DOI: 10.15587/978-617-8360-13-9.CH4

Svitlana Kashkevich, Olha Matsyi, Anastasiia Voznytsia,
Oleksiy Buyalo, Daniil Krant, Kostiantin Radchenko

CHAPTER 4

SCIENTIFIC AND METHODOLOGICAL APPARATUS FOR PROCESSING DIVERSE DATA IN AUTOMATED CONTROL SYSTEMS

ABSTRACT

This section of the study proposes the conceptual foundations for the use of artificial intelligence in intelligent decision support systems.

In the course of the research, the authors:

– justified the feasibility of using artificial intelligence theory for processing heterogeneous data in automated control systems;

– developed a methodology for data distribution in automated control systems;

– designed a model for evaluating the process of heterogeneous data processing in automated troop control systems using expert information;

– improved the methodology for configuring an information system to evaluate the process of heterogeneous data processing in automated control systems under conditions of uncertainty.

The analysis conducted in the study established that the application of fuzzy graphs and the mathematical apparatus of fuzzy logic in decision support tasks for data distribution and the evaluation of heterogeneous data processing under various conditions, including uncertainty, enables the distribution of data among elements of automated control systems based on the importance of the elements and the number of features in real-time.

The methodology for rational data distribution based on the importance of automated control system elements and the number of features in such systems under conditions of uncertainty has been improved. This methodology differs from existing ones by combining the mathematical apparatus of information theory, fuzzy logic, and expert evaluation, enabling the formalization of features in a unified parameter space and the intellectualization of information processing processes.

A quantitative assessment of the proposed methodology's efficiency was conducted. The results of this assessment demonstrated that data distribution among the elements of automated control systems based on importance and the number of features using the proposed methodology improves the timeliness of data processing and decision-making regarding the state of the heterogeneous data processing process by 15–17 %.

An enhanced methodology for configuring the information system for evaluating the heterogeneous data processing process in automated control systems under conditions of uncertainty, utilizing

a genetic algorithm, was developed. This methodology addresses limitations of other methods in varying specific features while holding other indicators constant, thus improving the efficiency of the developed information system for evaluating heterogeneous data processing in automated control systems.

The scientific outcome is the improvement of the genetic algorithm for differentiated tuning of the fuzzy knowledge base of the information system for evaluating heterogeneous data processing in automated control systems based on posterior data.

A quantitative assessment of the improved methodology's effectiveness was performed. The results indicated that the proposed methodology enhances the timeliness of configuring the information system for processing heterogeneous data in automated control systems under conditions of uncertainty.

KEYWORDS

Artificial intelligence, heterogeneous data processing, automated control systems, reliability, and timeliness.

4.1 JUSTIFICATION FOR THE FEASIBILITY OF USING ARTIFICIAL INTELLIGENCE THEORY FOR PROCESSING HETEROGENEOUS DATA IN AUTOMATED CONTROL SYSTEMS

The purpose of this section is to justify the necessity of applying the theory of fuzzy graphs to describe the process of heterogeneous data processing in automated control systems (ACS) [1–20].

One of the possible ways to model the process of heterogeneous data processing in ACS is through the application of fuzzy graph theory, whose primary advantage lies in its ability to adequately represent output data in relation to input information that is characterized by weakly structured (fuzzy) indicators. This advantage makes fuzzy graph theory applicable in tasks involving the analysis of operational situations under conditions of uncertainty [21–39].

The model of the heterogeneous data processing process in ACS can be represented in the form of a knowledge matrix (knowledge base) (**Table 4.1**), which contains quantitative and qualitative features and characteristics of ACS functioning [40–59].

A knowledge matrix [60, 61] is a table formed according to the following rules:

1. The dimensionality of the matrix $(n+1)\times N$, where $(n+1)$ – the number of columns, and $N=k_1+k_2+\ldots+k_m$ – the number of rows.

2. The first n the columns of the matrix correspond to the input variables $i=\overline{1,n}$, and $(n+1)$-th a column corresponds to the values dj of the output variable $y\left(j=\overline{1,m}\right)$.

3. Each row of the matrix represents a specific combination of knowledge about the input variables, which has been assigned by an expert to one of the possible values of the output variable y. At the same time: first k_j rows correspond to the value of the output variable $y=d_1$, second k_2 rows correspond to the value $y=d_2$, last k_m – value $y{=}d_m$.

4. Element a_i^{jp}, that lies at the intersection *i*-th of the column and *jp*-th the row corresponds to the linguistic evaluation of the parameter x_1 in the row of the fuzzy knowledge base with the number *jp*. At the same time, the linguistic evaluation a_i^{jp} is selected from the term set of the corresponding variable x_i, so $a_i^{jp} \in A_i$, $i=\overline{1,n}$, $j=\overline{1,m}$, $p=\overline{1,k_j}$.

Table 4.1 Model of ACS functioning over a time interval

Feature vector number at the input	Features of ACS elements (input variables)				Situation assessment decision (Output variable)
	x_1	x_2	… x_i …	x_n	y
11	a_1^{11}	a_2^{11}	… a_i^{11} …	a_n^{11}	d_1
12	a_1^{12}	a_2^{12}	… a_i^{12} …	a_n^{12}	
…	…	…	…	…	
$1k_1$	$a_1^{1k_1}$	$a_2^{1k_1}$	… $a_i^{1k_1}$ …	$a_n^{1k_1}$	
…					
j_1	a_1^{j1}	a_2^{j1}	… a_i^{j1} …	a_n^{j1}	d_j
j_2	a_1^{j2}	a_2^{j2}	… a_i^{j2} …	a_n^{j2}	
…	…	…	…	…	
jk_j	$a_1^{jk_j}$	$a_2^{jk_j}$	… $a_i^{jk_j}$ …	$a_n^{jk_j}$	
…					
$m1$	a_1^{m1}	a_2^{m1}	… a_i^{m1} …	a_n^{m1}	d_m
$m2$	a_1^{m2}	a_2^{m2}	… a_i^{m2} …	a_n^{m2}	
…	…	…	…	…	
mk_m	$a_1^{mk_m}$	$a_2^{mk_m}$	… $a_i^{mk_m}$ …	$a_n^{mk_m}$	

The presented model structurally consists of layers of features (sets of input informational arrays) at specific time intervals and possible variants of ACS functioning (sets of decisions). Decision-making is performed at each stage, taking into account the features of ACS functioning [62–86].

Hierarchical Fuzzy Inference Systems. For modeling multidimensional "input-output" dependencies, it is advisable to use hierarchical fuzzy inference systems. In such systems, the output of one knowledge base serves as the input to another, higher-level hierarchy. Hierarchical knowledge bases lack feedback loops. **Fig. 4.1** presents an example of a hierarchical fuzzy system that models the dependency $y=f(x_1, x_2, x_3, x_4, x_5, x_6)$ using three knowledge bases f_1, f_2, f_3.

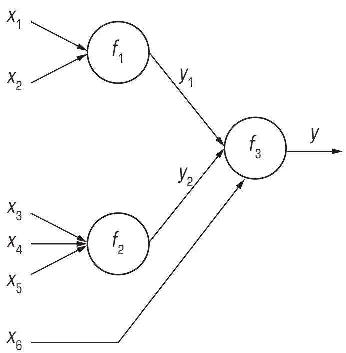

Fig. 4.1 Example of a hierarchical fuzzy inference system

These knowledge bases describe dependencies $y_1=f_1(x_1, x_2)$, $y_2=f_2(x_4, x_5, x_6)$ and $y=f_3(y_1, x_3, y_2)$. The application of hierarchical fuzzy knowledge bases allows overcoming the "curse of dimensionality". Another advantage of hierarchical knowledge bases is their compactness. A small number of fuzzy rules in hierarchical knowledge bases can adequately describe multidimensional "input-output" dependencies.

Let's assume that five terms are used for the linguistic evaluation of variables. In this case, the maximum number of rules required to define the dependency $y=f(x_1, x_2, x_3, x_4, x_5, x_6)$ using a single knowledge base will amount to $5^6=15625$. In fuzzy inference using a hierarchical knowledge base, the defuzzification and fuzzification procedures for intermediate variables y_1 and y_2 (**Fig. 1**) are not performed. The result of the logical inference in the form of a fuzzy set $\tilde{y}^*=\left(\frac{\mu_1(X^*)}{\tilde{d}_1},\frac{\mu_2(X^*)}{\tilde{d}_2},\frac{\mu_m(X^*)}{\tilde{d}_m}\right)$ is directly transmitted to the fuzzy inference engine of the next level in the hierarchy. Therefore, for intermediate variables in hierarchical fuzzy knowledge bases, it is sufficient to define only the term sets without describing the membership functions.

