UDC 004.8 **doi:** https://doi.org/10.20998/2522-9052.2024.3.07

Svitlana Kashkevich[1], Olexander Litvinenko[1], Andrii Shyshatskyi[1], Serhii Salnyk[2], Vira Velychko[3]

[1] National Aviation University, Kyiv, Ukraine
[2] Institute of special communication and information protection, Kyiv, Ukraine
[3] Military Institute of Telecommunications and Information Technologies named after Heroes of Kruty, Kyiv, Ukraine

THE METHOD OF SELF-ORGANIZATION OF INFORMATION NETWORKS IN THE CONDITIONS OF THE COMPLEX INFLUENCE OF DESTABILIZING FACTORS

Abstract. The experience of modern military conflicts of the past decades and the experience of the Russian-Ukrainian war, requires a fundamental revision of approaches regarding the order of organizing the interaction of information networks and their individual components. Traditional approaches to organization require significant time to organize the interaction of information network elements and also depend significantly on the experience and personal qualities of administrators who operate them. That is why, in this research, the authors proposed a method of organizing information networks under the influence of destabilizing factors. The method of self-organization of information networks consists of the following sequence of actions: input of initial data, display of individuals of the combined flock on the search area, numbering of individuals in the flock of the combined algorithm; determination of the initial speed of individuals of the flock of the combined algorithm; preliminary assessment of the search (feeding) area by individuals of the combined flock; classification of food sources for agents of a combined flock; procedure for optimizing a flock of hawk agents; implementation of the coot herd optimization algorithm; combining individual optimization algorithms into a mixed one; checking the presence of predator agents of the combined flock; escape and fight with predators of combined pack agents; checking the stop criterion; training of knowledge bases of combined swarm agents, determination of the amount of necessary computing resources, intelligent decision making support system. The work of the specified method was modeled on the example of the self-organization of the information network of the operational group of troops (forces). The specified example showed an increase in the efficiency of data processing at the level of 12-17% due to the use of additional improved procedures of adding correction factors for uncertainty and noise of data, selection of combined swarm agents, crossing of different types of swarm optimization approaches, as well as training of combined swarm agents.

Keywords: information networks; bio-inspired algorithms; self-organization; fuzzy cognitive maps; uncertainty.

Introduction

Optimization is a complex process of identifying multiple solutions for a variety of functions. Many calculation tasks today belong precisely to optimization tasks [1–10]. While solving optimization tasks, decision variables are defined in such a way that information networks work at their best point (mode) according to the optimization criterion.

The problems of optimization of information networks are discontinuous, undifferentiated and also multimodal.

Thus, it is impractical to use classical gradient deterministic algorithms [8–18] to solve the tasks of self-organization of information networks.

To overcome the shortcomings of classical optimization algorithms for solving problems of self-organization of information networks, a significant number of stochastic optimization algorithms, known as metaheuristic algorithms, were created [15–25].

Swarm intelligence algorithms (swarm algorithms) are one of the types of algorithms for stochastic optimization of information networks. 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 flock optimization algorithm and the sparrow flock algorithm [24‒34].

Unfortunately, most of the basic metaheuristic algorithms mentioned above are not able to balance exploration and use, which leads to unsatisfactory performance for real tasks of self-organization of information networks [35‒48].

This motivates the implementation of various strategies to improve the speed, convergence and accuracy of metaheuristic algorithms. One of the options for increasing the efficiency of decision-making using metaheuristic algorithms is their combination, that is, adding the basic procedures of one algorithm to another [47‒60].

Considering the above, the actual scientific task is the development of the self-organization method of information networks under the complex influence of destabilizing factors with the use of artificial intelligence, which would allow you to increase the efficiency of the decisions made regarding the management of self-organization parameters of information networks with a given reliability [61–73].

The analysis of works [1–81] showed that the common shortcomings of the above-mentioned researches are:

– the absence of the possibility of forming a hierarchical system of indicators regarding the self-organization process of information networks;

– the failure to take into account the computing resources of the system that perform process self-organization analysis of information networks;

– the lack of mechanisms for adjusting the system of indicators during process self-organization evaluation of information networks;

– the lack of consideration of the type of uncertainty and noise of data about the self-organization process of information networks, which creates corresponding errors in assessing their real state;

– the lack of deep learning mechanisms of the knowledge bases characterizing the self-organization process of information networks;

– high computational complexity while calculating the self-organization process of information networks;

– the lack of consideration of computational (hardware) resources available in the system that evaluates the self-organization process of information networks;

– the absence of the possibility of determining the priority of the search for a solution, in a certain direction, regarding the state of the self-organization process of information networks.

The purpose of the research is the method self-organization development of information networks in conditions of complex influence of destabilizing factors. This will increase the efficiency of process self-organization evaluation of information networks with a given reliability and making subsequent management decisions.

This will make it possible to develop software for intelligent decision making support systems that analyze the state of complex dynamic objects.

The problem, which is solved in the research, is to increase the efficiency of decision making in tasks of managing the process of self-organization of information networks while ensuring the given reliability regardless of the hierarchy of information networks. The objects of research are information networks.

The subject of the research is the decision making process in the tasks of self-organization of information networks using an improved algorithm of a flock of hawks, a soft search procedure using a flock of coots, an improved genetic algorithm and evolving artificial neural networks.

Research hypothesis is a possibility of increasing the efficiency of self-organization of information networks with a given assessment reliability.

MathCad 14 software (USA) was used to simulate the work process of the method of self-organization of information networks. The hardware of the research process is AMD Ryzen 5.

The research is based on a combined onethe algorithm of a flock of hawks and a flock of coots - for finding a solution to the state of complex dynamic objects. To train the individuals of the combined algorithm, evolving artificial neural networks are used and to select the best individuals in the combined bio-inspired algorithm, an advanced genetic algorithm is used.

Presentation of the main research material

In order to detail the process of self-organization of information networks, we will determine the parameters by which self-organization is carried out according to the levels of interaction of open systems (Table 1).

Table 1 − **Approximate relationship between parameters and control variables by levels of the OSI model**

OSI layer	Management objects	Main optimization parameters	Controlling influence of the node
Physical	A channel within the limits of connectivity with neighboring nodes	Bandwidth, channel transmission time, battery power consumption, transmission power, etc	Power (direction) of transmission, type of modulation, type of correction code, parameters, etc
Channel	Channels within connectivity with neighboring nodes	Bandwidth and transmission time in the channel, battery energy consumption, amount of service information, etc	Channel level exchange algorithms: deterministic, random, hybrid; sizes of packets and receipts
Network	One or more transmission routes	The amount of service information, route parameters (time of construction and existence, quantity, throughput, delivery time, battery energy consumption, etc.).	Network level exchange algorithms: table, probe, hybrid, wave asymmetric, hierarchical, etc. Topology control algorithms
Transport	Information direction of communication	Bandwidth, time and variation of its transmission in direction	Queue management algorithms. Overload window size, timeout time, etc
Applied	Node, neighbor nodes, network zone, entire network	Bandwidth, transmission time and time variation, battery energy consumption, transmission security	Algorithms (protocols) of application-level information exchange, coordination and intellectualization by OSI levels

As can be seen from Table 1, the self-organization term of information networks mainly includes parameters 1-4 levels of interaction of open systems. So, in the research, we will consider the specified levels of interaction of open systems and the main parameters used in them.

The combined algorithm consists of three main actions (procedures): research, exploitation and transitional processes from research to exploitation.

The method of self-organization of information networks in conditions of destabilizing factors consists of the following sequence of actions:

Step 1. Input of initial data. At this stage, the initial data available on the self-organizing information network are entered.

