

NEUROMODELING OF OPERATIONAL PROCESSES

Subbotin S. A. – Dr. Sc., Professor, Head of the Department of Software Tools, National University “Zaporizhzhia Polytechnic”, Zaporizhzhia, Ukraine.

Pukhalska H. V. – PhD, Associate Professor, Associate Professor of the Department of Machinery Engineering Technology, National University “Zaporizhzhia Polytechnic”, Zaporizhzhia, Ukraine.

Leoshchenko S. D. – Post-graduate student of the Department of Software Tools, National University “Zaporizhzhia Polytechnic”, Zaporizhzhia, Ukraine.

Oliinyk A. O. – Dr. Sc., Associate Professor, Professor of the Department of Software Tools, National University “Zaporizhzhia Polytechnic”, Zaporizhzhia, Ukraine.

Gofman Ye. O. – PhD, Senior Researcher of the Research Unit, National University “Zaporizhzhia Polytechnic”, Zaporizhzhia, Ukraine.

ABSTRACT

Context. The problem of synthesis a neural network model of operational processes with the determination of the optimal topology, which is characterized by a high level of logical transparency and acceptable accuracy, is considered. The object of the study is the process of neural network modeling of operational processes using an indicator system to simplify the selection of the topology of neuromodels.

Objective of the work is to synthesis a neural network model of operational processes with a high level of logical transparency and acceptable accuracy based on the use of an indicator system.

Method. It is proposed to use a system of indicators to determine the topological features of ANN, which is the basis for modeling operational processes. The assessment of the level of complexity of the task obtained on the basis of information about the input data and the values of the criteria for assessing the specificity of the task allows to categorize the task to one of the types of complexity in order to determine the approach to the synthesis of a neuromodel. Complexity category OS allows, based on analytical data about the selection of input data, to obtain the exact number of neurons in the hidden layer for the synthesis of a neuromodel with a high level of logical transparency, which significantly expands their practical use and reduces the cost of subsequent operational processes.

Results. The obtained neuromodels of operational processes based on historical data. The use of the indicator system made it possible to significantly increase the level of logical transparency of the models, while maintaining high accuracy. Synthesized neuromodels reduce the resource intensity of operational processes by increasing the level of previous modeling.

Conclusions. The conducted experiments confirmed the operability of the proposed mathematical software and allow to recommend it for use in practice when modeling operational processes. The prospects for further research may consist in the use of more complex methods of feature selection to fix the group relationships of information features for the construction of more complex models.

KEYWORDS: modeling, operational processes, indicator system, neuromodel, sampling, training, error.

ABBREVIATIONS

ANN is an artificial neural net;
CTS is complex technical system;
IoT is Internet of Things;
MT is maintenance of technical system;
OC is organized complexity;
OS is organized simplicity;
NF is natural frequency.

NOMENCLATURE

J is informative weight of independent attribute;
 n is a number of input features that characterize sample instances;
 N_i is a multiple neurons at the network input;
 N_{i_j} is a neuron at the network input;
 N_o is a multiple neurons at the network output;
 N_{o_p} is a neuron at the network output;
 N_h is a multiple neurons of the hidden network layer;
 N_{h_r} is a hidden network layer neuron;

$Num_{elemtype}$ is a number of element types in the neural network4
 NN is a neural network;
 NN_{struct} is a structure of neural network;
 l is a number of neurons at the network input;
 $Lev_{accmeas}$ is a measurement accuracy level;
 Lev_{fctr} is a level of significant and less significant and/or non-significant factors4
 Lev_{manag} is a level of possible control and management;
 Lev_{task} is a conditional difficulty level of the task;
 $Lev_{smpifctm}$ is a level of possible simplification of the structure;
 m is a number of dependent (categorical) features of sample instances;
 p is a number of neurons at the network output;
 q is a number of connections between neurons in the network;
 r is number of neurons in the hidden network layer;

Sample is a data set;

w is a multiple of connections between neurons;

w_q is a connection between neurons in the network;

x_n is an independent attribute of the sample instance;

X is a set of independent attribute (variables);

y_m is a value of the dependent variable (attribute) of the sample instance;

Y is a set of values of dependent variables.

INTRODUCTION

The quality of modern CTS is largely determined by their reliability. One of the most significant factors of reliability changes is the operating conditions and the adopted operation strategy, which should be understood as a set of organizational and technical measures for maintenance and restoration of serviceability or operability of failed objects. The decrease in the intensity and volume of such measures negatively affects the reliability indicators of the CTS. To the greatest extent, this trend is characteristic of systems that operate autonomously, without the possibility of carrying out preventive control and restoration (diagnostics and repair) measures or with the possibility of carrying them out in a reduced volume. Such features are inherent in road transport objects, marine objects and special-purpose systems [1–5].

During the operation of the CTS of this class, periodically, after a while t_{stat} , they leave stationary points (bases, airfields, airports, etc.) to perform tasks for their intended purpose during the time t_{work} . In this case, the object can be in conditions that ensure its immediate use with intensity λ_{usng} or it is in reserve (shelving). During this time, objects can be monitored only superficially, in a small volume (control inspection, state check, etc.) with intensity $\lambda_{contusng}$ and duration $t_{contusng}$. During inspections and, possibly, based on the results of continuous monitoring, failures that occur with intensity are detected λ_0 . Failures lead to an inoperable state before returning to a stationary point (base) where serviceability or working capacity is restored during the time t_{rep} . The intensity of the reserve demand is determined by product failures and the intensity of successful completion of tasks for the intended purpose, i.e. $\lambda_{usng} + \lambda_0$ [1, 3].

During CTS locate in a stationary point (base, airport) at the facility, in addition to restoring serviceability, MT is carried out with frequency t_{PTM} and duration t_{TM} . MT is also carried out after making n_{TM} flights (exits) (when the number $n_{TM} > 2$). Meanwhile MT, especially in the case of a long stay on the base, the object may also fail with the parameter ω_0 . The resulting failures are eliminated during the next maintenance or during checks before departure for the flight. The duration of these checks t_{PTM} ($t_{PTM} > t_{TM}$) has a significant impact on the failure rate when performing tasks offline outside of stationary points, i.e. $\lambda_0 = f(t_{PTM})$ and $\lambda_0 \geq \omega_0$ more-

over, equality is achieved when $t_{PTM} = t_{TM}$. In the simplest case $\lambda_0 = \omega_0 \cdot \frac{t_{TM}}{t_{PTM}}$ [1–4].

During modeling, it is convenient to represent the considered process of CTS operation as a random process in a discrete phase space. The phase space of the process includes two states (the first MT and the second MT), as well as indicators of natural oscillations. Taking into account the availability of historical experimental data, an ANN will be used as the basis for the model [1–5].

ANN are statistical computational models applied to a variety of practical tasks, including diagnostics (technical and medical based on multimedia data about an object), assessment, forecasting, etc. [6, 7]. During process of supervised learning ANN trains on the example of already known data, that is, so it is exist a predefined correct answer for all the initial data. The main idea of training a neural network is to set up a configuration in which the model's responses will be as close as possible to the correct ones. However, at the moment there are many ANN topologies that can be used as a neuromodel. So some can provide a context for each subsequent prediction (recurrent ANN). This helps the ANN to maintain the state in which the decision was made. Therefore, it is so important at the initial stage to correctly assess the complexity of the problem for further selection of the ANN topology and the choice of the training approach for the synthesis of the most optimal model [6, 7].

The task of studying the process is to obtain a model that will reflect the behavior inherent in the source data. Such a task can be attributed to the number of template recognition tasks. Regarding the event log as training data, we will be trained to evaluate the results for each event in the ANN log. The ultimate goal will be to synthesize a model based on the ANN representing a neuromodel of the operational process encoded in the event log

The object of study is the process of synthesis neuromodels of operational processes with a high level of interpretability and acceptable accuracy of operation.

Using the assessment of the complexity level, it is possible at the initial stage to determine the approach to the synthesis of the model based on the ANN.

The subject of the study is a neural network model of operational processes, characterized by a high level of interpretability and acceptable accuracy.

Using the information about the modeling task and the evaluation of the input data, it is necessary to synthesize a neuromodel.

The purpose of the work is to build and study neuromodels of operational processes with the previous definition of structural features based on the assessment of the level of complexity of the task.

1 PROBLEM STATEMENT

The operational process can be represented as a modeling problem. Where, at the initial stage, a set of various characteristics (features) of the operation of the object (system) that is being studied is available [8, 9]. A set of

characteristics is represented by a set of conditionally independent features $X = \{x_1, x_2, \dots, x_n\}$ of an object consisting of the n number of such. As a rule, such characteristics are the values of the results of the operation of the object measured using special sensors, sensor systems or devices [6–9].

In accordance with these independent features, a set of values of the dependent characteristics of the object is compared: $Y = \{y_1, y_2, \dots, y_m\}$ consisting of m elements. It is assumed that to some extent each independent feature x_k affects the value of the corresponding one y_l . The degree of this influence can be represented as the information weight of an independent attribute [6–9].