4.2 METHODOLOGY FOR DATA DISTRIBUTION IN AUTOMATED CONTROL SYSTEMS

The essence of the data distribution methodology in automated control systems (ACS) lies in the rational allocation of data among ACS elements.

By selecting the most critical data sources and an optimal number of gradations, it ensures the desired probability of correctly identifying the type of data circulating within the system.

To choose the best plan for distributing data sources among ACS elements, the following partial quality indicators for distribution are used:

1. Completeness of ACS Element Coverage by Observation: this is calculated as the ratio of the sum of importance coefficients of ACS elements Y_j included in the distribution plan to the sum of importance coefficients of all ACS elements:

$$\Pi = \frac{\sum_{j=1}^{\{m\}_u} Y_j}{\sum_{j=1}^{J} Y_j}, \tag{4.1}$$

where $\{m\}_u$ – the number of ACS elements selected for observation in u-th distribution plan.

2. The expenditure of technical resources for ACS load, determined as the sum of the technical resource expenditures of the ACS $S_{sum_{ACS}}$.

3. The probability of tracking the status and nature of activities of the entire set of ACS elements subject to distribution – $\bar{P}$:

$$\bar{P} = \frac{1}{m}\sum_{j=1}^{m} P_j, \tag{4.2}$$

where P_j – the probability of tracking the status and nature of activities of ACS elements included in the distribution plan.

Then, the system of partial quality indicators for selecting a data distribution plan among ACS elements will have the following form:

$$\begin{cases} \Pi \to \max; \\ S_{sum_{ACS}} \to \min; \\ \bar{P} \to \max. \end{cases} \tag{4.3}$$

Taking into account the system of partial indicators, the functional reflecting the quality of data distribution among ACS elements, depending on the selection of a specific distribution plan option Π, can be expressed as:

$$F_{\Pi_{U_{opt}}} = \max F(\Pi_{\Pi_U}, S_{sum_{ACS_{\Pi_U}}}, \bar{P}_{\Pi_U}). \tag{4.4}$$

However, in existing methodologies for information distribution based on the importance of ACS elements, the calculation of importance coefficients is performed implicitly (4.4), and the procedure for their calculation is not defined.

Thus, a relevant scientific problem arises: multi-criteria optimization of the data distribution process among ACS elements, considering their importance, to improve data processing efficiency.

Then, (4.4) can be rewritten as:

$$F_{\Pi_{U_{opt}}} = \arg\max_{im \in Im} F(Im, S_{sum_{ACS_{\Pi_U}}}, \bar{P}_U), \tag{4.5}$$

where Im – the vector of importance (priority) coefficients of ACS elements in the observation range; P_{Π_U} – the probability of tracking the status and nature of an ACS element's activity when selecting u-th a distribution plan.

The importance of an ACS element can be considered as a non-metric utility criterion (NMUC). The main challenge in solving this problem lies in representing the NMUC in a quantitative form for its subsequent integration into a utility function (UF).

To represent the NMUC in a quantitative form, non-metric partial utility criteria (NMPC) have been defined to characterize the importance of an ACS element.

The main NMPCs include the degree of task priority for which the distribution is carried out or the priority level of the ACS element (X_{pr}); the degree of informational value (X_{inf}); the degree of operational value of the ACS element (X_{op}).

Let $Q(X_{pr}, X_{inf}, X_{op})$, denote the utility function of the NMPC. X_{pr}, X_{inf}, X_{op} independent systems of values. Then the utility functions of the NMPC can be represented by the following system of expressions:

$$\begin{cases} \psi_{pr} = Q(X_{pr}) f(X_{pr}), \\ \psi_{inf} = Q(X_{inf}) f(X_{inf}), \\ \psi_{op} = Q(X_{op}) f(X_{op}), \end{cases} \tag{4.6}$$

where $f\left(X_{pr}\right), f\left(X_{inf}\right), f\left(X_{op}\right)$ – functions of utility dependence on metric criteria.

In turn, the utility function of an ACS element will be expressed as:

$$\psi = Q(X_{pr}, X_{inf}, X_{op}) f(X_{pr}, X_{inf}, X_{op}). \tag{4.7}$$

To study the impact of non-metric criteria, let's introduce a constraint, the essence of which is that the influence of metric criteria is equivalent, meaning there is no dependence on metric criteria:

$$f(X_{pr}) = f(X_{inf}) = f(X_{op}) = 1. \tag{4.8}$$

Analysis of the constraint (4.8) reveals that the indicators are equivalent to each other concerning the metric criterion. In turn, the utility dependence functions on non-metric criteria vary linearly and are determined by the lower and upper values of the accepted evaluations. By performing normalization based on the maximum value and the adopted scale, any preference for one of the indicators in expression (4.8) concerning a non-metric criterion will lead to the dominance of the utility function of the corresponding indicator.

Considering (4.8), expression (4.7) can be represented as:

$$\psi = Q(X_{pr}, X_{inf}, X_{op}). \tag{4.9}$$

For the purpose of selecting the rational form of the utility function $Q(X_{pr}, X_{inf}, X_{op})$ it is convenient to represent X_{pr}, X_{inf}, X_{op} in the form of fuzzy sets [63, 64], with NMPC evaluations as their elements, respectively. Then $Q(X_{pr}, X_{inf}, X_{op})$ it can be identified with the membership function of the set of input values of the primary NMPC indicators x_{pr}, x_{inf}, x_{op} to fuzzy sets X_{pr}, X_{inf}, X_{op}, accordingly. Thus, the task of determining the importance of an ACS element can be formulated as a decision-making problem regarding the importance of the ACS element, and the result of the decision-making process can be represented as:

$$\mathrm{Im} = Q(x_{pr}, x_{inf}, x_{op}), \tag{4.10}$$

where x_{pr}, x_{inf}, x_{op} – a set of input values of the primary NMPC indicators; Im – a decision regarding the determination of the importance of the ACS element.

The task of decision-making regarding the determination of the importance of the ACS element is to, based on information about the vector of input indicators $(x_{pr}, x_{inf}, x_{op})$ determine the outcome Im. A necessary condition for the formal solution of the stated problem is the presence of the dependency (4.10). To establish such a dependency, it is necessary to consider the input indicators (NMPC) and the output decision as linguistic variables defined on universal sets.

To evaluate such linguistic variables, it is proposed to use qualitative terms that form a term set:

$X_{inf} = \{L, bA, M, aA, H\}$ – the term set of a variable x_{inf},

$X_{op} = \{L, bA, M, aA, H\}$ – the term set of a variable x_{op},

$X_{pr} = \{L, M, H\}$ – the term set of a variable x_{pr},

$\mathrm{Im} = \{L, bA, M, aA, H\}$ – the term set of a variable Im,

where L, bA, M, aA, H – respectively "low", "below average", "average", "above average", "high"; Im – the set of variables characterizing the importance of an ACS element:

$$\begin{aligned} X_{inf} &= [1,5], \\ X_{op} &= [1,5], \\ X_{pr} &= [1,3], \\ \mathrm{Im} &= [1,5]. \end{aligned} \tag{4.11}$$

To evaluate the values of linguistic variables x_{pr}, x_{inf}, x_{op}, in accordance with (4.11), let's use the corresponding scale of qualitative terms.

In accordance with cognitive engineering methods for knowledge base synthesis, knowledge bases have been developed that characterize the importance of elements. Using the mathematical apparatus of fuzzy set theory, the knowledge base is transformed into logical equations.

$$\mu^{\mathrm{Im}_j}(X_{pr}, X_{inf}, X_{op}) = \max_J \left\{ \min_i \left[\mu^J(x_{i(pr)}), \mu^J(x_{i(inf)}), \mu^J(x_{i(op)}) \right] \right\}, \tag{4.12}$$

where μ – the membership functions of the corresponding linguistic variables $x_{pr}, x_{inf}, x_{op}, \mathrm{Im}$, the sets $X_{pr}, X_{inf}, X_{op}, \mathrm{Im}$, $J \in \{L, bA, M, aA, H\}$; $x_{i(pr)} \in [1,2,3]$; $x_{i(inf)} \in [1,2,3,4,5]$; $x_{i(op)} \in [1,2,3,4,5]$; $\mathrm{Im}_i \in [1,2,3,4,5]$.