Step 2. Exposure of individuals of the combined flock on the search plane.

All the listed individuals, namely hawks and coots, form a population of the combined algorithm, which can be modeled from a mathematical point of view using a matrix according to equation. The presentation of individuals of the combined algorithm is carried out taking into account the uncertainty regarding the self-organizing information network and the initialization of the basic model of its state [2, 19, 21]:

$$X = \begin{bmatrix} X_1 \\ \cdots \\ X_i \\ \cdots \\ X_N \end{bmatrix}_{N\times m} = \qquad (1)$$

$$= \begin{bmatrix} x_{1,1}\times t_{1,1} & \cdots & x_{1,d}\times t_{1,d} & \cdots & x_{1,m}\times t_{1,m} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{i,1}\times t_{i,1} & \cdots & x_{i,d}\times t_{i,d} & \cdots & x_{i,m}\times t_{i,m} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{N,1}\times t_{N,1} & \cdots & x_{N,d}\times t_{N,d} & \cdots & x_{N,m}\times t_{N,m} \end{bmatrix}_{N\times m}.$$

The main position of the flock of the combined algorithm is initialized at the beginning of the algorithm execution using equation (2):

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

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

Since the position of each individual of the combined algorithm in the solution space of the problem represents a solution to the problem, the value of the objective function can be estimated accordingly for each individual of the combined algorithm. Accordingly, the set of estimated values for the objective function can be written according to equation (3):

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

where *F* is the vector of the estimated objective function, F_i is the estimated objective function based on the *i*-th member of the combined algorithm flock.

The estimated values of the objective function provide valuable information about the quality of the solution options proposed by the swarm members of the combined population. The best value obtained for the objective function corresponds to the best member of the combined algorithm swarm (i.e., the best possible solution), and the worst value obtained for the objective function corresponds to the worst member of the combined algorithm swarm (i.e., the worst possible solution).

As at each iteration the position of the swarm of individuals of the combined algorithm in the problem solution space is updated, the best member must also be updated based on the comparison of the updated values for the objective function. At the end of the algorithm implementation, the position of the best member of the flock of the combined algorithm, obtained during the iterations of the algorithm, is presented as a solution to the problem.

Step 3. Numbering of individuals in the flock of the combined algorithm, i, iϵ [0,S]. At this stage, each individual of the flock of the combined algorithm is assigned a serial number.

Step 4. Determining the initial speed of individuals of a flock of the combined algorithm.

The initial speed v_0 of each individual of the combined algorithm is determined by the following expression:

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

In the planning of the proposed approach of the combined algorithm, the position of the population members in the problem-solving space is updated based on the simulation of the hunting strategy of the individuals of the combined flock in the wild.

Step 5. Preliminary evaluation of the search (feeding) area by individuals of the combined flock. The diet of individuals of a combined flock is diverse, for a flock of hawks it is food of animal origin and for a flock of coots it is just food of plant origin. They eat hares, birds.

Therefore, it is advisable to sort the quality of food. The choice of the place of food is carried out taking into account the degree of noise of the initial data, which is proposed in the work [30].

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

The location of the best food source (i.e., minimum suitability) is considered to be hares for hawk agents, small fish for coot agents (FS_{ht}), locations from the next three food sources are small birds (hawk agents), duckweed (coot agents) (FS_{at}) and the rest are considered regular food (FS_{nt}):

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

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

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

Step 7. Sorting of the best individuals of the flock of the combined algorithm. The selection of the best individuals of the flock of the combined algorithm is carried out using the improved genetic algorithm proposed in the work [24]. While searching for food, the strongest individual of the combined algorithm flock with the largest sizes directs another individual from the combined flock in the group to search for food. This search behavior of combined swarm agents leads to different scanning areas of the search space, which improves the research ability of agents in global search.

Steps 1−7, 10−15 are common to all individuals of the combined algorithm. The remaining procedures are unique for each of the swarm optimization algorithms.

Step 8. Procedure for optimizing a flock of hawk agents.

Step 8.1 Execution of the intelligence procedure of the algorithm of a flock of hawk agents.

The formula for updating the hawks' position at this stage is as follows:

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

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

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

where N is the size of the population, $X_i\left(t\right)$ is the vector of the current position of the i-th hawk agent from the flock.

8.2 Research phase for the hawk-agent swarm algorithm.

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

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

where E_0 is a random number between [1, 1] representing the initial energy state of the prey, T is the maximum number of iterations. When $|E|\geq 1$, the hawk agent will continue to locate prey in the target area defined as the exploration phase. In the case of $|E|<1$, the hawk agent will start hunting the prey found in the previous stage and enter the exploitation stage.

In the operation phase, there are four possible strategies, including soft siege, hard siege, soft siege with gradual rapid dives and hard siege with gradual rapid dives to simulate the process of a hawk attacking its prey. r represents the probability of whether the victim will be able to escape the danger before a hawk attack, which is a random number between [0,1]. If $r < 0.5$ means that the prey successfully passed through the dangerous situation, $r\geq 0.5$ means a case of unsuccessful escape. Different combinations of r-value and escape energy E correspond to different hunting strategies. When $|E|<0.5$, a tough siege is being carried out. Otherwise, a soft siege is conducted.

8.3 Implementation of the soft siege strategy by individuals from the flock of hawk agents.

A soft siege is performed when $r\geq 0.5$ and $|E|\geq 0.5$. At this stage, the position of the hawk agent is updated as follows:

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

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

$$J=2(1-r_5), \quad (13)$$

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

8.4 Execution of a hard siege strategy by hawk agents.

A hawk agent will take a tough siege when $r\geq 0.5$ and $|E|<0.5$. The mathematical description of such behavior can be presented as follows:

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

8.5 Executing a soft siege strategy with gradual rapid dives for a flock of hawk agents.

When r < 0.5 and $|E|\geq 0.5$, the hawk agent will take a soft siege with gradual rapid dives. Levi's flight is integrated into the search procedure of the hawk flock algorithm and the mathematical model of the behavior described above is as follows:

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

$$Z=Y+S\times LF(D); \quad (16)$$

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

where D is the dimension of the problem. S is a random vector whose size is $1\times D$, $F(\cdot)$ is the objective function. Only the best position between Y and Z is selected as the next position. $LF(\cdot)$ is the Lévy flight function, which is calculated as follows:

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

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

8.5 Execution of a hard siege strategy of hawk agents with gradual rapid dives

When r < 0.5 and $|E|<0.5$, the hawk agent will perform a tight siege to get close to the prey and then launch a surprise attack.

The mathematical model of this behavior is written as follows:

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

$$Z=Y+S\times LF(D), \quad (20)$$

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

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

Step 9. Execution of the coot herd optimization algorithm.

Coot flock optimization is a population-based, gradient-free optimization method that simulates the collective behavior of American coots (a small waterfowl) on the water surface. Four different irregular and regular motions are implemented in this algorithm: random motion, chain motion, position adjustment based on group leaders, and leader motion.

Coots usually live in a group and create a chain structure to move towards a target area (food). At the front of the flock are several coots, also known as flock leaders, who direct the direction and take responsibility for the entire flock. Therefore, according to the coot's habits, the original population is divided into two parts: coot-leader and coot-follower. If N is the population size, then 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.