Then the neuromodel of the operational process can be represented as an ANN $NN = (struct, param)$. The structure of such a neuromodel is determined by sets of computational nodes: neurons and connections between them: $struct = \{N, c\}$, $N = \{N_i, N_h, N_o\}$, $c = \{c\}$. In turn, the aggregates of the set of neurons are divided into subsets by layers: the neurons of the input layer $N_i = \{N_{i_1}, N_{i_2}, \dots, N_{i_l}\}$, $l = 1, 2, \dots, |N_i|$, the output layer $N_o = \{N_{o_1}, N_{o_2}, \dots, N_{o_p}\}$, $p = 1, 2, \dots, |N_o|$ and the hidden one $N_h = \{N_{h_1}, N_{h_2}, \dots, N_{h_r}\}$, $r = 1, 2, \dots, |N_h|$. The subset of links consists of the links themselves and their weighting coefficients: $c = \{c_1, c_2, \dots, c_k\}$, $k = 1, 2, \dots, |c|$, $w = \{w_k\}$.

Accordingly, the task can be represented as a synthesis of the ANN with optimal structure and accuracy $NN = (struct, param)$, based on a sample of initial data about the object under study during operation $Sample = \langle X, Y \rangle$.

2 REVIEW OF THE LITERATURE

The idea of ANN is to model (repeat) the behavior of various processes based on historical (experimental) information. The ANN itself is a set of special mathematical functions with many parameters that are configured in the process of learning from previous data. Then the trained ANN processes the initial real data and gives its forecast of the future behavior of the studied system. The essence of ANN is the desire to imitate the processes taking place. In its structure, the neural network is similar to the human brain and is also capable of learning [6–8].

The main difference between models based on ANNs and growth curves or regression methods is that if these methods adjust a real process or phenomenon to a standard mathematical function, then ANNs select the parameters of a system of equations, bringing it to real life [7].

Schematically, an artificial neural network consists of a layer of input signals, an output layer and several internal layers (Fig. 1).

The processes of building and training a network in a software package that supports the creation of neural net-

works are as follows: the values of input variables are fed to the input, the type of connection and weight coefficients are selected randomly, then the values of the output variable are calculated. The obtained values are compared with the real ones, after that, the weights and the type of network are adjusted, aimed at reducing the error. The general scheme is shown in Fig. 2 [6–9].

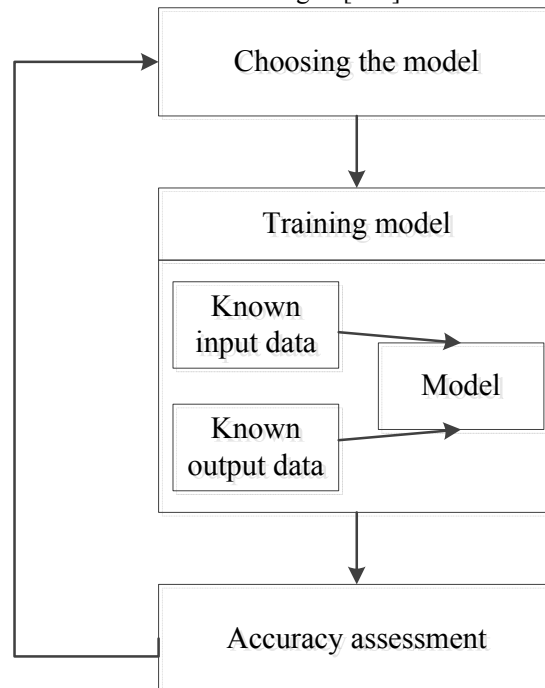


Figure 2 – General scheme of training process

An important issue is also the organization of the transmission of the same historical (experimental) data. Thus, with the correct organization of data transmission and storage processes, it is possible to organize a complex system within the IoT technology, which will be aimed at online diagnostics and MT the operability of the technical system [10–11]. Such an organization involves the installation of deployed sensor networks on technical elements and nodes. Sensor networks provide automated recording of operational indicators with a specified time lag, which can reach miles and microseconds. All recorded data is transferred to the cloud, where a data bank is formed. Wireless network data transmission technologies are used for transmission. An external computing server or several such servers have access to the accumulated data bank. These installations can perform real-time analysis of constantly updated data. Among the possible types of analysis, the following can be distinguished [10–11]:

- data verification: checking the truth and correctness of the received data (for example, filling in empty data or tracking unexpected run-ups in indicators);
- statistical data analysis: identifying and visualizing the simplest patterns and relationships between data. Data normalization and standardization can also be included here;
- data reduction: reducing the dimension of data is sometimes necessary to optimize resource consumption in

data analysis. Thus, the selection of informative features, preliminary data mining and other operations help to speed up the process of further processing (for example, the synthesis of models based on data), and sometimes also improve accuracy (by removing noisy data).

Also, an important task of the server in such systems of the synthesis, updating (additional training) or modernization of new or pre-built models [10–11].

Such models, based on constantly updated data, can more accurately diagnose, monitor, or make a forecast. All the results generated by the models are redirected to the workstation (this can be done via the cloud or directly), since in some systems the results obtained either have to pass moderation, or may require the involvement of the operator.

Thus, for the most part, such an organization of IoT systems is aimed at maintaining the operability of technical systems: diagnostics, non-destructive control or forecasting of operability. The general principle is shown in Fig. 3.

However, the organization of an IoT system or simpler solutions is associated with an assessment of the complexity of the simulated problem for choosing a model synthesis strategy based on the ANN and choosing the appropriate topology.

3 MATERIALS AND METHODS

As it was given in the previous section, the modeling task can be unified for a specific task after a certain comprehensive assessment of its complexity. Given that the structure of ANN ($NN = (struct, param)$) allows to most subtly encode the relationships between the input data ($X = \{x_1, x_2, \dots, x_n\}$), it is necessary to accurately select the synthesis option for such a non-network model. Based on the values of the indicators to assess the complexity of the task it can be choose a way $Lev_{Task} = \{Inf_{sample}, Lev_{smplfctm}, Lev_{fctr}, Lev_{accmeas}, Lev_{manag}\}$ to synthesize the most acceptable structure [12].

In the case when a problem with input data $Sample = \langle X, Y \rangle$ that is questionable can be modeled (there is a question about the accuracy of the data, their excess or a high degree of interconnectedness), it is necessary to resort to input data preprocessing. Thus, the selection of informative features will allow to exclude uninformative features $Sample^d = \langle X^d, Y \rangle$, which will subsequently increase the level of logical transparency of the neuromodel. By spending more time on data preprocessing, it is possible to significantly reduce the time resources at the stage of model synthesis based on the ANN [12].

Stepwise regression methods can be used to feature selection. Stepwise regression is a method that iteratively checks the statistical significance of each independent variable in a linear regression model [8, 9]. This is done through iteration, that is, the process of obtaining results or solutions by repeating rounds or cycles of analysis.

Automatic testing with the help of statistical software packages allows you to save time and reduce the number of errors. A bidirectional exception is a combination of forward and reverse exclusion methods that check which variables should be included or excluded [8, 9].

Firstly, it must be sorted $x^- = \arg \max J(X_n - x)$, where $x \in X_n$. After that it must be update $X_{n-1} = X_n - x^-$. Finally, internal iteration calculation must be update.

Such manipulation on the first step guarantees that it will be removed a feature, x^- from our feature subset X_n . Moreover, x^- is the feature that maximizes our criterion function upon removal, that is, the feature that is associated with the best classifier performance if it is removed from X_n .

Secondly, pull must be update based on the rule $x^+ = \arg \max J(X_n + x)$, where $x \in Y - X_n$ with special condition:

$$J(X_n + x) > J(X_n), \quad (1)$$

so in this case have: $X_{n+1} = X_n + x^+$. And again internal iteration calculation must be update.

Second manipulation search for features that improve the classifier performance if they are added back to the feature subset. If such features exist, we add the feature x^+ for which the performance improvement is maximized. If internal iteration calculation came to the 2 or an improvement cannot be made (i.e., such feature x^+ cannot be found), go back to exclusion manipulation; else, repeat the adding.

However, after the selection of features, the problem can be considered already in the category of OS, when a simple direct propagation ANN is sufficient for modeling, and the number of neurons in the hidden layer is calculated based on the statistical characteristics of the data sample [12]:

$$Lev_{Task} \{ \dots, Lev_{accmeas} = 1 \} \rightarrow OC$$

$$Lev_{Task} \xrightarrow{Feature\ Selection} \{ \dots, Lev_{accmeas} = 0 \} \rightarrow OS. \quad (2)$$

4 EXPERIMENTS

The blades of the single stage compressor engine TV3-117, made of alloy BT8 and having operational damage to the feather of the engine blades, were selected as the object of research. The studies were carried out on two engines operated under the same conditions, but having different operating hours and, accordingly, different degrees of damage to the blades. Engine D1 have 1971 h and D2: 990 h. Operational damage to the pen creates not only a stress concentration, but also leads to a change in the geometry of the blades. For research, 20 blades with

no gross mechanical damage were selected from two engines [13–16].

The study of the geometry of the blades consisted in measuring the chord, C_1 and C_2 in sections from A2–A2 to A8–A8. The measurement results indicate that the greatest change in the geometry of the blade parameters occurs in the peripheral zone (sections A7–A7 and A8–A8) [13–16].