From the analysis of the numerical results of the experiment, it was concluded that the decision regarding the importance of the elements of the automated control system (ACS) is determined by the expression:

$$\frac{x_{inf} + x_{op}}{2} \cdot x_{pr} = \mathrm{Im}. \tag{4.13}$$

The minimum value that expression (4.13) can take is Im=1, then the maximum value Im=15. Let's define the FN (membership function) for the term sets of the importance of the elements of the ACS.

For this, it is possible to normalize the measurement intervals of each variable to a single universal interval [0, 4] using the following relationship:

$$\mu^j(\mathrm{Im}_i) = \tilde{\mu}^j(u), u = 4\frac{\mathrm{Im}_i - \underline{\mathrm{Im}}}{\overline{\mathrm{Im}} - \underline{\mathrm{Im}}}, j = L, bA, M, aA, H. \tag{4.14}$$

The analytical model of the membership function is represented by the expression:

$$\mu^j(u) = \frac{1}{1 + \left(\frac{u-b}{c}\right)^2}, \tag{4.15}$$

where the parameters b and c are set based on the results of the previous assessment of the functioning of the ACS.

The graphical representation of the membership function according to expression (4.15) is shown in **Fig. 4.2.**

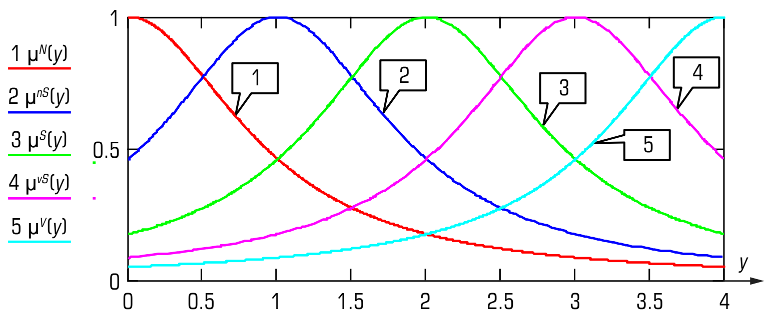

Fig. 4.2 Graphical representation of the membership function of the fuzzy set of the importance degrees of an element in the automated control system (ACS)

CHAPTER 4

Calculation of coefficients PI. Priority coefficient is determined x_{pr}, coefficient of the degree of informational value x_{inf}, coefficient of the degree of operational value x_{op}.

Coefficient of the degree of operational value OP. Using (4.13), the values of the importance of the corresponding element of the ACS are calculated.

Determination of the linguistic value of the importance of the ACS element.

Optimization of the feature vector of the functioning of the ACS element.

It is known that as the number of features that characterize the functioning of the ACS element increases, the time required for identifying its operational mode and other costs, primarily hardware costs, also increase, which in turn reduces the operational efficiency of the evaluation process.

Identifying informative features in a real situation is a complex task, especially when evaluating the functioning process of system elements in an ACS, where the feature set can be very large and the features themselves may be correlated with each other. Therefore, the task is to select and extract the most informative features to reduce the dimensionality of the input data vector, while simultaneously finding a coordinate system in which the probability of correctly recognizing ACS elements will be maximized or sufficient for decision-making.

Reducing the dimensionality of the feature space in the presence of a large number of ACS system elements plays a significant role, as it increases the throughput of the ACS system's channels as a whole. This is because an increase in the number of features that characterize an ACS element significantly leads to an increase in identification errors.

The dependence of the identification probability of the functional process of an ACS element on the dimensionality of the feature space is shown in **Fig. 4.3**.

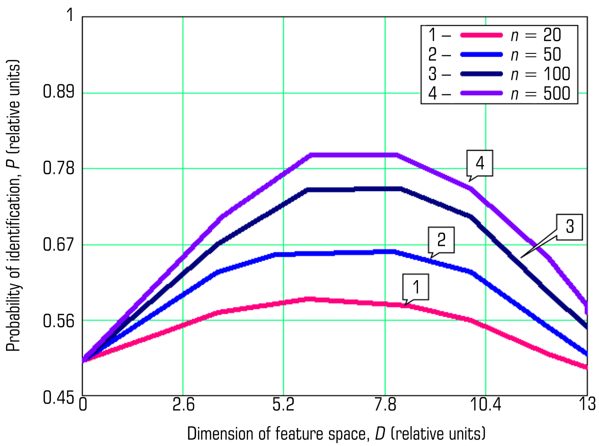

Fig. 4.3 The dependence of the identification probability of the operational mode of an ACS element on the dimensionality of the intelligence feature

The graphs show that arbitrary increases in the dimensionality of the feature space may lead to a deterioration in the probability of correct recognition.

The formation of the feature vector for the ACS element can be mathematically represented as:

$$Y = AX, \tag{4.16}$$

where X – the feature vector that characterizes the operation process of the ACS element; Y – the vector of possible decisions; A – the transformation matrix.

Checking the optimal dimensionality of the feature vector. The condition for the termination of the grading elimination cycle is the value of the informativeness loss threshold for all features $\left(\sum \Delta I_k\right)_{\max}$ or for a specific feature. It is also possible to set a maximum number of gradations that need to be retained during the minimization process.

Fig. 4.2 shows the curves of changes in the informativeness of features depending on the number of their gradations. Analysis of these curves for all features allows minimizing the number of gradations in terms of the memory usage of the identification device and the total loss of informativeness for identification parameters. The presented dependencies suggest that the number of gradations for features should be chosen to be no more than 4–6 (at the inflection point of most curves), which coincides with the results presented in **Fig. 4.3**.

The adjustment of system parameters is carried out based on an improved methodology, which is based on the use of a genetic approach. This method facilitates the correction of the system parameters.

4.3 MODEL OF THE PROCESS OF EVALUATING THE PROCESSING OF HETEROGENEOUS DATA IN AN AUTOMATED CONTROL SYSTEM (ACS) USING EXPERT INFORMATION

Let the following be known: the set of solutions $D = \{d_j\}$, $\left(j = \overline{1,m}\right)$, that corresponds to the result of evaluating the processing of various types of data in an automated control system (ACS) y; the set of input indicators $X = \{x_i\}$, $\left(i = \overline{1,n}\right)$; the ranges of quantitative variation for each input information; membership functions that allow representing the indicators $x_i \in \left[\underline{x_i}, \overline{x_i}\right]$, $i = \overline{1,n}$, x_i, $i = \overline{1,n}$ in the form of fuzzy sets; a knowledge matrix defined by rules (**Table 4.1**). It can be graphically represented as shown in **Fig. 4.4**.

Let's consider the application of the model for utilizing expert information to synthesize an algorithm for evaluating the processing of various types of data in an automated control system (ACS).

From the analysis of the functioning of ACS elements under various situational conditions, the evaluation directions have been identified: the similarity of situational indicators and their changes during the operation of the ACS.

Let's describe the model for evaluating the processing of various types of data in an automated control system (ACS):

$$D(k) = f\begin{bmatrix} Y_1(k-1), Y_2(k-1), \ldots, Y_{14}(k-1), Q(k-1), R(k-1), \\ Z_1(k-1), \ldots, Z_4(k-1) \end{bmatrix}, \tag{4.17}$$

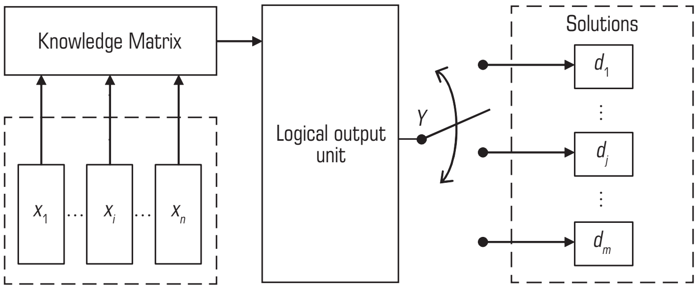

Fig. 4.4 Model of the operational situation assessment process

where $Y_1(k-1)$ – a vector that characterizes the operating mode of ACS element No. 1 at k–1 modeling step; $Y_2(k-1)$ – a vector that characterizes the operating mode of an ACS element at k–1 modeling step; $Y_{14}(k-1)$ – a vector that characterizes the operating mode of ACS element No. 14 at k–1 modeling step; $Q(k-1)$ – a vector that characterizes the operating mode of the control and communication system of ACS element No. 1; $R(k-1)$ – a vector that characterizes the operating mode of the control and communication system of an ACS element; $Z_1(k-1),\ldots,Z_4(k-1)$ – vectors that characterize the operating modes of control and communication systems of group ACS elements.