9.1 Random movement of coot agents.

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

$$Q = rand\,(1, D).*\left(UB - LB\right)+LB, \quad (22)$$

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

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

where $X_i(t+1)$ is the position of the i-th 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):

$$A=1-t/T, \quad (24)$$

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

9.2 Chain movement of coot agents.

In the algorithm of a flock of coot agents, the average position of two people is used to perform chain movements. The new position of the follower coot is calculated as follows:

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

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

9.3 Setting positions based on group leaders.

As a general rule, the whole group is led by one of the leaders of the group in front and all coots that remain 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, using equation (26) designed to select the leader as follows:

$$k=1+\left(i \mod L\right), \quad (26)$$

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

The next position of the coot follower based on the selected leader k is calculated using equation (27).

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

where $LeaderX_k(t)$ is the position of the selected leader, r_7 is a random number in the interval [0, 1] and R denotes a random number in the interval [−1, 1].

9.4 Leadership movement of coot agents

The group must be oriented to 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 written as follows:

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

In equation (28), $gBest$ is the current optimal position, r_8 and r_9 are the random numbers in the interval [0, 1] and R is a random number in the interval [1, 1]. r_8 generates more significant stochastic movement to help the algorithm eliminate local optimal solutions. And cos($2R\pi$) is for finding the best person with different radius to get the top position. The value of B is calculated using equation (29).

$$B=2-t\times\left(1/T\right), \quad (29)$$

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

Action 10. Combining individual optimization algorithms into a mixed one.

To combine different types of natural optimization algorithms, an ensemble mutation strategy is used, which can generate various individuals to improve the global search capabilities of the hybrid algorithm, which is written as follows:

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

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

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

where V_{i1}, V_{i2} and V_{i3} are newly generated mutant positions of the i-th search agent. $R_1 \sim R_{11}$ are different integer indicators in the range [1, N]. F_1, F_2 and F_3 are scaling factors with values of 1.0, 0.8, and 1.0, respectively, $r_{10} \sim r_{12}$ are random numbers in the range [0,1]. In addition, the parameters C_1, C_2 and C_3 are equal to 0.1, 0.2 and 0.9, which indicates the speed of the crossover.

After generating candidate mutant positions V_{i1}, V_{i2} and V_{i3}, the best position V_i with the lowest fitness value will be 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).

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

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

Step 11. Checking the presence of predator agents of the combined flock. At this stage, agents check for prey. If there are predators, go to step 12. If there are no predators, go to step 11.

Step 12. Escape and struggle with predators of combined flock agents. The strategy of escaping and fighting these predators causes the combined algorithm agents to change their position near the position they are at. Simulating this natural behavior of the combined algorithm agents improves the power of using the combined algorithm in local search in the problem-solving space around potential solutions. Since this process occurs near the position of each agent of the combined swarm, it is assumed that this range of changes in the position of the agents occurs in a corresponding zone centered on each agent of the combined swarm with a certain radius. 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 to create a variable radius with iteration of the algorithm. To model this phenomenon, a neighborhood is assumed around each agent of the combined swarm, which first randomly generates a new position in this neighborhood using (34) and (35).

Then, if the value of the objective function improves, this new position replaces the previous position according to the work (36):

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

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

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

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

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. Training of the knowledge bases of agents of the combined flock.

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

Step 15. Determining the amount of necessary computing resources, intelligent decision making support system.

In order to prevent looping of calculations on steps 1−14 of the method and to increase the efficiency of calculations, the system load is additionally determined. When the defined 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 the work [31].

Discussion of the results of research on the development of a method of self-organization of information networks

The effectiveness of the method of self-organization of information networks under the influence of destabilizing factors is compared with the algorithms listed in tables 2-4.

The comparison was made with unimodal and multimodal functions. Each function is calculated for ten independent runs to better compare the results of different algorithms.

The information network of the operational grouping of troops (forces), which was formed according to the wartime states, was considered as the basis in the research.

The following raw data were used for modeling:

As a result of the analysis of the practical application of information networks and the characteristics of its elements, the following output data should be obtained:

1. The characteristics of the information networks:

– the dimension of the information network, where N is the number of nodes of the information network, d is the diameter and area of placement of elements of the information network;

– characteristics of information network nodes, which are described at different levels of presentation:

- physical (frequency, type of modulation, transmitter power, etc.),
- channel (type of channel access protocol; method of channel organization – frequency, time, code; transmission speed in the channel, etc.);
- network and transport; hardware (volume of buffers, processor characteristics, battery capacity, etc.);
- operational (ratio of nodes by mobility: low, medium and high; part of nodes and their time of operation in silent mode, etc.);

– channel characteristics (r_{ij} is the transmission rate in the channel (i, j); s_{ij} is the channel capacity);

– initial variants of the topology of the information network (N is the total number of BTM nodes or its zone, d is the network diameter, k is the number of neighbors of the i-th node, ω is the dynamics of topological changes of the network or information direction, which takes into account the speeds and directions of movement of nodes, intensity failure of radio channels and nodes as a result of fire or radio-electronic influence of the enemy).

2. Parameters of information exchange in the network:

– the requirements for the quality of information exchange: $t^{\xi}_{with\ add}$ is the limit value of the transmission delay time of the ξ-th type of traffic; $S^{\xi}_{\min}$ is the bandwidth required for the transmission of the ξ-th type of traffic; BER^{ξ}_{add} is the permissible probability of packet loss for the ξ-th type of traffic, SNR^{ξ}_{add} is the permissible signal-to-noise ratio;

– $g^{\xi}_{ij}(t)$ is the value of the input load between the i-th and j-th subscribers in the form of messages of the ξ-th type of traffic.

To evaluate the effectiveness of the proposed method of self-organization of self-organizing information networks, it is necessary to evaluate each i-th scientific result according to the j-th resource of the network

$$R_i^j = \left\{R_i^1, R_i^2, R_i^3, R_i^4\right\},$$

which is spent on obtaining management decisions W_i^l according to the layers of the OSI model.

Network model: the network is represented by an undirected graph $G = (N, k, \omega, d)$, where N is the number of nodes, k is the number of neighboring nodes for some node v_i, ω is the dynamics of network topology change; d is the maximum diameter, which is expressed in the number of retransmissions.

Assumption: during the transmission of information, changes in the network topology do not occur, service messages are transmitted without errors, the correction of routing tables and the knowledge base of the nodal control system occurs "synchronously", the sizes of information and service messages do not change and are the same for each type of message.

As can be seen from the tables 2-4 increasing the efficiency of decision making is achieved at the level of 12-17% due to the use of additional procedures.

It can be seen that the proposed method of self-organization of information networks 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 particle swarm algorithm are far from satisfactory.

Table 2 – **Effectiveness of optimization algorithms while deciding the composition of the information network**

The name of the algorithm	T_s	Optimal variables		L	Optimal cost
		T_h	R		
Algorithm for the optimization of a pack of walruses	0.7280271	0.3845792	40.312284	200	5882.8955
Particle swarm algorithm	0.7480269	0.3845797	40.312282	200	5882.9013
Flying squirrel algorithm	0.7690308	0.384581	40.312476	199.99732	5882.9077
Artificial bee colony algorithm	1.1950157	0.64038	60.549321	48.031984	7759.8234
Ant colony algorithm	0.7780271	0.3845792	40.312284	200	5882.9013
The proposed method	0.7794994	0.385819	40.386517	200	5909.3749
Algorithm of a flock of monkeys	0.911517	0.4510723	46.230782	133.83941	6270.8621
The bat swarm algorithm	0.8344267	0.4164052	43.217775	163.90679	6003.8497
Locust swarm algorithm	0.7784599	0.3858127	40.320627	199.96442	5890.2105
Genetic algorithm	1.5622593	0.4813024	47.695987	124.64823	10,807.366
Algorithm for optimization of a flock of cats	1.1300127	1.1576349	44.110061	190.7876	11,984,417
Algorithm of invasive weeds	1.55006	0.6231249	63.139483	49.78495	9998.6395
Firefly swarm algorithm	1.406417	0.7832762	58.253368	73.964478	10,920,286