The table shows that $x_1, x_4, x_7, x_{10}, x_{13}, x_{16}, x_{19}$: B , the value of the chord, in Table 1 in different sections; $x_2, x_5, x_8, x_{11}, x_{14}, x_{17}, x_{20}$: C_1 , the thickness of the input edge; $x_3, x_6, x_9, x_{12}, x_{15}, x_{18}, x_{21}$: C_2 , the thickness of the output edge; x_{22} : HB, the hardness of the initial blade, HRC.

x_{23} : $\sigma_{0,2}$, yield strength of the starting material, MPa; x_{24} : σ_B tensile strength, MPa; y_1 : T_1 total operating time; y_2 : T_2 operating time up to first repair, h; y_3 is the frequency of natural vibrations of the blades, Hz.

5 RESULTS

The table 1 shows the part of the sample that was used for the experiments.

Table 2 presents regression models for different engines and their aggregates. The models are based on a reduced number of input features.

Table 3 presents ANN-based models in matrix form.

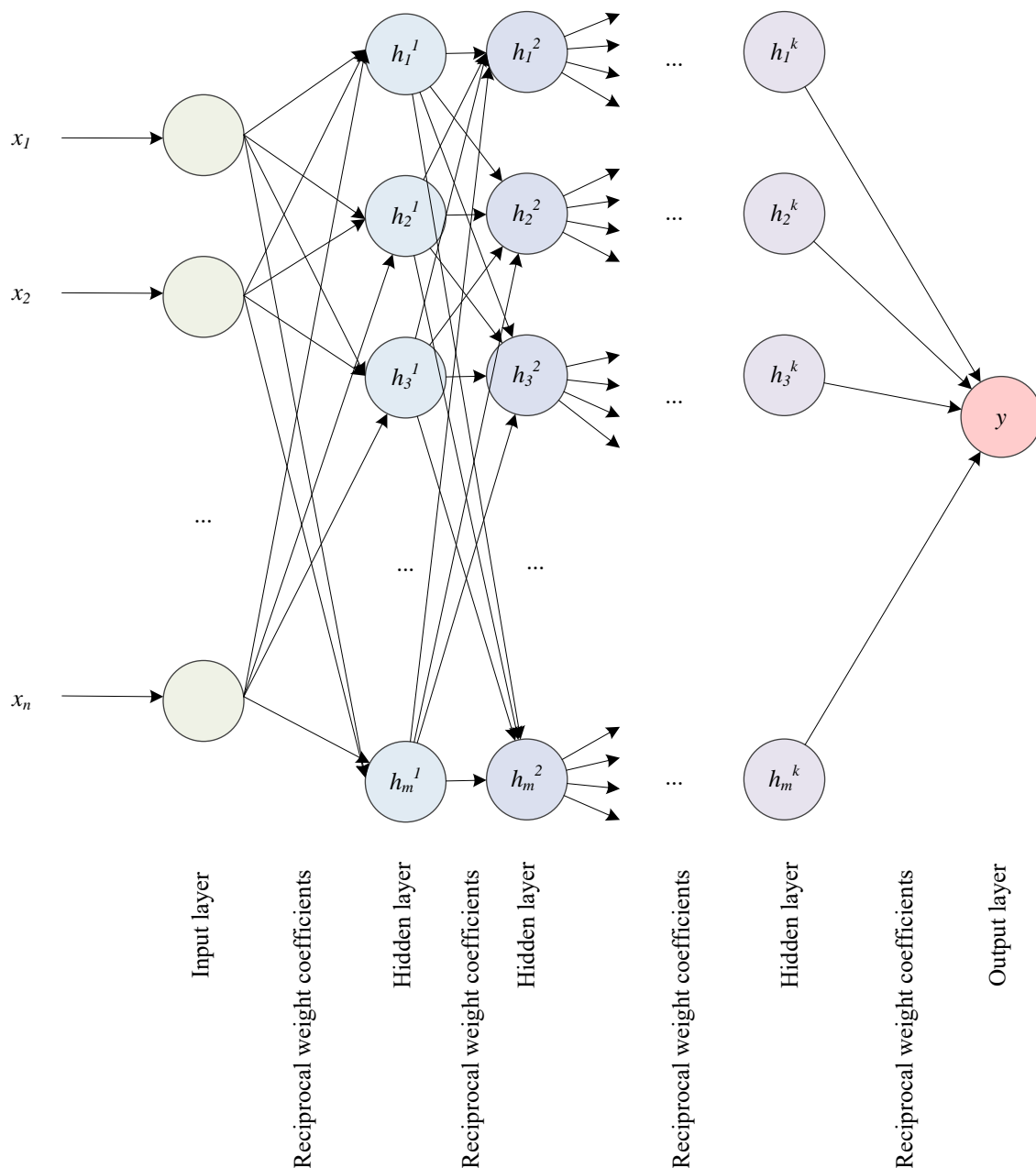


Figure 1 – General scheme of simple topology of ANN

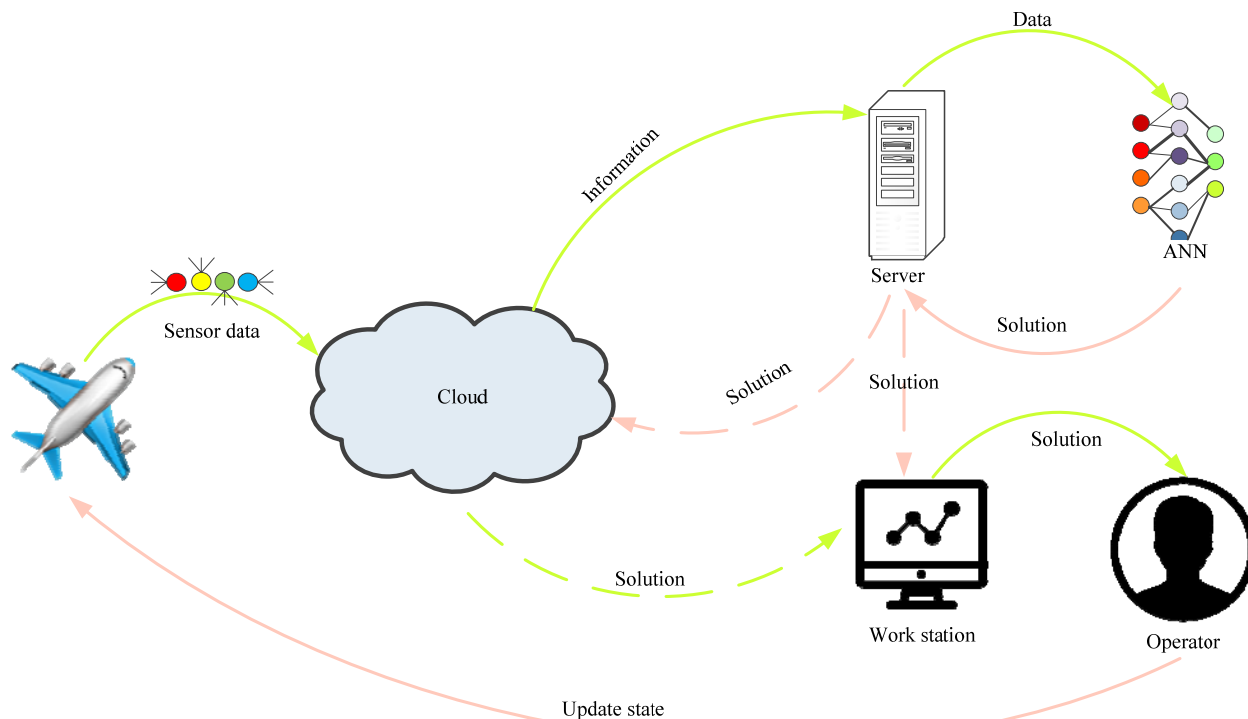


Figure 3 – General scheme of interaction in IoT systems for operational process

Table 1 – General information about data set

Blade A2–A2			Blade A3–A3			Blade A4–A4			...	T ₁	T ₂	NF
B ₂₋₂	C1 ₂₋₂	C2 ₂₋₂	B ₂₋₂	C1 ₂₋₂	C2 ₂₋₂	B ₄₋₄	C1 ₄₋₄	C2 ₄₋₄	...	y ₁	y ₂	y ₃
x ₁	x ₂	x ₃	x ₄	x ₅	x ₆	x ₇	x ₈	x ₉	...	y ₁	y ₂	y ₃
26.50	1.33	0.64	27.00	0.00	0.00	590.00	0.97	0.48	...	0.00	0.00	590.00
27.10	1.35	0.66	27.60	0.00	0.00	620.00	0.99	0.50	...	0.00	0.00	620.00
25.90	1.31	0.62	26.40	0.00	0.00	650.00	0.95	0.46	...	0.00	0.00	650.00
26.53	1.39	0.54	26.91	1971	621	647.3	0.89	0.42	...	1971	621	647.3
26.60	1.37	0.68	26.77	1971	621	646.7	0.90	0.44	...	1971	621	646.7
26.44	1.53	0.60	26.83	1971	621	642.4	0.98	0.40	...	1971	621	642.4
26.52	1.25	0.62	26.92	1971	621	643.1	0.81	0.45	...	1971	621	643.1
26.72	1.35	0.57	27.16	1971	621	648.5	0.91	0.38	...	1971	621	648.5
26.70	1.57	0.70	27.07	1971	621	668.6	0.99	0.45	...	1971	621	668.6
26.25	1.46	0.62	26.65	1971	621	645.2	0.96	0.44	...	1971	621	645.2
26.20	1.42	0.58	26.60	1971	621	649.4	0.97	0.50	...	1971	621	649.4
26.51	1.42	0.65	26.85	1971	621	648.8	0.93	0.44	...	1971	621	648.8
...
26.63	1.20	0.59	27.01	1.15	0.44	27.86	0.95	0.35	...	990	451	627.4