In turn, the vectors of the data processing evaluation process in an ACS are determined by the following indicators: $Y_1,\ldots,Y_{14},Q,R,Z_1,\ldots,Z_4=\{k_{11}(x),\ldots,k_{145}(x)\}$.

For indicators with quantitative measurements, the range of variation is divided into four quanta. This ensures the possibility of transforming a continuous universal set $U=[\underline{u},\overline{u}]$ into a discrete five-element set:

$$U=\{u_1,u_2,\ldots,u_5\},$$

where $u_1=\underline{u}$, $u_2=\underline{u}+\Delta_1$, $u_3=u_2+\Delta_2$, $u_4=u_3+\Delta_3$, $u_5=\overline{u}$, and $\Delta_1+\Delta_2+\Delta_3+\Delta_4=\overline{u}-\underline{u}$, $\overline{u}(\underline{u})$ – the upper (lower) boundary of the indicator's range of variation. Thus, all pairwise comparison matrices have a dimension. The choice of four quanta is determined by the possibility of approximating nonlinear curves through five points.

For evaluating the values of linguistic variables, it is possible to use the following scale of qualitative terms.

CHAPTER 4

In the general case, the input variables $x_1, x_2, \ldots, x_n$ can be defined as a number, a linguistic term, or based on the thermometer principle [51, 56].

The evaluation of the data processing process in an ACS using expert information is carried out using fuzzy logical equations, which represent a knowledge matrix and a system of logical statements. These equations allow for the calculation of membership function values for various identification results at fixed input indicator values. As the outcome of the evaluation process for data processing in an ACS, the decision with the highest membership function value will be accepted.

Linguistic evaluations $\underline{\alpha_i^{jp}}$ variables $x_1, x_2, \ldots, x_n$, that are part of the logical statements regarding decisions $d_j, j = \overline{1,m}$, will be considered as fuzzy sets defined on universal sets $X_i = \left[\underline{x_i}, \overline{x_i}\right], i = \overline{1,n}$.

So $\mu^{a_i^{jp}}(x_i)$ – (MF) of the indicator $x_i \in \left[\underline{x}, \overline{x}\right]$ to a fuzzy term $\alpha_i^{jp}, i = \overline{1,n}, j = \overline{1,m}, p = \overline{1,l_j}$; $\mu^{d_j}(x_1, x_2, \ldots, x_n)$ – (MF) of the input variables vector $X = (x_1, x_2, \ldots, x_n)$ to the value of the output evaluation $y = d_j, j = \overline{1,m}$.

The relationship between these functions is determined by a fuzzy knowledge base and can be represented in the form of the following logical equations:

$$\begin{gathered}\mu^{d_j}(x_1, x_2, \ldots, x_n) = \mu^{a_1^{j1}}(x_1) \wedge \mu^{a_2^{j1}}(x_2) \wedge \ldots \wedge \mu^{a_n^{j1}} \vee \\ \mu^{a_1^{j2}}(x_1) \wedge \mu^{a_2^{j2}}(x_2) \wedge \ldots \wedge \mu^{a_n^{j2}}(x_n) \ldots \\ \ldots \mu^{a_1^{jl_j}}(x_1) \wedge \mu^{a_2^{jl_j}}(x_2) \wedge \ldots \wedge \mu^{a_n^{jl_j}}(x_n), j = \overline{1,m}.\end{gathered} \tag{4.18}$$

The equations are derived from the fuzzy knowledge base by replacing variables (linguistic terms) with their membership functions (MFs), and the AND and OR operations with the respective operations $\wedge$ and $\vee$.

Briefly, the system (4.18) can be written as follows:

$$\mu^{d_j}(x_i) = \bigvee_{p=1}^{l_j} \left[\bigwedge_{i=1}^{n} \mu^{a_i^{jp}}(x_i) \right], j = \overline{1,m}. \tag{4.19}$$

Fuzzy logical equations are an analog of the fuzzy inference procedure introduced by Zadeh, which is performed using the "fuzzy (min-max) composition" operation, where the $\wedge$ and $\vee$ operations correspond to the min and max operations. From (4.19), let's obtain:

$$\mu^{d_j}(x_i) = \max_{p = \overline{1,l_j}} \left\{ \min_{j = \overline{1,n}} \left[\mu^{a_i^{jp}}(x_i) \right] \right\}. \tag{4.20}$$

From expression (4.20), it is evident that for the calculations it is only necessary to have the membership functions (MFs) of the variables to the fuzzy terms.

CHAPTER 4

4.4 ENHANCED METHODOLOGY FOR CONFIGURING THE INFORMATION SYSTEM FOR EVALUATING THE PROCESS OF PROCESSING VARIOUS TYPES OF DATA IN AN ACS UNDER CONDITIONS OF UNCERTAINTY

The essence of the methodology for configuring the information system for evaluating the process of processing various types of data in an ACS under conditions of uncertainty lies in selecting the weight coefficients of production rules, minimizing the error between the reference and experimental decisions.

Identification based on fuzzy logical inference is carried out in accordance with the defined knowledge base:

IF $(x_1 = a_{1,j1})$ AND $(x_2 = a_{2,j1})$ AND ...AND $(x_n = a_{n,j1})$ with weight w_{j1},

OR $(x_1 = a_{1,j2})$ AND $(x_2 = a_{2,j2})$ AND ...AND $(x_n = a_{n,j2})$ with weight w_{j2},

OR $(x_1 = a_{1,jk_j})$ AND $(x_2 = a_{2,jk_j})$ AND ...AND $(x_n = a_{n,jk_j})$ with weight w_{jk_j}, (4.21)

THEN $y = d_j, j = \overline{1,m}$,

where $a_{i,jp}$ – the fuzzy term that evaluates the variable x_i in the row with number $jp(p = \overline{1,k_j})$, i.e., $a_{i,jp} = \int \mu_{jp}(x_i)/x_i$; k_j – the number of rows-conjunctions in which the output y is evaluated by the value d_j; $w_{jp} \in [0,1]$ – the weight coefficient of the rule with the number jp.

Functions of correspondence in the process of handling different types of data $X^* = \left(x_1^*, x_2^*, \ldots, x_n^*\right)$ are calculated for the classes d_j as follows:

$$\mu_{d_j}\left(X^*\right) = \underset{p=\overline{1,k_j}}{\vee} w_{jp} \cdot \underset{i=\overline{1,n}}{\wedge}\left(\mu_{jp}\left(x_i^*\right)\right), j = \overline{1,m}, \tag{4.22}$$

where $\mu_{jp}\left(x_i^*\right)$ – the input correspondence function x_i^* an unclear term $a_{i,jp}$; $\wedge(\vee)$ – s-norm (t-norm), which in classification tasks usually corresponds to the maximum (minimum).

As a solution, the class with the maximum matching function of the calculated solution is selected $d_1 \ldots d_m$:

$$y^* = \underset{\{d_1, d_2, \ldots, d_m\}}{\arg} \max\left(\mu_{d_1}\left(X^*\right), \mu_{d_2}\left(X^*\right), \ldots, \mu_{d_m}\left(X^*\right)\right). \tag{4.23}$$

Thus, the adaptation or adjustment of the information system for evaluating the processing of various types of data under uncertainty conditions will be performed.

The work applies an adaptation method based on solving the optimization problem using the genetic algorithm method.

Let's introduce the constraint that there is a reference sample from M a pair of experimental data that link the inputs $X = \left(x_1, x_2, \ldots, x_n\right)$ with the output y of the dependency being studied:

$$\left(X_r, y_r\right), \left(r = \overline{1,M}\right), \tag{4.24}$$

where $X_r = \left(x_{r,1}, x_{r,2}, \ldots, x_{r,n}\right)$ – the input vector in r-th pair; y – the corresponding output.

CHAPTER 4

Tuning the model involves finding such parameters of the matching functions for the input variable terms and the weighting coefficients of the rules that minimize the deviation between the expected and obtained results on the reference sample. The proximity criterion can be defined in various ways.

The first method involves selecting the classification error percentage on the reference sample used for training the system as the tuning criterion. Let's introduce the following notation:

P – the vector of parameters of the matching functions for the input and output variables;

W – the vector of the weighting coefficients of the knowledge base rules;

$F(X_r,P,W)$ – the output result according to the knowledge base with the parameters (P,W) with the input values X_r.