Table 3 – **Statistical results while deciding the composition of the information network**

The name of the algorithm	Average	Best	Worst	Standard	Median	Rank
Algorithm for the optimization of a pack of walruses	5882.8955	5882.8955	5882.8955	1.87E-12	5882.8955	1
Particle swarm algorithm	5891.226	5882.9013	5965.0365	22.218932	5882.9017	3
Flying squirrel algorithm	6219.5386	5882.9077	7046.3206	352.35848	6047.6955	5
Artificial bee colony algorithm	12,409,586	7759.8234	19,991,769	3127.065	11,403.338	9
Ant colony algorithm	5882.9013	5882.9013	5882.9013	3.68E-06	5882.9013	2
The proposed method	6271.132	5909.3749	6948.3792	333.1584	6143.6153	6
Algorithm of a pack of monkeys	7998.6372	6270.8621	12,805,388	1681.8974	7579.6333	8
The bat swarm algorithm	6518.1019	6003.8497	7050.4059	320.31898	6572.19	7
Locust swarm algorithm	6012.3675	5890.2105	6670.9945	239.38549	5898.5494	4
Genetic algorithm	28,273,334	10,807.366	60,311.64	13,795.65	24,975,491	12
Algorithm for optimization of a flock of cats	20,643,589	11,984,417	32,105,445	6711.6675	19,830,394	10
Algorithm of invasive weeds	29,687,575	9998.6395	50,712.307	12,915,318	32,709,339	13
Firefly swarm algorithm	25,427,766	10,920,286	45,530,922	10,828,815	22,551,255	11

Table 4 – **Effectiveness of optimization algorithms while deciding the composition of the information network**

The name of the algorithm	***h***	**Optimal variablesl *t***		***b***	Optimal cost
Algorithm for the optimization of a flock of walruses	0.2057296	3.4704887	9.0366239	0.2057296	1.7246798
Particle swarm algorithm	0.2057296	3.4704888	9.0366238	0.2057296	1.7248523
Flying squirrel algorithm	0.205056	3.4850976	9.0365299	0.2057339	1.7257923
Artificial bee colony algorithm	0.1977725	3.5270192	9.8188942	0.2163579	1.9455461
Ant colony algorithm	0.2057296	3.4704887	9.0366239	0.2057296	1.7248523
The proposed method	0.2043787	3.4924081	9.0609	0.2061055	1.7327713
Algorithm of a flock of monkeys	0.2127738	3.3465331	8.9813136	0.2191754	1.8098061
The bat swarm algorithm	0.2059617	3.465488	9.0437236	0.2060167	1.7279454
Locust swarm algorithm	0.2056085	3.4732685	9.0362859	0.2057905	1.7254435
Genetic algorithm	0.3021777	4.3080233	7.0649912	0.3989007	2.86852
Algorithm for optimization of a flock of cats	0.2833168	2.8111202	7.6140935	0.2957385	2.0415263
Algorithm of invasive weeds	0.3526171	3.4301508	7.5466754	0.5299732	3.7483578
Firefly swarm algorithm	0.2220901	6.5032092	7.9154768	0.2925869	2.6371953

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

– at the initial display of agents of the combined flock, the type of uncertainty is taken into account during their search (step 2), in comparison with works [9, 14, 21];

– the initial speed of the agents of the combined flock is taken into account (step 4), in comparison with works [9−15];

– the suitability of the search location for agents of the combined swarm is determined, which reduces the time of searching for a solution (step 5), in comparison with works [14, 16, 17];

– the universality of strategies for finding food places, which allows you to classify the type of data to be processed (steps 6, 7), in comparison with works [14, 16, 17];

– there is a classification of food sources, which determines the priority of finding a solution (step 6), in comparison with works [11, 13, 17−19];

– the possibility of combining various search strategies of swarm algorithms, due to the use of the ensemble mutation strategy (step 10), in comparison with works [1-22];

– the presence of strategies for both adjustable global and adjustable local optimizations, which allows you to adapt the method for solving non-typical tasks (steps 8, 9), in comparison with works [1−22];

– taking into account the presence of a predator, which allows you to avoid local optima (steps 9, 10), in comparison with works [12, 13, 15−18];

– taking into account the degree of distortion (unreliability) of a priori information while determining the place of food (step 5), in comparison with works [12, 13, 15−20];

– accelerated selection of individuals in the flock due to the use of an improved genetic algorithm (step 7), in comparison with works [9, 12, 13−18];

– the universality of solving the task of self-organization of information networks due to the hierarchical nature of its description (steps 1–15), in comparison with works [9, 12, 13−18];

– the possibility of quick search for solutions due to the simultaneous search for a solution by several individuals of the combined algorithm (steps 1−15, tables 2−4);

– the adequacy of the obtained results (steps 1−15), in comparison with works [9−23].

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

– the possibility of in-depth learning of knowledge bases of agents of the combined algorithm (step 14), in comparison with works [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 15), in comparison with works [9−23].

The disadvantages of the proposed method include:

– the loss of informativeness while evaluating the process of self-organization of information networks due to the construction of the function of belonging;

– lower accuracy of assessment on a single parameter of assessment of the process of self-organization of information networks;

– 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 self-organization of information networks;

– to determine effective measures to increase the efficiency of the process of self-organization of information networks;

– to increase the speed of managing the process of self-organization of information networks;

– 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 initial state of the information network, the need to take into account the delay time for collecting and proving information from sources of information extraction.

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

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

Conclusions

1. The research proposed a method of self-organization of special-purpose information networks, which, thanks to additional and improved procedures, allows you:

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

– to implement adaptive strategies for finding food sources;

– to combine individual search swarm strategies into a single strategy;

– to take into account the presence of a predator while choosing food sources;

– to take into account the available computing resources of the system while implementing self-organization of information networks;

– to change the search area by individual agents of the combined algorithm swarm;

– to change the speed of movement of agents of the combined algorithm flock;

– to take into account the priority of searching for swarm agents of the combined algorithm;

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

– to carry out accurate training of individuals of the flock of the combined algorithm;

– to determine the best individuals of the flock of the combined algorithm with the help of an improved genetic algorithm;

– to conduct a local and global search taking into account the degree of data noise in the process of self-organization of information networks;

– 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, 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 task of self-organization of information networks due to the hierarchical description of their self-organization process;

– to check the adequacy of the obtained results;

– to avoid the problem of local extremum;

– to adjust the local and global search procedures, which allows you to make the adaptation of this method to different information networks.

2. Simulation of the operation of the specified method was carried out on the example of self-organization of the information network of the operational group of troops (forces). The specified example showed an increase in the efficiency of data processing at the level of 12-17% due to the use of additional improved procedures of adding correction factors for uncertainty and noise of data, selection of combined swarm agents, crossing of different types of swarm optimization approaches and training of combined swarm agents.