Table 2 – Regression models for engines after feature selection

	Depended attribute	Independent feature	Weight	Depended attribute	Independent feature	Weight	Depended attribute	Independent feature	Weight
First engine	y ₁	Free member	5493.3	y ₂	Free member	1730.8	y ₃	Free member	1015.7
		x ₃	787.81		x ₃	248.21		x ₆	71.182
		x ₁₃	-174.09		x ₁₃	-54.85		x ₁₆	-0.85927
		x ₁₅	-15979		x ₁₅	-5034.5		x ₁₇	46.001
		x ₁₆	907.93		x ₁₆	286.06		x ₁₈	-160.38
		x ₁₇	-3533.5		x ₁₇	-1113.3		x ₁₉	-11.27
		x ₁₈	19309		x ₁₈	6083.6		x ₂₀	-161.8
		x ₁₉	-735		x ₁₉	-231.58		x ₂₁	70.668
	x ₂₀	-4496.7	x ₂₀	-1416.8					
	x ₂₁	-5967	x ₂₁	-1880					

Table 2 continued

Second engine	y ₁	Free member	-6329,7	y ₂	Free member	-2883,5	y ₃	Free member	989,16
		x ₅	104.18		x ₅	47.46		x ₄	-3.2178
		x ₁₆	577.55		x ₁₆	263.1		x ₁₆	-17.851
		x ₁₇	-887.97		x ₁₇	-404.52		x ₁₇	-57.823
		x ₁₈	-6417		x ₁₈	-2923.3		x ₁₈	-132.7
		x ₁₉	-194.56		x ₁₉	-88.633		x ₁₉	10.239
		x ₂₀	426.08		x ₂₀	194.1		x ₂₀	24.261
x ₂₁	-8265.3	x ₂₁	-3765.3	x ₂₁	-22.105				
General sample for two engines	y ₁	Free member	15976	y ₂	Free member	6146	y ₃	Free member	1236.9
		x ₁₃	111.95		x ₁₃	39.731		x ₁₃	-31.607
		x ₁₄	1444.9		x ₁₄	-31.269		x ₁₄	58.769
		x ₁₅	-2869.8		x ₁₅	-1359.9		x ₁₅	-66.747
		x ₁₆	213.64		x ₁₆	38.801		x ₁₆	17.876
		x ₁₇	-3010.7		x ₁₇	-472.48		x ₁₇	-64.156
		x ₁₈	3429		x ₁₈	788.81		x ₁₈	47.107
		x ₁₉	-780.91		x ₁₉	-254.39		x ₁₉	-6.5416
		x ₂₀	-2449.6		x ₂₀	-411.03		x ₂₀	-79.377
		x ₂₁	-1397		x ₂₁	-532.82		x ₂₁	51.834

Table 3 – Neural networks models for engines in matrix format

		Number of layer	Number of neuron at layer	Number of input of neuron								
				0	1	2	3	4	5	6	7	8
First engine	y ₁	1	1	0.4658	-1.5179	1.2285	-0.3745	-1.3876	1.4247	0.8189	-0.7037	-0.0544
			2	1.1000	-0.2338	0.1421	0.5441	-0.5349	-1.2274	-0.3497	-2.2515	-2.1745
			3	-0.8452	0.9026	-1.2838	0.1328	0.7307	2.2389	0.7699	0.3432	-1.4166
			4	0.3374	5.4603	-2.0717	4.4364	-9.3008	1.9159	-6.5536	-8.9370	-3.4884
	y ₂	1	1	1.5418	0.9468	1.8911	-0.8538	-1.6638	0.2718	-1.057	-3.2833	0.0146
			2	0.9662	0.1809	-1.1613	-0.6279	-1.7703	0.2904	-1.172	-1.3198	0.9192
			3	0.968	0.2189	1.5918	1.7624	-2.655	1.5395	-2.8073	-2.7069	-0.2501
			4	0.2781	0.2975	1.2749	0.8646	0.6666	0.9116	-0.7911	0.5896	-0.4745
	y ₃	1	1	0.9422	1.8965	-0.8817	-1.287	1.7724	1.622	0.9202		
			2	-2.947	-4.7925	2.8078	2.4996	-4.2739	-2.8727	-2.5115		
			3	-1.199	0.6908	-1.5036	-1.2848	-0.8571	-0.6933	1.133		
			4	-0.4857	-0.5768	-0.2348	-0.1187	-0.4004	1.2633	-0.3987		
Second engine	y ₁	1	1	0.4939	-0.4511	1.4086	0.1413	7.6385	-0.9269	-3.1648		
			2	-2.6959	-7.3992	-2.4984	6.6421	-0.9476	4.9028	-2.6633		
			3	0.2274	-0.7406	1.3095	0.7900	-0.4079	-0.6326	4.0847		
			4	0.9888	-5.0414	-0.1517	5.6668	1.8127	-2.8537	2.3601		
	y ₂	1	1	-0.0252	-1.239	-0.593	-0.4079	-0.2391	-0.2076	0.5937		
			2	-0.7033	-3.6831	0.6049	3.3973	1.3224	0.1675	5.3631		
			3	-1.0152	-0.1636	1.4365	1.9574	2.3268	-0.3212	2.9033		
			4	0.2268	1.4866	-0.3067	0.1771	0.9589	-0.5183	0.7572		
	y ₃	1	1	1.0348	0.9792	-3.1372	0.9754	-2.5656	2.457	-1.0045		
			2	5.2574	-6.495	-0.4098	3.5421	1.8056	-3.8595	-1.2348		
			3	-5.4322	-2.435	2.1287	-1.3974	-0.585	0.6515	5.4371		
			4	-1.0328	5.7223	1.4171	1.3843	-1.9217	0.2199	-2.1831		
General sample for two engines	y ₁	1	1	1.5137	-1.5685	-2.5358	-0.7641	8.9924	-6.6683	0.5498	9.3855	-2.607
			2	-12.005	1.4226	0.1941	-1.0293	36.1094	-11.951	0.5653	16.5209	8.7779
			3	17.6653	0.7479	-17.545	-12.728	-6.4936	-7.7768	-8.5848	11.3728	-3.6221
			4	-8.5863	5.8129	4.6873	-5.543	2.7865	11.7949	-17.555	-12.495	1.3517
	y ₂	1	1	-4.3625	-6.5875	-1.5977	1.2776					
			2	-1.1947	1.8355	-1.0277	-0.9955	-2.4798	3.4032	-1.4717	-4.0901	4.664
			3	1.3742	-1.8717	-0.0821	-0.1832	4.4903	-2.391	1.5867	6.7386	-2.3275
			4	2.1382	-0.0421	-2.9346	1.556	0.0176	-3.4581	-0.6713	0.0464	-4.587
	y ₃	1	1	1.7524	-1.8059	-1.381	1.121	1.8792	-2.9072	-0.2687	1.6761	-4.4536
			2	-0.6609	1.7264	1.9347	2.0265					
			3	4.1667	0.7699	-0.0491	0.5137	3.3659	-1.0307	0.9681	-2.0161	1.0408
			4	0.5733	1.4045	-0.3144	3.7791	0.2089	-4.0823	2.4539	0.9196	2.6012
y ₃	1	1	-0.1248	0.2645	0.5078	-1.5247	-0.127	1.0278	-1.0693	-0.6277	-0.783	
		2	2.8772	3.0926	0.5896	-4.3669	0.689	1.8098	-3.1684	-2.1971	0.1409	
		3	-1.3179	3.2033	-0.7647	1.5967						
		4	-1.3179	3.2033	-0.7647	1.5967						

6 DISCUSSION

At first, the task for modeling was assigned to the OC category. However, after preprocessing the input data, information-important features were selected. Accordingly, after data reduction, the task was transferred to the OS group. In the end, the input sample was not excessive, and the risks of human influence were excluded. The only significant complicating factor is poorly conditioned correlation matrices.

Further calculations showed that the use of 6–8 neurons in the hidden layer is sufficient to build a neural model with acceptable accuracy.

Analyzing the initial results, we should note a fairly large run-up in the model training time: from 4 seconds (the largest indicator among the ANN-based models) to 34.37 for linear regression models.

The results obtained on the data after the reduction showed that the accuracy increased when constructing a neural model with a certain structure based on a system of indicators, and the time was significantly reduced.

In addition, it should be noted the high level of logical transparency of the obtained models based on the ANN.

CONCLUSIONS

The urgent scientific and applied problem of determining the optimal and logically transparent structure of a neuromodel for modeling the operational processes is solved.

The scientific novelty lies in the fact that using the feature selection methods for pre-processing of input data allows to re-define the level of task complexity and use more resource-efficient methods for synthesis model based on ANN. Such models have the optimal, logically transparent topology and high level of accuracy.

The practical significance lies in the fact that such approaches that were used allow speed up the process of synthesis 8.6 times. Moreover, such models based on ANN demonstrate better accuracy average by 6%.

Prospects for further research are using additional information as input data set for tracking implicit factors on operational processes. In this case using more complex topologies of ANN with modern methods for training can demonstrate good results.