Then the tuning of the fuzzy model is reduced to the following optimization problem: find such a vector (P,W), to:

$$\frac{1}{M}\sum_{r=\overline{1,M}}\Delta_r \to \min, \tag{4.25}$$

where Δ_r – classification error of the state of processing various types of data X_r:

$$\Delta_r = \begin{cases} 1, if \quad y_r \neq F(X_r,P,W); \\ 0, if \quad y_r = F(X_r,P,W). \end{cases} \tag{4.26}$$

The advantage of the tuning criterion lies in its simplicity and clear substantive interpretation. The error percentage is widely used as a training criterion for various pattern recognition systems.

The objective function of the optimization problem takes on discrete values, which complicates the use of gradient optimization methods. It is particularly difficult to select the necessary parameters of gradient algorithms (for example, the increment of arguments for calculating partial derivatives) when tuning a fuzzy classifier on a small data sample.

The second method involves using the distance between the output in the form of a fuzzy set as the tuning criterion $\left(\frac{\mu_{d_1}(x)}{d_1},\frac{\mu_{d_2}(x)}{d_2},\ldots,\frac{\mu_{d_m}(x)}{d_m}\right)$ and the value of the output variable in the reference sample, which is intended for system training. For this purpose, the output variable of the reference sample (4.23) is fuzzified as follows:

$$\left.\begin{array}{l} \tilde{y} = (1/d_1, 0/d_2, \ldots, 0/d_m), if \quad y = d_1 \\ \tilde{y} = (0/d_1, 1/d_2, \ldots, 0/d_m), if \quad y = d_2 \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ \tilde{y} = (0/d_1, 0/d_2, \ldots, 1/d_m), if \quad y = d_m \end{array}\right\}. \tag{4.27}$$

CHAPTER 4

In this case, tuning the fuzzy classifier is reduced to the following optimization problem: find such a vector (P,W), so that:

$$\frac{1}{M}\cdot\sum_{r=1}^{M}\sum_{j=1}^{m}\left(\mu_{d_j}(y_r)-\mu_{d_j}(X_r,P,W)\right)^2 \to \min, \tag{4.28}$$

where $\mu_{d_j}(y_r)$ – membership function of the output variable value y in r-th the pair of the reference sample to the decision d_j; $\mu_{d_j}(X_r,P,W)$ – membership function of the fuzzy model output with parameters (P,W) to the decision d_j, with the input values from r-the pair of the reference sample (X_r).

The objective function in problem (4.27) does not have extensive plateaus, so it is suitable for gradient-based optimization methods. However, the optimization results are not always satisfactory: the fuzzy knowledge base that ensures the minimum of criterion (4.27) does not always also ensure the minimum classification errors. This is explained by the fact that points close to the maxima of the class partitions usually make the same contribution to the tuning criterion, both in the case of correct classification and in the case of misclassification.

The third method inherits the advantages of the previous methods. The idea is to increase the contribution of misclassified objects to the tuning criterion by multiplying the distance $\sum_{j=1}^{m}\left(\mu_{d_j}(y_r)-\mu_{d_j}(X_r,P,W)\right)^2$ by a penalty coefficient. As a result, the optimization problem takes the form:

$$\frac{1}{M}\cdot\sum_{r=1}^{M}(\Delta_r\cdot penalty+1)\cdot\sum_{j=1}^{m}\left(\mu_{d_j}(y_r)-\mu_{d_j}(X_r,P,W)\right)^2 \to \min, \tag{4.29}$$

where $penalty > 0$ – penalty coefficient.

Problems (4.26), (4.28), and (4.29) can be solved by various optimization technologies, among which the method of steepest descent, quasi-Newton methods, and genetic algorithms are often used.

Usually, constraints are imposed on the controlled variables to ensure the linear ordering of the elements of the term sets. In addition, the cores of the fuzzy sets must not go beyond the ranges of variation of the corresponding variables. This ensures transparency, that is, meaningful interpretability of the fuzzy knowledge base after tuning. As for the vector W, its coordinates must lie in the range [0,1]. If the requirements for the interpretability of the knowledge base are high, the rule weights are not tuned, leaving them equal to 1. There is also an intermediate option where the weighting coefficients can take values of 0 and 1. In this case, a zero value of the weighting coefficient is equivalent to excluding the rule from the fuzzy knowledge base.

Parameters of the matching functions and rule weights can be tuned simultaneously or separately. When only the rule weights are tuned, the computational volume can be significantly reduced, since the membership functions $\mu_{jp}(x_i^*)$, do not depend on W. For this, at the beginning of the optimization, it is necessary to calculate the degrees of rule execution with unit weighting coefficients $(w_{jp})=1$ for each object in the reference sample:

$$g_{jp}\left(X_r\right)=\underset{i=\overline{1,n}}{\wedge}\mu_{jp}\left(x_{r,i}\right), j=\overline{1,m}, p=\overline{1,k_j}, r=\overline{1,M}.$$

For the new weighting coefficients of the membership functions for the process of processing various types of data in the ACS X_r classes d_j are calculated as follows:

$$\mu_{d_j}\left(X_r\right)=\underset{p=\overline{1,k_j}}{\vee}w_{jp}\cdot g_{jp}\left(X_r\right), j=\overline{1,m}.$$

Considering the specifics of the process of processing various types of data in an ACS, one of the ways to solve it based on fuzzy logic is to apply combined optimization methods that combine the advantages of the gradient method and the random search method. One such method is the genetic algorithm, which makes it possible to perform optimization for multimodal, non-smooth, and non-convex functions with a convergence speed greater than that of random search methods.

Thus, taking into account the identified features of the process of processing various types of data in an ACS, the hierarchical nature of the logical inference tree, and the structural-semantic model of processing various types of data in an ACS, it is advisable to carry out the process of tuning the information system for processing various types of data in a differentiated manner, i.e., by tuning the fuzzy knowledge base of each individual element of the ACS system.

Let's explore the possibilities of applying and functioning of the "genetic algorithm" for tuning the information system for assessing the operational situation.

Let's assume the following initial data are known:

S – the system structure vector that defines the system parameters that do not change during optimization (regarding the information system for assessing the operational situation – a set of IF-THEN rules represented using the mathematical apparatus of fuzzy logic, and the rule weight coefficients);

B – the reference vector that contains a set of sample stimulus-response pairs (input indicators – decisions) by which the information system for assessing the operational situation is tuned;

W_{jp} – the vector of the rule weights of the fuzzy knowledge base, whose value is being optimized;

F – the mismatch function that determines the quality of the solution proposed by the information system for assessing the operational situation compared to the solution in the reference vector;

F_{ACS} – the mismatch function that determines the quality of the solution proposed by the information system for processing various types of data regarding a separate element of the ACS, compared to the solution in the reference vector.

By decision-making in the information system for processing various types of data in an ACS, let's understand the output result provided by the system according to the entered features $x_{11}...x_{nm}$.

Let's introduce the following constraints:

– the vector B covers the entire practically significant range of solutions within the application domain of the information system for processing various types of data in the ACS;

– the vector S (IF-THEN rules) is formed in advance based on the results of working with experts and does not contain logical errors;

– based on the results of working with experts, the values of the membership function parameters have been determined;

– the mismatch function F calculates the decision-making error of the information system for processing various types of data in an ACS by the least squares method with an ordinal scale according to the formula:

$$e_{ACS} = \sum_{j=1}^{n}\sum_{i=1}^{k}(d_i^{B_j} - d_i^{O_j})^2, \tag{4.30}$$

where e_{ACS} – the tuning error of a separate ACS element; n – the dimension of the vector B; k – the maximum number of solutions issued by the information system for processing various types of data in the ACS; $d_i^{B_j}$ – the reference i-th decision for the j-th input element of vector B; $d_i^{O_j}$ – i-th the solution of the information system for processing various types of data in the ACS for j-th the input element of vector B, taking into account the rule weights W_{jp}; i – the decision number issued by the information system for assessing the operational situation, $i \in \overline{1,k}$; j – the number of the input indicator set in the reference vector B.

For tuning the knowledge base set (KBS) of a separate element of the ACS system, the proposed criterion is:

$$e_{ACS\_\min} = Min(F_{ACS}(S,W)), \tag{4.31}$$

where $e_{ACS\min}$ – the minimally acceptable final total error, as the difference between the membership function values of the decision regarding the state of operation of a separate element of the ACS system and the reference decision.

As a result of using criterion (4.30) for each separate element of the ACS, it is logical to use criterion (4.31), which indicates the tuning of the information system for processing various types of data in the ACS as a whole:

$$e_{\Sigma\min} = Min(F(S,W)), \tag{4.32}$$

where $e_{\Sigma\min}$ – the minimally acceptable final total error as the difference between the membership function values of the decision regarding the state of operation of the control and communication system element of the ACS and the reference decision.