References

1. Dudnyk, V., Sinenko, Yu., Matsyk, M., Demchenko, Ye., Zhyvotovskyi, R., Repilo, Iu., Zabolotnyi, O., Simonenko, A., Pozdniakov, P., Shyshatskyi, A (2020), “Development of a method for training artificial neural networks for intelligent decision support systems”, *Eastern-European Journal of Enterprise Technologies*, vol. 3, no. 2 (105), pp. 37–47, doi: https://doi.org/10.15587/1729-4061.2020.203301

2. Sova, O., Shyshatskyi, A., Salnikova, O., Zhuk, O., Trotsko, O. and Hrokholskyi, Y. (2021), "Development of a method for assessment and forecasting of the radio electronic environment", *EUREKA: Physics and Engineering*, vol. 4, pp. 30–40, doi: https://doi.org/10.21303/2461-4262.2021.001940
3. Pievtsov, H., Turinskyi, O., Zhyvotovskyi, R., Sova, O., Zvieriev, O., Lanetskii, B., and 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*, vol. 4, pp. 78−89, doi: https://doi.org/10.21303/2461-4262.2020.001353
4. Zuiev, P., Zhyvotovskyi, R., Zvieriev, O., Hatsenko, S., Kuprii, V., Nakonechnyi, O., Adamenko, M., Shyshatskyi, A., Neroznak, Y. and Velychko, V. (2020), "Development of complex methodology of processing heterogeneous data in intelligent decision support systems", *Eastern-European Journal of Enterprise Technologies*, vol. 4, no. 9 (106), pp. 14–23, doi: https://doi.org/10.15587/1729-4061.2020.208554
5. Kuchuk, N., Mohammed, A.S., Shyshatskyi, A. and 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*, vol. 8,. no. 1.2., pp. 1–6, doi: https://doi.org/10.30534/ijatcse/2019/0181.22019
6. Shyshatskyi, A., Zvieriev, O., Salnikova, O., Demchenko, Ye., Trotsko, O. and 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*, vol. 9, no. 4, pp. 5583–5590, doi: https://doi.org/10.30534/ijatcse/2020/206942020.
7. Pozna, C., Precup, R.-E., Horváth, E. and Petriu, E.M. (2022), "Hybrid Particle Filter–Particle Swarm Optimization Algorithm and Application to Fuzzy Controlled Servo Systems", *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 10, pp. 4286−4297, Oct. 2022, doi: https://doi.org/10.1109/TFUZZ.2022.3146986
8. Yang X.S. and Deb. S (2014), "Cuckoo search: recent advances and applications", *Neural Computing and Applications*, vol. 24, pp. 169–174, doi: https://doi.org/10.1007/s00521-013-1367-1
9. Mirjalili, S. (2015), "The ant lion optimizer", *Advances in Engineering Software*, vol. 83, pp. 80–98, doi: https://doi.org/10.1016/j.advengsoft.2015.01.010.
10. Yu, J..J.Q and Li, V.O.K. (2015), "A social spider algorithm for global optimization", *Applied Soft Computing*, vol. 30, pp. 614–627, doi: https://doi.org/10.1016/j.asoc.2015.02.014
11. Mirjalili, S., Mirjalili, S.M. and Lewis, A. (2014), "Grey Wolf Optimizer", *Advances in Engineering Software*, vol. 69, pp. 46–61, doi: https://doi.org/10.1016/j.advengsoft.2013.12.007
12. Koval, V., Nechyporuk, O., Shyshatskyi, A., Nalapko, O., Shknai, O., Zhyvylo, Y., Yerko, V., Kreminskyi, B., Kovbasiuk, O. and Bychkov, A. (2023), "Improvement of the optimization method based on the cat pack algorithm", *Eastern-European Journal of Enterprise Technologies*, vol. 1, no. 9 (121), pp. 41–48, doi: https://doi.org/10.15587/1729-4061.2023.273786
13. Gupta, E. and Saxena, A. (2015), "Robust generation control strategy based on Grey Wolf Optimizer", *Journal of Electrical Systems*, vol. 11, pp. 174–188, available at: https://journal.esrgroups.org/jes/article/view/387
14. Chaman-Motlagh, A. (2015), "Superdefect Photonic Crystal Filter Optimization Using Grey Wolf Optimizer", *IEEE Photonics Technology Letters*, vol. 27, no. 22, pp. 2355−2358, doi: https://doi.org/10.1109/LPT.2015.2464332
15. Nuaekaew, K., Artrit, P., Pholdee, N. and Bureerat, S. (2017), "Optimal reactive power dispatch problem using a two-archive multi-objective grey wolf optimizer". *Expert Systems with Applications*. vol. 87, pp. 79–89, doi: https://doi.org/10.1016/j.eswa.2017.06.009
16. Koval M., Sova O., Shyshatskyi A., Orlov O., Artabaiev Yu., Shknai O., Veretnov A., Koshlan O., Zhyvylo Ye. and Zhyvylo I. (2022), "Improvement of complex resource management of special-purpose communication systems", *Eastern-european journal of enterprise technologies*, vol. 5, no. 9 (119), pp. 34−44, doi: https://doi.org/10.15587/1729-4061.2022.266009
17. Ali, M., El-Hameed, M.A. and Farahat, M.A. (2017) "Effective parameters' identification for polymer electrolyte membrane fuel cell models using grey wolf optimizer". *Renewable Energy*. vol. 111, pp. 455–462, doi: https://doi.org/10.1016/j.renene.2017.04.036
18. Zhang, S. and Zhou, Y (2017), "Template matching using grey wolf optimizer with lateral inhibition", *Optik*, vol. 130, pp. 1229–1243, doi: https://doi.org/10.1016/j.ijleo.2016.11.173
19. Khouni, S.E., Menacer, T. (2024), "Nizar optimization algorithm: a novel metaheuristic algorithm for global optimization and engineering applications", *The Journal of Supercomputing*, vol. 80, pp. 3229–3281, doi: https://doi.org/10.1007/s11227-023-05579-4
20. Saremi, S, Mirjalili, S. and Lewis, A (2017), "Grasshopper optimisation algorithm: theory and application", *Advances in Engineering Software*, vol. 105, pp. 30–47, doi: 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*, vol. 174, 114685, doi: https://doi.org/10.1016/j.eswa.2021.114685
22. Thamer, K. A., Sova, O., Shaposhnikova, O., Yashchenok, V., Stanovska, I., Shostak, S., Rudenko, O., Petruk, S., Matsyi, O., and Kashkevich, S. (2024), "Development of a solution search method using a combined bio-inspired algorithm", *Eastern-European Journal of Enterprise Technologies*, vol. 1, no. 4 (127), pp. 6–13, doi: https://doi.org/10.15587/1729-4061.2024.298205.
23. Yapici, H. and Cetinkaya, N. (2019), "A new meta-heuristic optimizer: Pathfinder algorithm", *Applied Soft Computing*, vol. 78, pp. 545–568, doi: https://doi.org/10.1016/j.asoc.2019.03.012
24. Duan, H. and Qiao, P. (2014), "Pigeon-inspired optimization: a new swarm intelligence optimizer for air robot path planning", *International Journal of Intelligent Computing and Cybernetics*, vol. 7, iss. 1, pp. 24–37, doi: https://doi.org/10.1108/IJICC-02-2014-0005
25. Shyshatskyi, A., Romanov, O., Shknai, O., Babenko, V., Koshlan, O., Pluhina, T., Biletska, A., Stasiuk, T., Kashkevich, S., and Kryvosheiev, V. (2023), "Development of a solution search method using the improved emperor penguin algorithm". *Eastern-European Journal of Enterprise Technologies*, vol. 6, no. 4 (126), pp. 6–13, doi: https://doi.org/10.15587/1729-4061.2023.291008
26. Yang, X.S (2012), "Flower pollination algorithm for global optimization", *Unconventional computing and natural computation*, pp. 240–249, doi: https://doi.org/10.1007/978-3-642-32894-7_27.
27. Gomes, G.F, da Cunha, S.S. and Ancelotti, A.C (2019), "A sunflower optimization (SFO) algorithm applied to damage identification on laminated composite plates", *Engineering with Computers*, vol. 35, iss. 2, pp. 619–626, doi: https://doi.org/10.1007/s00366-018-0620-8.
28. Mehrabian, A.R. and Lucas, C (2006), "A novel numerical optimization algorithm inspired from weed colonization", *Ecological Informatics*. vol. 1, Iss. 4, pp. 355–366, doi: https://doi.org/10.1016/j.ecoinf.2006.07.003

