possibility of simultaneous solution search in different directions (Steps 1–14, Tables 1–3);

– the adequacy of the obtained results (Steps 1–14), compared to [9–23];

– the ability to combine different swarm algorithms, due to the use of the ensemble mutation procedure (Step 9), compared to [9–23];

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

– the possibility of deep learning of knowledge bases of flock agents of the combined algorithm (Step 14), compared to [9–23];

– the possibility of calculating the necessary amount of computing resources, which must be involved in case of impossibility of carrying out calculations with available computing resources (Step 14), compared to [9–23].

The disadvantages of the proposed method include:

– the loss of informativeness while assessing the state of complex dynamic objects due to the construction of the membership function;

– lower accuracy of assessment by a single assessment parameter for the state of complex dynamic objects;

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

– lower assessment accuracy compared to other assessment methods.

This method will allow you:

– to assess the state of complex dynamic objects;

– to determine effective measures to increase the efficiency of managing complex dynamic objects;

– to increase the speed of assessing the state of complex dynamic objects;

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

The limitations of the study are the need to have an initial database on the state of a complex dynamic object, the need to take into account the delay time for collecting and communicating information from intelligence sources.

The proposed approach should be used to solve problems of evaluating complex and dynamic processes characterized by a high degree of complexity.

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

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

7. Conclusions

1. The algorithm for implementing the method is defined, due to additional and improved procedures, which allows you:

– to take into account the type of uncertainty and noise of data during the operation of the combined swarm algorithm, by using appropriate correction factors;

– to implement adaptive strategies for searching for food sources, by the flock agents of the combined swarm algorithm, by ranking the priority of search areas;

– to take into account the presence of a predator while choosing food sources by the flock agents of the combined swarm algorithm, by determining undesirable search areas by the flock agents of the combined swarm algorithm;

– to take into account the available computing resources of the state analysis system of complex dynamic objects by the flock agents of the combined swarm algorithm, by using the procedure for comparing the available amount of computing resources with those required for involvement;

– to change the search area of separate flock individuals of the combined swarm algorithm, by ranking the priority of search locations by the flock agents of the combined swarm algorithm;

– to change the speed of movement of the flock agents of the combined swarm algorithm, by adapting the speed of movement of the flock agents of the combined swarm algorithm;

– to carry out the initial setting of flock agents of the combined swarm algorithm, taking into account the type of uncertainty, by using appropriate correction factors;

– to carry out accurate training of flock agents of the combined swarm algorithm, by using the method of deep learning of knowledge bases of the flock agents of the combined swarm algorithm;

– to determine the best individuals of the flock of the combined swarm algorithm using an improved genetic algorithm;

– to conduct a local and global search taking into account the noise degree of data on the state of complex dynamic objects, by applying the appropriate correction factors of data noise;

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

– to be used as a universal tool for solving the problem of analyzing the state of complex dynamic objects due to the hierarchical description of organizational and technical systems;

– to check the adequacy of the obtained results;

– to combine various swarm algorithms for mutual verification of the adequacy and reliability of the obtained results;

– to avoid the problem of local extremum.

2. An example of using the proposed method on the example of solving the problem of determining the composition of an operational group of troops (forces) and elements of its operational structure is given. The specified example showed a 14–20 % increase in the efficiency of data processing by usingadditional improved procedures.

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.

Financing

The research was conducted without financial support.

Data availability

The manuscript has associated data in the data repository.

Use of artificial intelligence

The authors confirm that they did not use artificial intelligence technologies while creating the presented work.