Tuning the rule weights (W_{jp}) and the parameters of the membership functions – vector P, will be performed using the "genetic algorithm" method.

The set of indicators being optimized is combined into a parameter vector called a chromosome. Indicators in the chromosome can be stored in a regular or encoded (transformed) form.

A specific section of the chromosome responsible for encoding a single indicator is called a gene; the length of a gene depends on the chosen type of encoding.

Each chromosome represents a solution to the problem, which is optimized with an efficiency expressed by a certain number calculated using the objective function. A set of chromosomes (a collection of solutions) is called a population. The population maintains a constant number of chromosomes. The main stages of the genetic algorithm method are shown in **Fig. 4.5**.

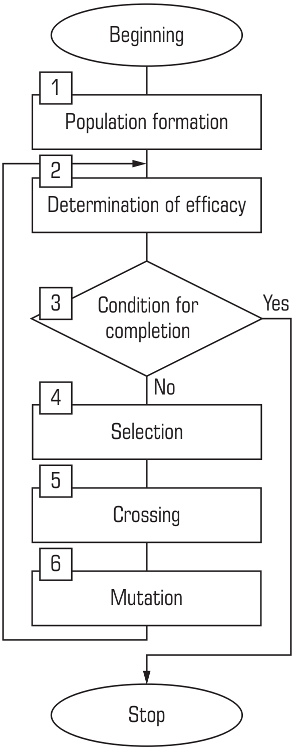

Fig. 4.5 Workflow diagram of the genetic algorithm tuning method

The formation of the initial population depends on the approach to forming chromosomes: tuning the weighting coefficients of the importance of the situation; tuning the weighting coefficients of the priority of logical rules in the knowledge base (KB) based on which decisions are made.

To reduce the volume of data arrays, it is proposed to form chromosomes where the genes are the weighting coefficients of the logical rules in the knowledge base (KB).

Formation of the initial population. The reference vector B is formed based on the results of assessing the operational situation under various conditions of operation of the ACS system elements.

Let's define the membership functions (MF) of the operational features of the ACS system element and decisions to fuzzy terms using the formula:

$$\mu^{d_j}\left(x_i\right)=\max_{p=\overline{1,l_j}}\left\{W_{jp}\min_{j=\overline{1,n}}\left[\mu^{a_i^{jp}}\left(x_i\right)\right]\right\}. \quad (4.33)$$

Let's randomly generate the rule weights W_{jp} of the knowledge base (KB) in the interval from 0 to 1.

Determining the chromosome efficiencies. The efficiency of each chromosome in the population for each set of weights W_{jp}, is determined using the objective function. Then, the chromosomes are sorted in ascending (or descending) order of their efficiency (forming a sequence based on error magnitude).

Selection. At this stage, chromosomes with the least efficiency are discarded if the total number of chromosomes in the population exceeds the permissible limit. Thus, the computational volume performed in the algorithm remains constant regardless of the iteration number.

Crossover. Two chromosomes are randomly selected from the population, considering their efficiency, and starting from a random position, they exchange genes. There can be multiple crossover points. If the two points defined by these chromosomes in the search space are in the vicinity of the same extremum, the average value between these points, resulting from the crossover, will be closer to the extremum. This is somewhat analogous to the gradient method. However, if the two points defined by these chromosomes are in the vicinity of different extrema, the average value between them will be random, akin to the random search method.

Mutation. This is fully analogous to the random search method. The values of individual genes in the population are randomly changed, i.e., the search point's position in the search space is altered.

End of the search. The search terminates if, over L L iterations, the efficiency of the best chromosome has increased by less than λ. Otherwise, a new iteration begins. A single iteration is called a generation. For instance, if the most efficient chromosome was found in the 30th generation, the condition for terminating the search algorithm was met at the 30th iteration, and the best chromosome in the final selection is considered the optimal solution.

Thus, the methodology for applying genetic algorithms to tune the information system for processing various types of data in the ACS is reduced to the following algorithm (**Fig. 4.6**):

1. Formation of reference states for the process of processing various types of data in the ACS.
2. Formation of the initial population.
3. Determination of the proximity criterion for tuning the fuzzy identifier.

4. Tuning the weights of the production rules in the knowledge base (KB) for each element of the ACS.

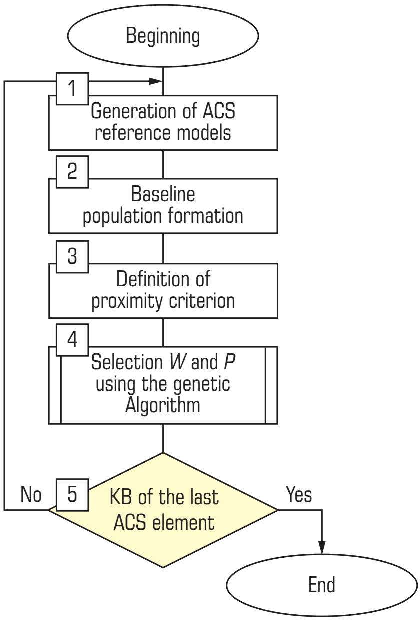

Fig. 4.6 Algorithm for applying the genetic algorithm (GA) to tune the information system for processing various types of data in the ACS

Formation of the initial population.

When calculating the rule weights in the knowledge base (KB), let's assume them to be equal to one.

Let's randomly generate the rule weights in the knowledge base (KB) within the interval from 0 to 1, i.e. $W_{jp} \in [0,1]$. Since there are 43 rules in the knowledge base (KB) and the number of chromosomes is 14, it is necessary to form a two-dimensional array of rule weight sets for tuning the information system for assessing the operational situation. The efficiency of each chromosome in the population for each set of weights is determined using the objective function W_{jp}, after that, the chromosomes are sorted in ascending (or descending) order of their efficiency.

Selection. At this stage, chromosomes with the lowest efficiency are discarded if the total number of chromosomes in the population exceeds the permissible limit. This means that the computational volume performed in the algorithm remains constant regardless of the iteration number.

Let's perform sorting of the chromosomes based on their efficiency:

$$\Delta_3, \Delta_9, \Delta_{12}, \Delta_{13}, \Delta_2, \Delta_4, \Delta_1, \Delta_{14}, \Delta_8, \Delta_7, \Delta_5, \Delta_{11}, \Delta_6, \Delta_{10}.$$

The highest efficiency, i.e., the smallest decision-making error, is held by the third and ninth chromosomes, while the lowest efficiency, i.e., the largest decision-making error, is held by the sixth and tenth chromosomes.

Crossover. In this case, for the crossover operation, chromosomes No. 3 and No. 9 are chosen as parent chromosomes. The resulting offspring chromosomes from the crossover are recorded in place of chromosomes No. 6 and No. 10, respectively.

Mutation. The mutation operation is performed on chromosome No. 10, and if the sum of the weights exceeds one, normalization is carried out: 0.15; 0.09; 0.08; 0.02; 0.2; 0.13; 0.07; 0.06; 0.16; 0.04.

Let's proceed to calculate its efficiency. The result obtained is $\Delta=1.1$, which is significantly better than the efficiency of the two previous chromosomes, the optimization of which we perform.

End of search.

CONCLUSIONS

1. Based on the analysis conducted in the study, it has been established that the application of fuzzy graphs and the mathematical apparatus of fuzzy logic in decision support tasks for data distribution and evaluating the process of processing various types of data under different conditions, including uncertainty, allows for the distribution of data between the elements of the ACS based on the importance of the ACS elements and the number of features in real-time.

2. The methodology for the rational distribution of data based on the importance of ACS elements and the number of features in the ACS under uncertainty conditions has been improved. This methodology differs from existing ones by combining the mathematical apparatus of information theory, fuzzy logic, and expert evaluation, which allowed for the formalization of features in a unified parameter space and, through the intellectualization of information processing processes, achieved more efficient data handling.

A quantitative assessment of the effectiveness of the proposed methodology has been conducted. The results of this assessment showed that the distribution of data between the elements of the ACS based on importance and the number of features using the proposed methodology increases the operational speed of data processing and decision-making regarding the state of the process of processing various types of data by 15–17 %.

3. The methodology for tuning the information system for evaluating the process of processing various types of data in the ACS under uncertainty conditions has been improved by using a genetic algorithm. In conditions where the application of other methods is limited due to the inability to vary

individual features with fixed values of other indicators, this approach has improved the operational speed of the developed information system for evaluating data processing in the ACS.