29. Qi, X., Zhu, Y., Chen, H. and Niu B. (2013), "An idea based on plant root growth for numerical optimization", *Intelligent Computing Theories and Technology*, Berlin, Heidelberg, pp. 571–578, doi: https://doi.org/10.1007/978-3-642-39482-9_66
30. Bezuhlyi, V., Oliynyk, V., Romanenko I., Zhuk, O., Kuzavkov, V., Borysov, O., Korobchenko, S., Ostapchuk, E., Davydenko, T., and Shyshatskyi, A. (2021), "Development of object state estimation method in intelligent decision support systems", *Eastern-European Journal of Enterprise Technologies*, vol. 5, no. 3 (113), pp. 54–64, doi: https://doi.org/10.15587/1729-4061.2021.239854
31. Mahdi, Q. A., Shyshatskyi, A., Prokopenko, Y., Ivakhnenko, T., Kupriyenko, D., Golian ,V., Lazuta, R., Kravchenko, S., Protas, N. and Momit, A.(2021), "Development of estimation and forecasting method in intelligent decision support systems". *Eastern-European Journal of Enterprise Technologies*, vol. 3, no. 9(111), pp. 51–62, doi: https://doi.org/10.15587/1729-4061.2021.232718.
32. Sova, O., Radzivilov, H., Shyshatskyi, A., Shevchenko, D., Molodetskyi, B., Stryhun, V., Yivzhenko, Yu., Stepanenko, Ye., Protas, N. and Nalapko, O. (2022), "Development of the method of increasing the efficiency of information transfer in the special purpose networks", *Eastern-european Journal of Enterprise Techno*logies, vol. 3, no. 4 (117)), pp. 6–14, doi: https://doi.org/10.15587/1729-4061.2022.259727
33. Zhang, H, Zhu, Y, and Chen, H (2014), "Root growth model: a novel approach to numerical function optimization and simulation of plant root system". *Soft Computing*, vol. 18, iss. 3, pp. 521–537, doi: https://doi.org/10.1007/s00500-013-1073-z
34. Labbi, Y, Attous, D.B., Gabbar, H.A Mahdad, B. and Zidan, A. (2016), "A new rooted tree optimization algorithm for economic dispatch with valve-point effect", *International Journal of Electrical Power & Energy Systems*, vol. 79, pp. 298–311, doi: 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*, vol. 29, iss.1–2, pp. 115–123, doi: https://doi.org/10.1016/S0168-1699(00)00139-3
36. Zhao, S, Zhang, T, Ma, S and Chen, M. (2022), "Dandelion optimizer: a nature−inspired metaheuristic algorithm for engineering applications", *Engineering Applications of Artificial Intelligence*, vol. 114, 105075, doi: https://doi.org/10.1016/j.engappai.2022.105075
37. Paliwal, N., Srivastava, L. and Pandit, M. (2022), "Application of grey wolf optimization algorithm for load frequency control in multi-source single area power system", *Evolutionary Intelligence*, vol. 15, pp. 563–584, doi: https://doi.org/10.1007/s12065-020-00530-5
38. Kovalenko, A. and Kuchuk, H. (2022), "Methods to Manage Data in Self-healing Systems", *Studies in Systems, Decision and Control*, vol. 425, pp. 113–171, doi: https://doi.org/10.1007/978-3-030-96546-4_3
39. Dorigo, M. and Blum, C. (2005), "Ant colony optimization theory: a survey", *Theoretical Computer Science*, vol. 344, pp. 243–278, doi: https://doi.org/10.1016/j.tcs.2005.05.020
40. Poli, R, Kennedy, J and Blackwell, T. (2007), "Particle swarm optimization: an overview", *Swarm Intelligence*, vol. 1, pp. 33–57, doi: https://doi.org/10.1007/s11721-007-0002-0
41. Kuchuk, G., Kovalenko, A., Komari, I.E., Svyrydov, A. and Kharchenko, V. (2019), "Improving Big Data Centers Energy Efficiency: Traffic Based Model and Method", Kharchenko V., Kondratenko Y., Kacprzyk J. (eds) Green IT Engineering: Social, Business and Industrial Applications, *Studies in Systems, Decision and Control*, vol 171, Springer, Cham, doi: https://doi.org/10.1007/978-3-030-00253-4_8
42. Bansal, J.C., Sharma, H., Jadon, S.S. and Clerc, M. (2014), "Spider Monkey Optimization algorithm for numerical optimization", *Memetic Computing*, vol. 6, pp. 31–47, doi: https://doi.org/10.1007/s12293-013-0128-0
43. Yeromina, N., Kurban, V., Mykus, S., Peredrii, O., Voloshchenko, O., Kosenko, V., Kuzavkov, V., Babeliuk, O., Derevianko, M. and. Kovalov, H. (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*, vol. 11, no. 05, pp. 37–41, doi: https://doi.org/10.46338/ijetae0521_05
44. Maccarone, A.D., Brzorad, J.N. and Stone, H.M. (2008), "Characteristics and Energetics Of Great Egret And Snowy Egret Foraging Flights", *Waterbirds*, vol. 31, no. 4, pp. 541–549, available at: https://www.jstor.org/stable/40212108
45. Ramaji, I. J. and Memari, A. M. (2018),. "Interpretation of structural analytical models from the coordination view in building information models", *Automation in Construction*, no. 90, pp. 117–133, doi: https://doi.org/10.1016/j.autcon.2018.02.025
46. Mukhin, V., Kuchuk, N., Kosenko, N., Kuchuk, H. and Kosenko, V. (2020),"Decomposition Method for Synthesizing the Computer System Architecture , Advances in Intelligent Systems and Computing", *AISC*, vol. 938, pp 289–300, doi: https://doi.org/10.1007/978-3-030-16621-2_27
47. Pérez-González, C. J., Colebrook, M., Roda-García, J. L. and Rosa-Remedios, C. B. (2019), "Developing a data analytics platform to support decision making in emergency and security management",. *Expert Systems with Applications*, no. 120, pp. 167–184, doi: https://doi.org/10.1016/j.eswa.2018.11.023
48. Chen, H. (2018), "Evaluation of Personalized Service Level for Library Information Management Based on Fuzzy Analytic Hierarchy Process", *Procedia Computer Science*, no. 131, pp. 952–958, doi: https://doi.org/10.1016/j.procs.2018.04.233
49. Mozhaev, O., Kuchuk, H., Kuchuk, N., Mykhailo, M. and Lohvynenko, M. (2017), "Multiservice network security metric", *2nd International Conference on Advanced Information and Communication Technologies*, AICT 2017 – Proceedings, pp. 133–136, doi: https://doi.org/10.1109/AIACT.2017.8020083
50. Chan, H. K., Sun, X. and Chung, S.-H. (2019), "When should fuzzy analytic hierarchy process be used instead of analytic hierarchy process?", *Decision Support Systems*. pp. 1–37, doi: https://doi.org/10.1016/j.dss.2019.113114
51. Petrovska, I. and Kuchuk, H. (2023), "Adaptive resource allocation method for data processing and security in cloud environment", *Advanced Information Systems*, vol. *7*(3), pp. 67–73, doi: https://doi.org/10.20998/2522-9052.2023.3.10
52. Osman, A. M. S., (2019), "A novel big data analytics framework for smart cities", *Future Generation Computer Systems*, vol. 91., pp. 620–633, doi: https://doi.org/10.1016/j.future.2018.06.046.
53. Nechyporuk, O., Sova, O., Shyshatskyi, A., Kravchenko, S., Nalapko, O., Shknai, O., Klimovych, S., Kravchenko, O., Kovbasiuk, O. and Bychkov, A. (2023), "Development of a method of complex analysis and multidimensional forecasting of the state of intelligence objects", *Eastern-European Journal of Enterprise Technologies*, vol. 2, no. 4 (122), pp. 31–41, doi: https://doi.org/10.15587/1729-4061.2023.276168.