ACKNOWLEDGEMENTS

The work was carried out with the support of the state budget research projects of the state budget of the National University “Zaporozhzhia Polytechnic” “Intelligent methods and software for diagnostics and non-destructive quality control of military and civilian applications” (state registration number 0119U100360) and “Development of methods and tools for analysis and prediction of dynamic behavior of nonlinear objects”.

REFERENCES

1. Haneveld W. K. K., van der Vlerk M. H., Romeijnders W. Stochastic Programming: Modeling Decision Problems Un-

- der Uncertainty. Berlin, Springer, 2020, 264 p. DOI: 10.1007/978-3-030-29219-5
2. Ceder A. Public Transit Planning and Operation: Modeling, Practice and Behavior. London, Chapman and Hall (CRC Press), 2019, 742 p. DOI: 10.3846/16484142.2015.1114968
3. Mu E., Rojas M. P. Practical Decision Making: An Introduction to the Analytic Hierarchy Process (AHP) Using Super Decisions V2 (SpringerBriefs in Operations Research). Berlin, Springer, 2017, 124 p. DOI: 10.1007/978-3-319-33861-3
4. Antoniou C., Dimitriou L., Pereira F. Mobility Patterns, Big Data and Transport Analytics, Tools and Applications for Modeling. Amsterdam, Elsevier, 2018, 452 p.
5. Skormin V. A. Introduction to Process Control: Analysis, Mathematical Modeling, Control and Optimization (Springer Texts in Business and Economics). Berlin, Springer, 2020, 264 p. DOI: 10.1007/978-3-319-42258-9
6. Loy J. Neural Network Projects with Python: The ultimate guide to using Python to explore the true power of neural networks through six projects. Birmingham, Packt Publishing, 2019, 308 p.
7. Aggarwal C. C. Neural Networks and Deep Learning: A Textbook. Berlin: Springer, 2020, 264 p. DOI: 10.1007/978-3-319-94463-0
8. Brunton S. L. Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control. Cambridge: Cambridge University Press, 2019, 492 p. DOI: 10.1017/9781108380690
9. F. Neitzel Stochastic Models for Geodesy and Geoinformation Science. Basel, Multidisciplinary Digital Publishing Institute, 2021, 200 p. DOI: 10.3390/books978-3-03943-982-9
10. Gama J., Pashami S., Bifet A. et al. IoT Streams for Data-Driven Predictive Maintenance and IoT, Edge, and Mobile for Embedded Machine Learning (Communications in Computer and Information Science). Berlin: Springer, 2021, 328 p. DOI: 10.1007/978-3-030-66770-2
11. Mathur P. IoT Machine Learning Applications in Telecom, Energy, and Agriculture, With Raspberry Pi and Arduino Using Python. Berlin, Apress, 2020, 264 p. DOI: 10.1007/978-1-4842-5549-0
12. Leoshchenko S. D., Subbotin S. A., Oliinyk A. O., Nariv's'kiy O. E. Implementation of the indicator system in modeling complex technical systems. *Radio electronics, computer science, control*, 2021, No. 1, pp. 117–127. DOI: 10.15588/1607-3274-2021-1-12.
13. Babenko O., Pribora T. Analysis of the results of the study of the frequencies and forms of natural vibrations of the working blade of the 1st stage of the SLP, *Bulletin of Engine Building*, 2018, Vol. 2, pp. 91–98. [In Russian]
14. Dvirnyk Ya., Pavlenko D. The influence of dust erosion on the gas dynamic characteristics of the axial compressor of the GTE Vestnik dvigatelstroeniya, *Bulletin of Engine Building*, 2017, Vol. 1, pp. 56–66. [In Russian]
15. Yefanov V., Prokopenko O., Ovchinnikov O., Vnukov U. Erosion resistance of compressor blades of helicopter gas turbine engines with various types of coatings, *Bulletin of Engine Building*, 2017, Vol. 1, pp. 120–123. [In Russian]
16. Dvirnyk Ya., Pavlenko D. Patterns of wear of the compressor blades of helicopter engines operating in a dusty atmosphere, *Bulletin of Engine Building*, 2016, Vol. 1, pp. 42–51. [In Russian]

Received 25.08.2021.

Accepted 20.10.2021.

НЕЙРОМОДЕЛЮВАННЯ ЕКСПЛУАТАЦІЙНИХ ПРОЦЕСІВ

Субботін С. О. – д-р техн. наук, професор, завідувач кафедри програмних засобів Національного університету «Запорізька політехніка», Запоріжжя, Україна.

Пухальська Г.В. – канд. техн. наук, доцент кафедри технологія машинобудування Національного університету «Запорізька політехніка», Запоріжжя, Україна.

Леощенко С. Д. – аспірант кафедри програмних засобів Національного університету «Запорізька політехніка», Запоріжжя Україна.

Олійник А. О. – д-р техн. наук, професор кафедри програмних засобів Національного університету «Запорізька політехніка», Запоріжжя, Україна.

Гофман С. О. – канд. техн. наук, старший науковий співробітник науково-дослідної частини Національного університету «Запорізька політехніка», Запоріжжя, Україна.

АНОТАЦІЯ

Актуальність. Розглянуто завдання побудови нейромережевої моделі експлуатаційних процесів з визначенням оптимальної топології, яка відрізняється високим рівнем логічної прозорості та прийнятною точністю. Об'єктом дослідження є процес нейромережевого моделювання експлуатаційних процесів із застосуванням індикаторної системи для спрощення вибору топології нейромоделей.

Мета роботи полягає в побудові нейромережевої моделі експлуатаційних процесів з високим рівнем логічної прозорості і прийнятною точністю на основі використання індикаторної системи.

Метод. Запропоновано використовувати систему індикаторів для визначення топологічних особливостей штучних нейронних мереж, які є базисом для моделювання експлуатаційних процесів. Оцінка рівня складності задачі отримана на основі інформації про вхідні дані та значень критеріїв оцінки специфічності задачі дозволяє категоризувати задачу до одного з видів складності, щоб визначити підхід до синтезу нейромоделі. Категорія складності організована простота дозволяє на основі аналітичних даних про вибірку вхідних даних отримати точну кількість нейронів в прихованому шарі для синтезу нейромоделі з високим рівнем логічної прозорості, що значно розширює їх практичне використання і знижує вартість наступних експлуатаційних процесів.

Результати. Отримані нейромоделі експлуатаційних процесів на основі історичних даних. Використання системи індикаторів дозволило в значній мірі збільшити рівень логічної прозорості моделей, зберігаючи високу точність. Синтезовані нейромоделі знижують ресурсоємність промислових процесів за рахунок збільшення рівня попереднього моделювання.

Висновки. Проведені експерименти підтвердили працездатність запропонованого математичного забезпечення і дозволяють рекомендувати його для використання на практиці при моделюванні експлуатаційних процесів. Перспективи подальших досліджень можуть полягати у використанні більш складних методів відбору ознак для фіксації групових взаємозв'язків інформаційних ознак для побудови більш складних моделей.

КЛЮЧОВІ СЛОВА: моделювання, експлуатаційні процеси, система індикаторів, нейромоделі, вибірка, навчання, помилка.

НЕЙРОМОДЕЛИРОВАНИЕ ЭКСПЛУАТАЦИОННЫХ ПРОЦЕССОВ

Субботин С. А. – д-р техн. наук, профессор, заведующий кафедрой программных средств Национального университета «Запорожская политехника», Запорожье Украина.

Пухальская Г.В. – канд. техн. наук, доцент кафедры технология машиностроения Национального университета «Запорожская политехника», Запорожье Украина.

Леощенко С. Д. – аспирант кафедры программных средств Национального университета «Запорожская политехника», Запорожье Украина.

Олейник А. А. – д-р техн. наук, профессор кафедры программных средств Национального университета «Запорожская политехника», Запорожье Украина.

Гофман Е. А. – старший научный сотрудник научно-исследовательской части Национального университета «Запорожская политехника», Запорожье Украина.

АННОТАЦИЯ

Актуальность. Рассмотрена задача построения нейросетевой модели эксплуатационных процессов с определением оптимальной топологии, которая отличается высоким уровнем логической прозрачности и приемлемой точностью. Объектом исследования является процесс нейросетевого моделирования эксплуатационных процессов с применением индикаторной системы для упрощения выбора топологии нейромоделей.

Цель работы заключается в построении нейросетевой модели эксплуатационных процессов с высоким уровнем логической прозрачности и приемлемой точностью на основе использования индикаторной системы.

Метод. Предложено использовать систему индикаторов для определения топологических особенностей искусственных нейронных сетей, которые являются базисом для моделирования эксплуатационных процессов. Оценка уровня сложности задачи полученная на основе информации про входные данные и значений критериев оценки специфичности задачи позволяет категоризировать задачу к одному из видов сложности, чтобы определить подход к синтезу нейромоделей. Категория

сложности организованная простота позволяет на основе аналитических данных про выборку входных данных получить точное количество нейронов в скрытом слое для синтеза нейромодели с высоким уровнем логической прозрачности, что значительно расширяет их практическое использование и снижает стоимость последующих эксплуатационных процессов.

Результаты. Полученные нейромодели эксплуатационных процессов на основе исторических данных. Использование системы индикаторов позволило в значительной степени увеличить уровень логической прозрачности моделей, сохраняя высокую точность. Синтезированные нейромодели снижают ресурсоемкость промышленных процессов за счёт увеличения уровня предыдущего моделирования.