The scientific result is the improvement of the genetic algorithm for the differentiated tuning of the fuzzy knowledge base of the information system for evaluating the processing of various types of data in the ACS based on a posteriori data.

A quantitative evaluation of the effectiveness of the improved methodology has been carried out. The results of this evaluation showed that the use of the proposed methodology increases the operational efficiency of tuning the information system for processing various types of data in the ACS under uncertainty conditions.

REFERENCES

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

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

22. Thamer, K. A., Sova, O., Shaposhnikova, O., Yashchenok, V., Stanovska, I., Shostak, S. et al. (2024). Development of a solution search method using a combined bio-inspired algorithm. Eastern-European Journal of Enterprise Technologies, 1 (4 (127)), 6–13. https://doi.org/10.15587/1729-4061.2024.298205
23. Yapici, H., Cetinkaya, N. (2019). A new meta-heuristic optimizer: Pathfinder algorithm. Applied Soft Computing, 78, 545–568. https://doi.org/10.1016/j.asoc.2019.03.012
24. Duan, H., Qiao, P. (2014). Pigeon-inspired optimization: a new swarm intelligence optimizer for air robot path planning. International Journal of Intelligent Computing and Cybernetics, 7 (1), 24–37. https://doi.org/10.1108/ijicc-02-2014-0005
25. Shyshatskyi, A., Romanov, O., Shknai, O., Babenko, V., Koshlan, O., Pluhina, T. et al. (2023). Development of a solution search method using the improved emperor penguin algorithm. Eastern-European Journal of Enterprise Technologies, 6 (4 (126)), 6–13. https://doi.org/10.15587/1729-4061.2023.291008
26. Yang, X.-S. (2012). Flower Pollination Algorithm for Global Optimization. Unconventional Computation and Natural Computation, 240–249. https://doi.org/10.1007/978-3-642-32894-7_27
27. Gomes, G. F., da Cunha, S. S., Ancelotti, A. C. (2018). A sunflower optimization (SFO) algorithm applied to damage identification on laminated composite plates. Engineering with Computers, 35 (2), 619–626. https://doi.org/10.1007/s00366-018-0620-8
28. Mehrabian, A. R., Lucas, C. (2006). A novel numerical optimization algorithm inspired from weed colonization. Ecological Informatics, 1 (4), 355–366. https://doi.org/10.1016/j.ecoinf.2006.07.003
29. Qi, X., Zhu, Y., Chen, H., Zhang, D., Niu, B. (2013). An Idea Based on Plant Root Growth for Numerical Optimization. Intelligent Computing Theories and Technology. Berlin: Heidelberg, 571–578. https://doi.org/10.1007/978-3-642-39482-9_66
30. Bezuhlyi, V., Oliynyk, V., Romanenko, I., Zhuk, O., Kuzavkov, V., Borysov, O. et al. (2021). Development of object state estimation method in intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 5 (3 (113)), 54–64. https://doi.org/10.15587/1729-4061.2021.239854
31. Mahdi, Q. A., Shyshatskyi, A., Prokopenko, Y., Ivakhnenko, T., Kupriyenko, D., Golian, V. et al. (2021). Development of estimation and forecasting method in intelligent decision support systems. Eastern-European Journal of Enterprise Technologies, 3 (9 (111)), 51–62. https://doi.org/10.15587/1729-4061.2021.232718
32. Sova, O., Radzivilov, H., Shyshatskyi, A., Shevchenko, D., Molodetskyi, B., Stryhun, V. et al. (2022). Development of the method of increasing the efficiency of information transfer in the special purpose networks. Eastern-European Journal of Enterprise Technologies, 3 (4 (117)), 6–14. https://doi.org/10.15587/1729-4061.2022.259727
33. Zhang, H., Zhu, Y., Chen, H. (2013). Root growth model: a novel approach to numerical function optimization and simulation of plant root system. Soft Computing, 18 (3), 521–537. https://doi.org/10.1007/s00500-013-1073-z

34. Labbi, Y., Attous, D. B., Gabbar, H. A., Mahdad, B., Zidan, A. (2016). A new rooted tree optimization algorithm for economic dispatch with valve-point effect. International Journal of Electrical Power & Energy Systems, 79, 298–311. https://doi.org/10.1016/j.ijepes.2016.01.028
35. Murase, H. (2000). Finite element inverse analysis using a photosynthetic algorithm. Computers and Electronics in Agriculture, 29 (1-2), 115–123. https://doi.org/10.1016/s0168-1699(00)00139-3
36. Zhao, S., Zhang, T., Ma, S., Chen, M. (2022). Dandelion Optimizer: A nature-inspired metaheuristic algorithm for engineering applications. Engineering Applications of Artificial Intelligence, 114, 105075. https://doi.org/10.1016/j.engappai.2022.105075
37. Paliwal, N., Srivastava, L., Pandit, M. (2020). Application of grey wolf optimization algorithm for load frequency control in multi-source single area power system. Evolutionary Intelligence, 15 (1), 563–584. https://doi.org/10.1007/s12065-020-00530-5
38. Dorigo, M., Blum, C. (2005). Ant colony optimization theory: A survey. Theoretical Computer Science, 344 (2-3), 243–278. https://doi.org/10.1016/j.tcs.2005.05.020
39. Poli, R., Kennedy, J., Blackwell, T. (2007). Particle swarm optimization. Swarm Intelligence, 1 (1), 33–57. https://doi.org/10.1007/s11721-007-0002-0
40. Bansal, J. C., Sharma, H., Jadon, S. S., Clerc, M. (2014). Spider Monkey Optimization algorithm for numerical optimization. Memetic Computing, 6 (1), 31–47. https://doi.org/10.1007/s12293-013-0128-0
41. Yeromina, N., Kurban, V., Mykus, S., Peredrii, O., Voloshchenko, O., Kosenko, V. et al. (2021). The Creation of the Database for Mobile Robots Navigation under the Conditions of Flexible Change of Flight Assignment. International Journal of Emerging Technology and Advanced Engineering, 11 (5), 37–44. https://doi.org/10.46338/ijetae0521_05
42. Maccarone, A. D., Brzorad, J. N., Stone, H. M. (2008). Characteristics And Energetics of Great Egret and Snowy Egret Foraging Flights. Waterbird, 31 (4), 541–549. https://doi.org/10.1675/1524-4695-31.4.541
43. Ramaji, I. J., Memari, A. M. (2018). Interpretation of structural analytical models from the coordination view in building information models. Automation in Construction, 90, 117–133. https://doi.org/10.1016/j.autcon.2018.02.025
44. Pérez-González, C. J., Colebrook, M., Roda-García, J. L., Rosa-Remedios, C. B. (2019). Developing a data analytics platform to support decision making in emergency and security management. Expert Systems with Applications, 120, 167–184. https://doi.org/10.1016/j.eswa.2018.11.023
45. Chen, H. (2018). Evaluation of Personalized Service Level for Library Information Management Based on Fuzzy Analytic Hierarchy Process. Procedia Computer Science, 131, 952–958. https://doi.org/10.1016/j.procs.2018.04.233
46. Chan, H. K., Sun, X., Chung, S.-H. (2019). When should fuzzy analytic hierarchy process be used instead of analytic hierarchy process? Decision Support Systems, 125, 113114. https://doi.org/10.1016/j.dss.2019.113114