54. 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*, vol. 33, pp. 292–303, doi: https://doi.org/10.1016/j.asoc.2015.04.048
55. Poliarus, O., Krepych, S., and Spivak, I. (2023), “Hybrid approach for data filtering and machine learning inside content management system”, *Advanced Information Systems*, vol. 7, no. 4, pp. 70–74, doi: https://doi.org/10.20998/2522-9052.2023.4.09
56. Kuchuk, N., Mozhaiev, O., Semenov, S., Haichenko, A., Kuchuk, H., Tiulieniev, S., Mozhaiev, M., Davydov, V., Brusakova, O. and Gnusov, Y. (2023), “Devising a method for balancing the load on a territorially distributed foggy environment”, *Eastern-European Journal of Enterprise Technologies*, vol. 1(4 (121), pp. 48–55, doi: https://doi.org/10.15587/1729-4061.2023.274177
57. Balochian, S and Baloochian, H. (2019), “Social mimic optimization algorithm and engineering applications”, *Expert Systems with Applications*, vol. 134, pp. 178–191, doi: https://doi.org/10.1016/j.eswa.2019.05.035
58. Melvix, J. L (2014), “Greedy politics optimization. Metaheuristic inspired by political strategies adopted during state assembly elections”, *International advance computing conference (IACC)*, IEEE, pp. 1157–1162, doi: https://doi.org/10.1109/IAdCC.2014.6779490.
59. Moosavian, N. and Roodsari, B.K. (2013), “Soccer league competition algorithm, a new method for solving systems of nonlinear equations,. *International Journal of Intelligence Science*. vol. 4, no. 1, pp. 7–16, doi: https://doi.org/10.4236/ijis.2014.41002
60. Hayyolalam, V, Kazem, A.A.P (2020), “Black widow optimization algorithm: a novel meta-heuristic approach for solving engineering optimization problems”, *Engineering Applications of AI*, vol. 87, doi: https://doi.org/10.1016/j.engappai.2019.103249
61. Abualigah, L., Yousri D., Elaziz M.A. and Gandomi A.H. (2021), “Aquila optimizer: a novel meta-heuristic optimization algorithm”, *Computers & Industrial Engineeringvol*, 157, 107250, doi: https://doi.org/10.1016/j.cie.2021.107250
62. Hodlevskyi, M., and Burlakov, G. (2023), “Information technology of quality improvement planning of process subsets of the spice model”, *Advanced Information Systems*, vol. 7, no. 4, pp. 52–59, doi: https://doi.org/10.20998/2522-9052.2023.4.06
63. Askari, Q., Younas, I. and Saeed, M. (2020), “Political optimizer: A novel socio-inspired meta-heuristic for global optimization”, *Knowledge-Based Systems*, vol. 195, 105709, doi: https://doi.org/10.1016/j.knosys.2020.105709
64. Kovalenko, A., Kuchuk, H., Kuchuk, N. and Kostolny, J. (2021), “Horizontal scaling method for a hyperconverged network”, *2021 International Conference on Information and Digital Technologies* (IDT), Zilina, Slovakia, doi: https://doi.org/10.1109/IDT52577.2021.9497534
65. Mohamed, A.W, Hadi, A.A and Mohamed, A.K (2020), “Gaining-sharing knowledge based algorithm for solving optimization problems: a novel nature-inspired algorithm”. *International Journal of Machine Learning and Cybernetics*. vol. 11, iss. 7, pp. 1501–1529, doi: https://doi.org/10.1007/s13042-019-01053-x
66. Gödri, I., Kardos, C., Pfeiffer, A. and Váncza, J. (2019), “Data analytics-based decision support workflow for high-mix low-volume production systems”, *CIRP Annals*, vol. 68, no.1, pp. 471–474, doi: https://doi.org/10.1016/j.cirp.2019.04.001
67. Harding, J. L. (2013), “Data quality in the integration and analysis of data from multiple sources: some research challenges”, *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XL-2/W1, pp. 59–63, doi: https://doi.org/10.5194/isprsarchives-XL-2-W1-59-2013
68. Dotsenko, N., Chumachenko, I., Galkin, A., Kuchuk, H. and Chumachenko, D. (2023), “Modeling the Transformation of Configuration Management Processes in a Multi-Project Environment”, *Sustainability (Switzerland)*, vol. 15(19), 14308, doi: https://doi.org/10.3390/su151914308
69. Orouskhani, M., Orouskhani, Y., Mansouri, M. and Teshnehlab, M. (2013), “A novel cat swarm optimization algorithm for unconstrained optimization problems”, *International Journal “Information Technology and Computer Science”*, no. 11, pp. 32−41, doi: https://doi.org/10.5815/ijitcs.2013.11.04
70. Karaboga, D. and Basturk B. (2007), “A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm”, *Journal of Global Optimization*. vol. 39, pp. 459–471, doi: https://doi.org/10.1007/s10898-007-9149-x
71. Fister, I., Yang X.S. and Brest J. (2013), “A comprehensive review of firefly algorithms”. *Swarm and Evolutionary Computation*, vol. 13, pp. 34–46, doi: https://doi.org/10.1016/j.swevo.2013.06.001
72. Sova, O., Radzivilov, H., Shyshatskyi, A., Shvets, P., Tkachenko, V., Nevhad, S., Zhuk, O., Kravchenko, S., Molodetskyi, B. and Miahkykh, H. (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*, vol. 2, no. 3 (116), pp. 6–14, doi: https://doi.org/10.15587/1729-4061.2022.254122.
73. Khudov, H., Khizhnyak , I., Glukhov, S.., Shamrai, N., and Pavlii, V. (2024), “The method for objects detection on satellite imagery based on the firefly algorithm”, *Advanced Information Systems*, vol. 8, no. 1, pp. 5–11, doi: https://doi.org/10.20998/2522-9052.2024.1.01
74. Owaid, S. R., Zhuravskyi, Y., Lytvynenko, O., Veretnov, A., Sokolovskyi, D., Plekhova, G., Hrinkov, V., Pluhina, T., Neronov, S., and Dovbenko, O. (2024), “Development of a method of increasing the efficiency of decision-making in organizational and technical systems”, *Eastern-European Journal of Enterprise Technologies*, vol. 1, no. 4 (127), pp. 14–22, doi: https://doi.org/10.15587/1729-4061.2024.298568
75. Kuchuk, G., Nechausov, S. and Kharchenko, V. (2015), “Two-stage optimization of resource allocation for hybrid cloud data store”, *International Conference on Information and Digital Technologies, Zilina*, pp. 266–271, doi: http://dx.doi.org/10.1109/DT.2015.7222982
76. Tyurin, V., Bieliakov, R., Odarushchenko, E., Yashchenok, V., Shaposhnikova, O., Lyashenko, A., Stanovskyi, O., Melnyk, B., Sus, S., and Dvorskyi, M. (2023). “Development of a solution search method using an improved locust swarm algorithm”, *Eastern-European Journal of Enterprise Technologies*, vol. 5, no. 4 (125), pp. 25–33, doi: https://doi.org/10.15587/1729-4061.2023.287316
77. Hunko, M., Tkachov, V., Kuchuk, H. and Kovalenko, A. (2023), “Advantages of Fog Computing: A Comparative Analysis with Cloud Computing for Enhanced Edge Computing Capabilities”, *2023 IEEE 4th KhPI Week on Advanced Technology, KhPI Week 2023 - Conference Proceedings*, code 194480, doi: https://doi.org/10.1109/KhPIWeek61412.2023.10312948
78. Yakymiak, S., Vdovytskyi, Y., Artabaiev, Y., Degtyareva, L., Vakulenko, Y., Nevhad, S., Andronov, V., Lazuta, R., Shapoval, P., and Artamonov, Y. (2023), “Development of the solution search method using the population algorithm of global search optimization”, *Eastern-European Journal of Enterprise Technologies*, vol. 3, no. 4 (123), pp. 39–46, doi: https://doi.org/10.15587/1729-4061.2023.281007