Выводы. Проведенные эксперименты подтвердили работоспособность предложенного математического обеспечения и позволяют рекомендовать его для использования на практике при моделировании эксплуатационных процессов. Перспективы дальнейших исследований могут заключаться в использовании более сложных методов отбора признаков для фиксации групповых взаимосвязей информационных признаков для построения более сложных моделей.

КЛЮЧЕВЫЕ СЛОВА: моделирование, эксплуатационные процессы, система индикаторов, нейромодель, выборка, обучение, ошибка.

ЛИТЕРАТУРА / LITERATURA

1. Haneveld W. K. K. Stochastic Programming: Modeling Decision Problems Under Uncertainty / W. K. K. Haneveld, M. H. van der Vlerk, W. Romeijnders. – Berlin : Springer, 2020. – 264 p. DOI: 10.1007/978-3-030-29219-5
2. Ceder A. Public Transit Planning and Operation: Modeling, Practice and Behavior / A. Ceder. – London, Chapman and Hall (CRC Press), 2019. – 742 p. DOI: 10.3846/16484142.2015.1114968
3. Mu E. Practical Decision Making: An Introduction to the Analytic Hierarchy Process (AHP) Using Super Decisions V2 (SpringerBriefs in Operations Research) / E. Mu, M. P. Rojas. – Berlin : Springer, 2017. – 124 p. DOI: 10.1007/978-3-319-33861-3
4. Antoniou C. Mobility Patterns, Big Data and Transport Analytics: Tools and Applications for Modeling / C. Antoniou, L. Dimitriou, F. Pereira. – Amsterdam : Elsevier, 2018. – 452 p.
5. Skormin V.A. Introduction to Process Control: Analysis, Mathematical Modeling, Control and Optimization (Springer Texts in Business and Economics) / V. A. Skormin. – Berlin : Springer, 2020. – 264 p. DOI: 10.1007/978-3-319-42258-9
6. Loy J. Neural Network Projects with Python: The ultimate guide to using Python to explore the true power of neural networks through six projects / J. Loy. – Birmingham : Packt Publishing, 2019. – 308 p.
7. Aggarwal C. C. Neural Networks and Deep Learning: A Textbook / C. C. Aggarwal. – Berlin : Springer, 2020. – 264 p. DOI: 10.1007/978-3-319-94463-0
8. Brunton S. L. Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control / S. L. Brunton. – Cambridge : Cambridge University Press, 2019. – 492 p. DOI: 10.1017/9781108380690
9. Neitzel F. Stochastic Models for Geodesy and Geoinformation Science / F. Neitzel. – Basel : Multidisciplinary Digital Publishing Institute, 2021. – 200 p. DOI: 10.3390/books978-3-03943-982-9
10. IoT Streams for Data-Driven Predictive Maintenance and IoT, Edge, and Mobile for Embedded Machine Learning (Communications in Computer and Information Science) / [J. Gama, S. Pashami, A. Bifet et al.] – Berlin : Springer, 2021. – 328 p. DOI: 10.1007/978-3-030-66770-2
11. Mathur P. IoT Machine Learning Applications in Telecom, Energy, and Agriculture: With Raspberry Pi and Arduino Using Python / P. Mathur. – Berlin : Apress, 2020. – 264 p. DOI: 10.1007/978-1-4842-5549-0
12. Implementation of the indicator system in modeling complex technical systems / [S. D. Leoshchenko, S. A. Subbotin, A. O. Oliinyk, O. E. Narivs'kiy] // Radio electronics, computer science, control. – 2021. – № 1. – P. 117–127. DOI: 10.15588/1607-3274-2021-1-12.
13. Бабенко О. Н. Анализ результатов исследования частот и форм собственных колебаний рабочей лопатки 1 ступени КНД / О. Н. Бабенко, Т. И. Прибора // Вестник двигателестроения. – № 2. – 2018. – С. 91–98.
14. Двирник Я.В. Влияние пылевой эрозии на газодинамические характеристики осевого компрессора ГТД / Я. В. Двирник, Д. В. Павленко // Вестник двигателестроения. – № 1. – 2017. – С. 56–66.
15. Эрозионная стойкость лопаток компрессора вертолетных гтд с различными типами покрытий / [В. С. Ефанов, А. Н. Прокопенко, А. В. Овчинников, Ю. Н. Внуков] // Вестник двигателестроения. – №1. – 2017. – С. 120–123.
16. Павленко Д. В. Закономерности изнашивания рабочих лопаток компрессора вертолетных двигателей, эксплуатирующихся в условиях запыленной атмосферы / Д. В. Павленко, Я. В. Двирник // Вестник двигателестроения. – № 1. – 2016. – С. 42–51

ПРОГРЕСИВНІ ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

PROGRESSIVE INFORMATION TECHNOLOGIES

ПРОГРЕССИВНЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

УДК 004.[852+94]; 519.837.3

МОДЕЛЮВАННЯ ІГРОВОЇ ЗАДАЧІ ПРИЗНАЧЕННЯ ПЕРСОНАЛУ ДЛЯ ВИКОНАННЯ ІТ-ПРОЕКТІВ НА ОСНОВІ ОНТОЛОГІЙ

Кравець П. О. – канд. техн. наук, доцент, доцент кафедри «Інформаційні системи та мережі», Національний університет «Львівська політехніка», Львів, Україна.

Литвин В. В. – д-р техн. наук, професор, завідувач кафедри «Інформаційні системи та мережі», Національний університет «Львівська політехніка», Львів, Україна.

Висоцька В. А. – канд. техн. наук, доцент, доцент кафедри «Інформаційні системи та мережі», Національний університет «Львівська політехніка», Львів, Україна.

АНОТАЦІЯ

Актуальність. У цій статті описано розв’язування ігрової задачі призначення персоналу для роботи над проектами на основі онтологічного підходу. Суть задачі полягає у такому. Існує потреба у створенні команд для виконання декількох проектів. Кожен проект задається набором необхідних онтологічних знань. Для виконання проектів менеджери залучають кваліфікованих спеціалістів (агентів), здібності яких також задаються наборами онтологій. Склад команд повинен бути таким, щоб об’єднані онтології їх агентів покривали множини онтологій відповідних проектів. Кожен агент з певними імовірностями може прийняти послідовну участь у виконанні декількох проектів. Одночасна робота агента над різними проектами не допускається. Необхідно визначити порядок виконання проектів і відповідний йому порядок призначення персоналу.

Метою дослідження є розроблення математичної моделі стохастичної гри, рекурентних марковських методів для її розв’язування, алгоритмічного та програмного забезпечення, проведення комп’ютерного експерименту, аналіз результатів та вироблення рекомендацій щодо їх практичного застосування.

Метод. Для планування виконання проектів використано стохастичний ігровий алгоритм розфарбовування неорієнтованого випадкового графа. Для цього кількість вершин графа прийнята рівною кількості проектів. Ребрами з’єднано ті вершини графа проектів, для виконання яких залучено одного і того ж агента. З урахуванням відновлювальних відмов агентів зв’язки між вершинами графа динамічно змінюються. Необхідно досягнути правильного розфарбування випадкового графа. Тоді проекти з однаково зафарбованими вершинами графа можуть бути виконані паралельно, а проекти з різними кольорами вершин – послідовно.

Результати. У статті побудовано математичну модель стохастичної гри та самонавчальний марковський метод для її розв’язування. Кожна вершина графа контролюється гравцем. Чистими стратегіями гравця є елементи палітри кольорів. Після вибору кольору власної вершини кожен гравець обчислює поточний програш як відносну кількість однакових кольорів у локальній множині сусідніх гравців. Мета гравців полягає у мінімізації функцій середніх програшів. Марковський рекурентний метод забезпечує адаптивний вибір кольорів вершин випадкового графа на основі динамічних векторів змішаних стратегій, значення яких залежать від поточних програшів гравців. Результатом стохастичної гри є асимптотично правильно розфарбований випадковий граф, коли кожному ребру початкового детермінованого графа будуть відповідати у середньому різні кольори вершин.

Висновки. Проведено комп’ютерний експеримент, який підтвердив збіжність стохастичної гри для задачі розфарбовування випадкового графа. Це дало можливість визначити порядок призначення персоналу для виконання проектів.

КЛЮЧОВІ СЛОВА: проект, агент, онтологія, призначення персоналу, розфарбовування випадкового графа, стохастична гра, марковський рекурентний метод, адаптація, самонавчання.

АБРЕВІАТУРА

ІС – інформаційна система;

ІТ – інформаційна технологія.