References

1. Bashkyrov, O. M., Kostyna, O. M., Shyshatskyi, A. V. (2015). Rozvytok intehrovanykh system zviazku ta peredachi danykh dlia potreb Zbroinykh Syl. Ozbroiennia ta viyskova tekhnika, 1, 35–39. Available at: http://nbuv.gov.ua/UJRN/ovt_2015_1_7
2. 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
3. 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
4. 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
5. 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
6. Shyshatskyi, A. (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. 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

8. Rotshteyn, A. P. (1999). Intellektual'nye tekhnologii identifikatsii: nechetkie mnozhestva, geneticheskie algoritmy, neyronnye seti. Vinnitsa: «UNIVERSUM», 320.
9. Ko, Y.-C., Fujita, H. (2019). An evidential analytics for buried information in big data samples: Case study of semiconductor manufacturing. Information Sciences, 486, 190–203. https://doi.org/10.1016/j.ins.2019.01.079
10. 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
11. 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
12. 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
13. 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
14. 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
15. 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
16. 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
17. Kosko, B. (1986). Fuzzy cognitive maps. International Journal of Man-Machine Studies, 24 (1), 65–75. https://doi.org/10.1016/s0020-7373(86)80040-2
18. Koval, M., Sova, O., Shyshatskyi, A., Artabaiev, Y., Garashchuk, N., Yivzhenko, Y. et al. (2022). Improving the method for increasing the efficiency of decision-making based on bio-inspired algorithms. Eastern-European Journal of Enterprise Technologies, 6 (4 (120)), 6–13. https://doi.org/10.15587/1729-4061.2022.268621
19. 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
20. 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
21. 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
22. Gorokhovatsky, V., Stiahlyk, N., Tsarevska, V. (2021). Combination method of accelerated metric data search in image classification problems. Advanced Information Systems, 5 (3), 5–12. https://doi.org/10.20998/2522-9052.2021.3.01
23. Braik, M., Ryalat, M. H., Al-Zoubi, H. (2021). A novel meta-heuristic algorithm for solving numerical optimization problems: Ali Baba and the forty thieves. Neural Computing and Applications, 34 (1), 409–455. https://doi.org/10.1007/s00521-021-06392-x
24. Meleshko, Y., Drieiev, O., Drieieva, H. (2020). Method of identification bot profiles based on neural networks in recommendation systems. Advanced Information Systems, 4 (2), 24–28. https://doi.org/10.20998/2522-9052.2020.2.05
25. Kuchuk, N., Merlak, V., Skorodelov, V. (2020). A method of reducing access time to poorly structured data. Advanced Information Systems, 4 (1), 97–102. https://doi.org/10.20998/2522-9052.2020.1.14
26. Shyshatskyi, A., Tiurnikov, M., Suhak, S., Bondar, O., Melnyk, A., Bokhno, T., Lyashenko, A. (2020). Method of assessment of the efficiency of the communication of operational troop grouping system. Advanced Information Systems, 4 (1), 107–112. https://doi.org/10.20998/2522-9052.2020.1.16
27. Raskin, L., Sira, O. (2016). Method of solving fuzzy problems of mathematical programming. Eastern-European Journal of Enterprise Technologies, 5 (4 (83)), 23–28. https://doi.org/10.15587/1729-4061.2016.81292
28. Lytvyn, V., Vysotska, V., Pukach, P., Brodyak, O., Ugryn, D. (2017). Development of a method for determining the keywords in the slavic language texts based on the technology of web mining. Eastern-European Journal of Enterprise Technologies, 2 (2 (86)), 14–23. https://doi.org/10.15587/1729-4061.2017.98750
29. Stepanenko, A., Oliinyk, A., Deineha, L., Zaiko, T. (2018). Development of the method for decomposition of superpositions of unknown pulsed signals using the secondorder adaptive spectral analysis. Eastern-European Journal of Enterprise Technologies, 2 (9 (92)), 48–54. https://doi.org/10.15587/1729-4061.2018.126578
30. Gorbenko, I., Ponomar, V. (2017). Examining a possibility to use and the benefits of post-quantum algorithms dependent on the conditions of their application. Eastern-European Journal of Enterprise Technologies, 2 (9 (86)), 21–32. https://doi.org/10.15587/1729-4061.2017.96321
31. 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