47. Osman, A. M. S. (2019). A novel big data analytics framework for smart cities. Future Generation Computer Systems, 91, 620–633. https://doi.org/10.1016/j.future.2018.06.046
48. Nechyporuk, O., Sova, O., Shyshatskyi, A., Kravchenko, S., Nalapko, O., Shknai, O. et al. (2023). Development of a method of complex analysis and multidimensional forecasting of the state of intelligence objects. Eastern-European Journal of Enterprise Technologies, 2 (4 (122)), 31–41. https://doi.org/10.15587/1729-4061.2023.276168
49. Merrikh-Bayat, F. (2015). The runner-root algorithm: A metaheuristic for solving unimodal and multimodal optimization problems inspired by runners and roots of plants in nature. Applied Soft Computing, 33, 292–303. https://doi.org/10.1016/j.asoc.2015.04.048
50. Poliarush, O., Krepych, S., Spivak, I. (2023). Hybrid approach for data filtering and machine learning inside content management system. Advanced Information Systems, 7 (4), 70–74. https://doi.org/10.20998/2522-9052.2023.4.09
51. Balochian, S., Baloochian, H. (2019). Social mimic optimization algorithm and engineering applications. Expert Systems with Applications, 134, 178–191. https://doi.org/10.1016/j.eswa.2019.05.035
52. Lenord Melvix, J. S. M. (2014). Greedy Politics Optimization: Metaheuristic inspired by political strategies adopted during state assembly elections. 2014 IEEE International Advance Computing Conference (IACC), 1157–1162. https://doi.org/10.1109/iadcc.2014.6779490
53. Moosavian, N., Roodsari, B. K. (2014). Soccer League Competition Algorithm, a New Method for Solving Systems of Nonlinear Equations. International Journal of Intelligence Science, 4 (1), 7–16. https://doi.org/10.4236/ijis.2014.41002
54. Hayyolalam, V., Pourhaji Kazem, A. A. (2020). Black Widow Optimization Algorithm: A novel meta-heuristic approach for solving engineering optimization problems. Engineering Applications of Artificial Intelligence, 87, 103249. https://doi.org/10.1016/j.engappai.2019.103249
55. Abualigah, L., Yousri, D., Abd Elaziz, M., Ewees, A. A., Al-qaness, M. A. A., Gandomi, A. H. (2021). Aquila Optimizer: A novel meta-heuristic optimization algorithm. Computers & Industrial Engineering, 157, 107250. https://doi.org/10.1016/j.cie.2021.107250
56. Hodlevskyi, M., Burlakov, G. (2023). Information technology of quality improvement planning of process subsets of the spice model. Advanced Information Systems, 7 (4), 52–59. https://doi.org/10.20998/2522-9052.2023.4.06
57. Askari, Q., Younas, I., Saeed, M. (2020). Political Optimizer: A novel socio-inspired meta-heuristic for global optimization. Knowledge-Based Systems, 195, 105709. https://doi.org/10.1016/j.knosys.2020.105709
58. Mohamed, A. W., Hadi, A. A., Mohamed, A. K. (2019). Gaining-sharing knowledge based algorithm for solving optimization problems: a novel nature-inspired algorithm. International Journal of Machine Learning and Cybernetics, 11 (7), 1501–1529. https://doi.org/10.1007/s13042-019-01053-x

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

Journal of Enterprise Technologies, 2 (4(122)), 23–30. https://doi.org/10.15587/1729-4061.2023.277608
70. Raskin, L., Sukhomlyn, L., Sokolov, D., Vlasenko, V. (2023). Multi-criteria evaluation of the multifactor stochastic systems effectiveness. Advanced Information Systems, 7 (2), 63–67. https://doi.org/10.20998/2522-9052.2023.2.09
71. Arora, S., Singh, S. (2018). Butterfly optimization algorithm: a novel approach for global optimization. Soft Computing, 23 (3), 715–734. https://doi.org/10.1007/s00500-018-3102-4
72. Mamoori, G. A., Sova, O., Zhuk, O., Repilo, I., Melnyk, B., Sus, S. et al. (2023). The development of solution search method using improved jumping frog algorithm. Eastern-European Journal of Enterprise Technologies, 4 (3 (124)), 45–53. https://doi.org/10.15587/1729-4061.2023.285292
73. Alieinykov, I., Thamer, K. A., Zhuravskyi, Y., Sova, O., Smirnova, N., Zhyvotovskyi, R. et al. (2019). Development of a method of fuzzy evaluation of information and analytical support of strategic management. Eastern-European Journal of Enterprise Technologies, 6 (2 (102)), 16–27. https://doi.org/10.15587/1729-4061.2019.184394
74. Shyshatskyi, A. V., Zhuk, O. V., Neronov, S. M., Protas, N. M., Kashkevych, S. O. (2024). Sukupnist metodyk pidvyshchennia operatyvnosti pryiniattia rishen z vykorystanniam metaevrystychnykh alhorytmiv. Modern Aspects of Science. Scientific Perspectives, 529–557. Available at: http://perspectives.pp.ua/public/site/mono/mono-40.pdf
75. Shyshatskyi, A. V., Matsyi, O. B., Yashchenok, V. Zh., Trotsko, O. O., Kashkevych, S. O. (2024). Sukupnist metodyk pidvyshchennia operatyvnosti pryiniattia rishen z vykorystanniam kombinovanykh metaevrystychnykh alhorytmiv. Modern Aspects of Science. Scientific Perspectives, 558–594. Available at: http://perspectives.pp.ua/public/site/mono/mono-40.pdf
76. Kashkevich, S., Dmytriieva, O., Trotsko, O., Shknai, O., Shyshatskyi, A. (2023). Mathematical model of information conflict of information networks. ScienceRise, 1, 3–13. https://doi.org/10.21303/2313-8416.2024.003395
77. Kashkevych, S. O., Dmytriieva, O. I., Yefymenko, O. V., Pliekhova, H. A., Shyshatskyi, A. V. (2024). Metody otsinky stanu skladnykh dynamichnykh obiektivz vykorystanniam bioinspirovanykh alhorytmiv. Modern Aspects of Science. Scientific Perspectives, 138–177. Available at: http://perspectives.pp.ua/public/site/mono/mono-40.pdf
78. Kashkevych, S. O., Dmytriieva, O. I., Trotsko, O. O., Shknai, O. V., Shyshatskyi, A. V. (2024). Metod samoorhanizatsii informatsiinykh merezh v umovakh destabilizuiuchykh vplyviv. The development of technical, agricultural and applied sciences as the main factor in improving life. Boston: Primedia eLaunch, 192–218. https://doi.org/10.46299/ISG.2024.MONO.TECH.2.7.2
79. Shyshatskyi, A., Dmytriieva, O., Lytvynenko, O., Borysov, I., Vakulenko, Y., Mukashev, T. et al. (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
80. Kashkevich, S., Litvinenko, O., Shyshatskyi, A., Salnyk, S., Velychko, V. (2024). The method of self-organization of information networks in the conditions of the complex influence

of destabilizing factors. Advanced Information Systems, 8 (3), 59–71. https://doi.org/10.20998/2522-9052.2024.3.07

81. Kashkevych, S. O., Dmytriieva, O. I., Plekhova, H. A., Protas, N. M., Neronov, S. M., Shyshatskyi, A. V. (2024). Naukovo-metodychnyi pidkhid z pidvyshchennia operatyvnosti obrobky riznotypnykh danykh z vykorystanniam metaevrystychnykh alhorytmiv. Modern Aspects of Science. Scientific Perspectives, 510–543. Available at: http://perspectives.pp.ua/public/site/mono/mono-40.pdf
82. Litvinenko, O., Kashkevich, S., Dmytriieva, O., Neronov, S., Plekhova, G.; Shyshatskyi, A. (Ed.) (2024). The method of self-organization of information networks in the conditions of destabilizing influences. Information and control systems: modelling and optimizations. Kharkiv: TECHNOLOGY CENTER PC, 3–34. https://doi.org/10.15587/978-617-8360-04-7.ch1
83. Kashkevich, S., Dmytriieva, O., Neronov, S., Plekhova, G., Zhyvylo, Y.; Shyshatskyi, A. (Ed.) (2024). The development of management methods based on bio-inspired algorithms. Information and control systems: modelling and optimizations. Kharkiv: TECHNOLOGY CENTER PC, 35–69. https://doi.org/10.15587/978-617-8360-04-7.ch2
84. Kashkevich, S., Plekhova, G., Kuchuk, N., Kuvshynov, O., Veretnov, A., Yefymenko, O.; Shyshatskyi, A. (Ed.) (2024). The development of methods for evaluating the state of complex technical systems using artificial intelligence theory. Information and control systems: modelling and optimizations. Kharkiv: TECHNOLOGY CENTER PC, 70–101. https://doi.org/10.15587/978-617-8360-04-7.ch3
85. Kashkevich, S., Plekhova, G., Yefymenko, O., Kuchuk, H., Davydov, V., Beketov, Y.; Shyshatskyi, A. (Ed.) (2024). The development of methods of learning artificial neural networks of intelligent decision-making support systems. Information and control systems: modelling and optimizations. Kharkiv: TECHNOLOGY CENTER PC, 102–137. https://doi.org/10.15587/978-617-8360-04-7.ch4
86. Kashkevich, S., Dmytriiev, I., Shevchenko, I., Lytvynenko, O., Shabanova-Kushnarenko, L., Apenko, N; Shyshatskyi, A. (Ed.) (2024). Scientific-method apparatus for improving the efficiency of information processing using artificial intelligence. Information and control systems: modelling and optimizations. Kharkiv: TECHNOLOGY CENTER PC, 138–167. https://doi.org/10.15587/978-617-8360-04-7.ch5