НОМЕНКЛАТУРА

O – онтологія;
 Ω – бібліотека онтологій;
 Π – множина ІТ-проектів;
 L – кількість ІТ-проектів або кількість вершин графа (кількість команд агентів, кількість гравців);
 Π_i – набори онтологічних знань або компетенцій, необхідних для виконання i -го проекту;
 Γ – скінченна множина команд агентів;
 K – загальна кількість агентів, що можуть бути залучені для виконання усіх проектів;
 A_k – агент з номером k або набір онтологій, що визначають здібності агента;
 $q_{i,k}$ – імовірність участі i -го агента у виконанні k -го проекту;
 V – скінченна множина вершин;
 E – множина ребер;
 D_i – множина номерів сусідніх вершин для i -ї вершини графа;
 t – момент часу;
 N – кількість чистих стратегій гравців або кількість фарб;
 X_i – вектор чистих стратегій i -го гравця;
 $x_{i,j}$ – колір j -ї фарби, вибраної i -м гравцем;
 p_i – змішана стратегія i -го гравця;
 $p_{i,j}$ – умовна імовірність вибору i -м гравцем j -го кольору;
 γ – параметр кроку навчання;
 α – коефіцієнт порядку кроку навчання;
 ε – параметр ε -симплекса;
 β – коефіцієнт порядку розширення ε -симплекса;
 λ – ваговий коефіцієнт;
 χ – індикаторна функція події;
 ξ_i – значення поточного програшу;
 E_i – поточні значення функцій середніх програшів (або втрат);
 t_{\max} – максимальна кількість кроків методу.

ВСТУП

У сучасному інформаційному просторі з розвиненими комп'ютерними мережами та засобами телекомунікації задача формування віртуальних професійних команд є важливою для організації виконання проектів, особливо в умовах дистанційної роботи [1]. Якісний підбір та належна організація командної роботи кваліфікованого персоналу є запорукою оперативного та успішного виконання проекту [2].

Проблема залучення персоналу для виконання проектів є подібною до класичної задачі про призначення засобів (устаткування або людей) для виконання робіт. Однак, у формулювання цієї задачі використані суттєві обмеження – для виконання кожної роботи можна призначити тільки один засіб,

усі засоби взаємозамінні та абсолютно надійні. У практиці формування команд виконавців проектів існує протилежний підхід. Для виконання проекту потрібно декілька різних спеціалістів, персонал може бути частково взаємозамінний і характеризується імовірністю відмов. Відмови можуть бути обумовлені недостатнім досвідом, нескоординованою роботою, зміною стану здоров'я виконавців, іншими непередбачуваними чинниками і впливають на якість і терміни виконання проекту.

Враховуючи вказані розбіжності, для розв'язування задачі про призначення персоналу для виконання проектів можна застосувати метод розфарбовування графа. Граф будується так, що вершини позначають проекти, а ребра зв'язують тільки ті вершини проектів, для виконання яких залучено однакових виконавців. Вершини на протилежних сторонах ребер повинні бути зафарбовані у різні кольори. Тоді проекти, вершини яких мають однаковий колір можна виконати паралельно, а ті, що мають різні кольори – послідовно у часі.

Відновлювальні відмови виконавців проектів формують динамічну структуру зв'язків між вершинами графа. Тому, замість детермінованого, необхідно розглядати випадковий граф проектів. Задача розфарбовування випадкового графа належить до класу NP-складних (nondeterministic polynomial time) [3, 4]. Для її розв'язування необхідно використати або розробити самонавчальні чи адаптивні методи, які можуть забезпечити наближене до оптимального розфарбовування за допустимий поліноміальний час.

Враховуючи, що граф є структурною моделлю розподіленої системи, і те, що задача розфарбовування випадкового графа має аспекти локальної конкуренції та глобальної узгодженості цілей, для її розв'язування використаємо адаптивний стохастичний ігровий метод. Адаптивні методи можуть прямо або опосередковано усереднювати контрольовані випадкові процеси, селективно формувати розв'язки, які забезпечують оптимізацію середніх значень випадкових величин або їх стохастичних моментів.

Метою дослідження є розв'язування задачі призначення персоналу для виконання ІТ-проектів на основі моделі стохастичної гри розфарбовування випадкового графа. Досягнення мети забезпечується розробленням математичної моделі стохастичної гри, рекурентних марковських методів для її розв'язування, алгоритмічного та програмного забезпечення, проведенням комп'ютерного експерименту, аналізом результатів та виробленням рекомендацій щодо їх практичного застосування.

Об'єктом дослідження є процес призначення кваліфікованого персоналу для роботи над проектами на основі онтологічного підходу в умовах невизначеності.

Предметом дослідження є стохастичний ігровий метод розфарбовування графів для розв'язування задачі безконфліктного розподілення персоналу між проектами.

1 ПОСТАНОВКА ЗАДАЧІ

Нехай необхідно організувати виконання L проектів $\Pi = \{\Pi_1, \Pi_2, \dots, \Pi_L\}$, кожен з яких задається необхідними для його виконання компетенціями $\Pi_i = \{O_1, O_2, \dots, O_r\}$ у вигляді набору онтологій. Кожна онтологія формально описує знання у певній проблемній галузі, які необхідні для виконання проекту [5].

Для онтологічної підтримки проектів необхідно залучити кваліфікованих спеціалістів у потрібних галузях знань. Використовуючи термінологію мультиагентних систем, інформаційну модель виконавців проектів робіт далі будемо називати агентами. Нехай ринок праці пропонує множину агентів $A = \{A_1, A_2, \dots, A_K\}$, $K \geq L$, які можуть бути залучені для виконання проектів. Кожен агент A_k визначається набором онтологій $A_k = \{O_{k,1}, O_{k,2}, \dots, O_{k,s}\}$ $k = 1..K$, які описують його здібності в одній або декількох галузях знань. Онтології агентів можуть мати непорожній перетин $A_i \cap A_j \neq \emptyset$, тобто агенти частково володіють однаковими здібностями. Припускається, що сукупні онтологічні знання агентів є достатніми для виконання усіх проектів.

Для виконання проектів необхідно сформувати множину команд $\Gamma = \{\Gamma_1, \Gamma_2, \dots, \Gamma_L\}$, кожна з яких є організованою групою агентів $\Gamma_i = \{A_{i,1}, A_{i,2}, \dots, A_{i,g}\}$, $i = 1..L$, де $\bigcup_{i=1..L} \Gamma_i = A$.

Необхідною умовою успішного виконання проекту є його повна онтологічна підтримка командою агентів. Здібності команди агентів повинні покривати компетенції, необхідні для виконання проекту: $\bigcup_{A_k^i \in \Gamma_i} A_k^i \supseteq \Pi_i$, $i = 1..L$.

Ігровий підхід до покриття потрібних для виконання проектів онтологій наявними онтологіями агентів розглянуто у [6].

Підбір команд агентів виконують менеджери проектів незалежно один від одного. Тоді в умовах обмеженої кількості кваліфікованих спеціалістів деякі з агентів можуть бути залучені для виконання різних проектів, тобто $\Gamma_i \cap \Gamma_j \neq \emptyset$.

Припустимо, що виконання кожного проекту не може бути перервано і виконавці не можуть переходити з одного проекту на інший до моменту його повного завершення. У зв'язку з цим виникає задача визначення порядку виконання проектів у часі. Подібно до формулювання задачі розподілу устаткування для виконання певних робіт будемо

вважати, що час виконання кожного проекту є однаковим. Тоді задачу планування послідовності виконання проектів можна звести до задачі розфарбовування неорієнтованого графа $G = (\Gamma, E)$. У такому графі вершини розмічені множинами команд агентів Γ_i , $i = 1..L$, залучених до виконання відповідного проекту, а ребра зв'язують ті вершини графа проектів, що містять однакових агентів:

$$E_i = \{e_{i,j} | \chi(\Gamma_i \cap \Gamma_j) \neq \emptyset\},$$

де $\chi() \in \{0,1\}$ – індикаторна функція події. Значення $\chi() = 1$ вказує на наявність відповідного ребра $e_{i,j}$ у графі. Якщо потужність перетину більша від 1, тобто $|\Gamma_i \cap \Gamma_j| > 1$, то розглядається мультиграф.

Тепер необхідно розфарбувати вершини графа проектів так, щоб вершини з'єднані ребрами $e_{i,j}$, $i, j = 1..L$ мали різні кольори. Тоді за таким розфарбуванням можна визначити порядок виконання проектів. Проекти, вершини яких мають однакові кольори, можуть бути виконані одночасно, а з різними кольорами – послідовно.

Однак у момент формування команд існує певна невизначеність участі агентів у виконанні того або іншого проекту. Цю невизначеність задамо імовірностями $q_{i,k}$ домовленостей між менеджером i -го проекту та k -м агентом. Ці імовірності охоплюють ряд факторів ризику невиконання проекту, наприклад, непокриття проекту необхідними онтологіями у зв'язку з недостатніми здібностями агентів, неможливість виконання проекту за станом здоров'я виконавців, непередбачувані зовнішні чинники тощо. Значення $q_{i,k}$ можуть бути визначені зі сторони менеджерів проектів, як ступінь їх впевненості у тому, що відповідний агент прийме участь у виконанні проекту, або зі сторони агента, як його схильність до виконання того або іншого проекту, або комплексно – з обох договірних сторін. Спрощено можна вважати, що $q_{i,k} = q_k$ – це імовірність участі k -го агента у виконанні будь-якого проекту. Тоді $1 - q_k$ – це імовірність відмови k -го агента. Надалі будемо вважати, що відмови агентів є відновлювальними.

Враховуючи таку стохастичну природу агентів, замість детермінованого графа необхідно розглядати випадковий граф зв'язків між проектами. Відновлювальна відмова одного із агентів, завдяки якому у графі є ребро між вершинами-проектами, призводить до тимчасової втрати цього ребра. Отже, замість заданого детермінованого графа у певні проміжки часу будуть спостерігатися його випадкові реалізації у вигляді усіх можливих підграфів.