79. Petrovska, I., Kuchuk, H., Mozhaiev, M. (2022), Features of the distribution of computing resources in cloud systems, 2022 *IEEE 4th KhPI Week on Advanced Technology, KhPI Week 2022 - Conference Proceedings*, 03-07 October 2022, Code 183771, doi: https://doi.org/10.1109/KhPIWeek57572.2022.9916459
80. Mohammed, B. A., Zhuk, O., Vozniak, R., Borysov, I., Petrozhalko, V., Davydov, I., Borysov, O., Yefymenko, O., Protas, N., and Kashkevich, S. (2023), "Improvement of the solution search method based on the cuckoo algorithm", *Eastern-European Journal of Enterprise Technologies*, vol. 2, no. 4 (122), pp. 23–30, doi: https://doi.org/10.15587/1729-4061.2023.277608.
81. Arora S.and Singh, S (2019), "Butterfly optimization algorithm: a novel approach for global optimization", *Soft Computing*, vol. 3, pp. 715–734, doi: https://doi.org/10.1007/s00500-018-3102-4

Надійшла (received) 08.04.2024
Прийнята до друку (accepted for publication) 17.07.2024

Відомості про авторів / About the Authors

Кашкевич Світлана Олександрівна – старший викладач кафедри комп'ютеризованих систем управління, Національний авіаційний університет, Київ, Україна;
Svitlana Kashkevich – Senior Lecturer of the Department of Computerized Management Systems, National Aviation University, Kyiv, Ukraine;
e-mail: svitlana.kashkevych@npp.nau.edu.ua; ORCID Author ID: https://orcid.org/0000-0002-4448-3839;
Scopus ID: https://www.scopus.com/authid/detail.uri?authorId=58244269900.

Литвиненко Олександр Євгенійович – доктор технічних наук, професор, завідувач кафедри комп`ютеризованих систем управління, Національний авіаційний університет, Київ, Україна;
Olexander Litvinenko – Doctor of Technical Sciences, Professor, Head of the Department of Computerized Control Systems, National Aviation University, Kyiv, Ukraine;
e-mail: oleksandr.lytvynenko@npp.nau.edu.ua; ORCID Author ID: https://orcid.org/0000-0002-8862-8032;
Scopus ID: https://www.scopus.com/authid/detail.uri?authorId=14063763700.

Шишацький Андрій Володимирович – доктор технічних наук, старший дослідник, доцент кафедри комп'ютеризованих систем управління, Національний авіаційний університет, Київ, Україна;
Andrii Shyshatskyi – Doctor of Technical Sciences, Senior Researcher, Associate Professor of the Department of Computerized Control Systems, National Aviation University, Kyiv, Ukraine;
e-mail: ierikon13@gmail.com; ORCID Author ID: https://orcid.org/0000-0001-6731-6390;
Scopus ID: https://www.scopus.com/authid/detail.uri?authorId=57201773300.

Сальник Сергій Васильович – кандидат технічних наук, заступник начальника Науково-дослідного центру Інституту спеціального зв'язку та захисту інформації, НТУ України "КПІ імені Ігоря Сікорського", Київ, Україна;
Serhii Salnyk – Candidate of Technical Sciences, Deputy head of Scientific Research Center, Institute of special communication and information protection, The National Technical University of Ukraine "I. Sikorsky KPI", Kyiv, Ukraine;
e-mail: s.sergey@i.ua; ORCID Author ID: https://orcid.org/0000-0003-4463-5705;
Scopus ID: https://www.scopus.com/authid/detail.uri?authorId=57226650377.

Величко Віра Петрівна – викладач кафедри автоматизованих систем управління, Військовий інститут телекомунікацій та інформатизації імені Героїв Крут, Київ, Україна;
Vira Velychko – Lecturer, Department of Automated Control Systems, Military Institute of Telecommunications and Information Technologies named after Heroes of Kruty, Kyiv, Ukraine;
e-mail: veravel@gmail.com; ORCID Author ID: https://orcid.org/0000-0001-9654-4560;
Scopus ID: https://www.scopus.com/authid/detail.uri?authorId=57219249989.

Метод самоорганізації інформаційних мереж в умовах комплексного впливу дестабілізуючих факторів

С. О. Кашкевич, О. Є. Литвиненко, А. В. Шишацький, С. В. Сальник, В. П. Величко

Анотація. Досвід сучасних воєнних конфліктів останніх десятиріч, а також досвід російсько-української війни вимагає фундаментального перегляду підходів, щодо порядку організації взаємодії інформаційних мереж та їх окремих складових. Традиційні підходи з організації вимагають суттєвого часу на організацію взаємодії елементів інформаційної мережі, а також суттєво залежать від досвіду та особистих якостей адміністраторів, які їх експлуатують. Саме тому, в даному дослідженні авторами було запропоновано метод організації інформаційних мереж в умовах впливу дестабілізуючих факторів. Метод самоорганізації інформаційних мереж складається з наступної послідовності дій: введення вихідних даних, виставлення особин комбінованої зграї на площі пошуку, нумерація особин в зграї комбінованого алгоритму; визначення початкової швидкості особин зграї комбінованого алгоритму; попереднє оцінювання ділянки пошуку (харчування) особинами комбінованої зграї; класифікація джерел їжі для агентів комбінованої зграї; процедура оптимізації зграї агентів-яструбів; виконання алгоритму оптимізації зграї лиски; об'єднання окремих алгоритмів оптимізації в змішаний; перевірка наявності хижака агентів комбінованої зграї; втеча та боротьба з хижаками агентів комбінованої зграї; перевірка критерію зупинки; навчання баз знань агентів комбінованої зграї, визначення кількості необхідних обчислювальних ресурсів, інтелектуальної системи підтримки прийняття рішень. Проведено моделювання роботи зазначеного методу на прикладі самоорганізації інформаційної мережі оперативного угруповання військ (сил). Зазначений приклад показав підвищення ефективності оперативності обробки даних на рівні 12–17 % за рахунок використання додаткових удосконалених процедур додавання корегувальних коефіцієнтів щодо невизначеності та зашумленості даних, відбору агентів комбінованої зграї, схрещування різнотипних підходів ройової оптимізації, а також навчання агентів комбінованої зграї.

Ключові слова: інформаційні мережі; біоінспіровані алгоритми; самоорганізація; нечіткі когнітивні карти; невизначеність.