Для розфарбовування випадкового графа детерміновані методи не можуть бути застосовані,

оскільки на кожному кроці гри змінюються реалізації графа і значення поточних програвів є випадковими величинами. Для цього необхідно використати адаптивні стохастичні методи, які зможуть пристосовуватись до випадкової зміни структури графа. Для безконфліктного планування призначення виконавців проектів нами пропонується метод на основі багатокрокової стохастичної гри [7].

2 ОГЛЯД ЛІТЕРАТУРИ

Задача про призначення персоналу для виконання проектів робіт формулюється подібно до задач розподілу устаткування, ресурсів, завдань, програм, складання розкладів, класифікації та інших аналогічних задач [8–13].

Класичне формулювання узагальненої задачі про призначення полягає у наступному. Нехай є L видів робіт або завдань, для виконання яких можна використати $K \geq L$ засобів (машин, пристроїв, робітників, програмних агентів, людей). Відомі витрати $c_{i,j}$ використання j -го засобу ($j=1..K$) для виконання i -ї ($i=1..L$) роботи. Кожен засіб можна використати тільки для одного виду роботи. Вважається, що засоби є взаємно замінними.

Необхідно знайти такий план $u = [u_{i,j} \in \{0,1\} | i=1..L, j=1..K]$, де $u_{i,j} \in \{0,1\}$, виконання робіт (розподілити засоби між роботами), щоб сумарні витрати $f(u)$ були мінімальними. Змінна $u_{i,j} = 1$, якщо засіб j виконує роботу i , та $u_{i,j} = 0$ – в іншому випадку.

Математична модель сформульованої задачі має вигляд:

1) цільова функція, яка визначає загальні витрати від використання засобів для виконання робіт:

$$f(u) = \sum_{i=1}^L \sum_{j=1}^K c_{i,j} u_{i,j} \rightarrow \min_x ;$$

2) система обмежень:

$\sum_{j=1}^K u_{i,j} = 1, i=1..L$ – для виконання i -ї роботи можна

призначити тільки один j -й засіб;

$\sum_{i=1}^L u_{i,j} \leq 1, j=1..K$ – деякі засоби можуть бути

незадіяними, якщо $K > L$.

Формулювання задачі може змінюватися залежно від інтерпретації параметрів задачі, як правило, розглядається вартість або час виконання проектів.

У частковому випадку, якщо $K = L$, то $\sum_{i=1}^L u_{i,j} = 1,$

$j=1..K$. Така задача називається основною або лінійною задачею про призначення. Вона є однією із фундаментальних задач комбінаторної оптимізації.

Для розв'язування задач про призначення можна використати методи лінійного програмування (угорський алгоритм, симплексний метод), цілочисельного програмування (метод гілок та меж, метод Гоморі) [14], мурашиної колонії [15], генетичного алгоритму [16], розфарбовування графів [17], штучних нейронних мереж [18], евристичні методи [19].

У виконаному вище формулюванні практичне використання задачі про призначення є обмеженим. На практиці для виконання однієї роботи як правило потрібно одночасно застосувати декілька різнотипних засобів, тобто $1 < \sum_{j=1}^K u_{i,j} \leq K$. Якщо ті ж засоби будуть

потрібні і для інших робіт, то $0 \leq \sum_{i=1}^L u_{i,j} \leq L$ для

$K \geq L$. У такому разі матриця $u = [u_{i,j}]$ вважається заданою і потрібно перейти до формування розкладу використання однакових засобів для виконання різних робіт. Розв'язком такої задачі є визначення порядку виконання робіт у часі, щоб один і той же засіб виконання робіт можна було використати тільки послідовно. Цю задачу можна звести до задачі розфарбовування вершин графа [20, 21].

Побудуємо неорієнтований граф, вершини якого позначають роботи (завдання), а ребра – засоби для їх виконання. Ребра з'єднують тільки ті роботи, для виконання яких потрібен однаковий засіб. Для кожного засобу $j=1..K$ побудуємо кліку графа, з'єднавши між собою вершини, для яких $u_{i,j} = 1, i=1..L$.

У результаті отримаємо неорієнтований граф $G = (V, E)$. Якщо для двох або більше робіт потрібно декілька однакових засобів, то отримаємо мультиграф.

Розфарбовуванням називають відображення $g: V \rightarrow X$, де $X = (x_1, x_2, \dots, x_N)$ ($N \leq L$) – палітра кольорів. Розфарбовування є правильним, якщо $g(k) \neq g(l)$ для кожного наявного у графі ребра, тобто $\forall e_{k,l} \in E, k, l \in V$.

Із розфарбованого графу можна отримати план виконання робіт у часі. Роботи, вершини яких зафарбовано однаковим кольором, можуть бути виконані одночасно, а зафарбовані різними кольорами – послідовно.

Розфарбовування графа великого порядку $|V|$ обмеженою кількістю N фарб вважають NP-складною задачею, яка не може бути розв'язана повним перебором варіантів за прийнятний поліноміальний час.

Для пришвидшення пошуку можна використати метод бектрекінгу (пошук з поверненням), який виключає з розгляду значну кількість варіантів однією перевіркою, будуючи дерево рішень та здійснюючи

його обхід у глибину [22]. Хоча метод класифікується як метаевристика, але він гарантовано знаходить всі розв'язки скінченної дискретної задачі за обмежений час.

Нехай вершини графа позначено літерами латинського алфавіту, а кольори – послідовними цілими числами від 1 до n . Спочатку вершина $v(a)$ фарбується в колір 1. Якщо вершина $v(b)$ не є суміжною з $v(a)$, то вона фарбується у колір 1, інакше – у колір 2. Далі розглядається вершина $v(c)$, для якої здійснюється спроба фарбування у найменший колір 1. Якщо це не можливо, то вибирається наступний допустимий колір. Після досягнення вершини, яку не можна розфарбувати жодним із n кольорів, відбувається повернення до останньої розфарбованої вершини, її колір замінюється на наступний можливий колір із впорядкованого списку кольорів і здійснюється спроба її розфарбування. Якщо і це неможливо, то відбувається повернення ще до попередньої вершини. Процес продовжується аналогічно до тих пір, поки не буде досягнуто правильного розфарбування графа, або з'ясується, що граф не можна розфарбувати в n кольорів.

Математична модель задачі розфарбування графа може бути сформульована як задача 0 – 1 цілочисельного програмування [23]:

1) цільова функція, яка вказує на те, що граф потрібно зафарбувати мінімальною кількістю кольорів:

$$z = \sum_{j=1}^Q \sum_{i=1}^L w_j c_{i,j} \rightarrow \min_c$$

2) система обмежень:

$$\sum_{j=1}^Q c_{i,j} = 1 \quad \forall i=1..L \quad - \text{кожна вершина може бути}$$

зафарбована тільки одним кольором;

$$M \cdot (1 - c_{i,j}) - \sum_{k=1}^L a_{i,k} c_{k,j} \geq 0 \quad \forall i=1..L, \quad \forall j=1..Q \quad - \text{кожна}$$

пара суміжних вершин не має однакового кольору.

Тут $[a_{i,j} | a_{i,j} \in \{0,1\}, i=1..L, j=1..L]$ – матриця суміжностей вершин графа;

$[c_{i,j} | c_{i,j} \in \{0,1\}, i=1..L, j=1..Q]$ – матриця зафарбованих вершин графа, де $c_{i,j}=1$, якщо вершина v_i має колір j ;

$(w_j | w_{j+1} > L \cdot w_j, w_1=1, j=1..Q)$ – вектор додатних ваг кольорів; $M > L$.

Для великих порядків графа визначення оптимального розв'язку задачі методами цілочисельного програмування може не дати задовільного результату у зв'язку з великою розмірністю матриць. Тому рекомендується використовувати інші методи з поліноміальним часом розв'язування задачі, наприклад, основані на

оптимізації перебору варіантів, «м'яких» обчислень або евристичних здогадках.

Огляд методів розфарбування графів можна знайти у роботах [24, 25]. Для розв'язування різноманітних практичних задач використовуються такі методи розфарбування графів:

- 1) динамічного програмування [26];
- 2) жадібний [27];
- 3) генетичний [28];
- 4) штучних нейронних мереж [29];
- 5) ройовий [30];
- 6) мурашиний [31];
- 7) багатоагентний [32];
- 8) ігровий [33, 34].

Задача розфарбування значно ускладнюється для часових випадкових графів $G(t)=(V(t), E(t))$, ребра яких розмічені імовірностями їх належності графу [35, 36]. Тоді у кожен момент часу $t=1,2,\dots$ граф проявляється однією із своїх можливих реалізацій.

Дослідження випадкових графів в основному пов'язані в отриманні імовірнісних асимптотичних оцінок його параметрів, у тому числі хроматичного числа, а інформація про ефективні методи розфарбування випадкового графа недостатньо висвітлена у наукових працях.

Детерміновані методи розфарбування випадкового графа є непродуктивними. Необхідно використати або розробити багатокрокові уточнюючі методи з елементами самонавчання, побудовані на основі «м'яких» обчислень або різноманітних евристик. Робота таких методів повинна бути спрямована на покращення хроматичної картини для досягнення правильного розмалювання графа в асимптотиці часу. Для розфарбування випадкового графа можуть бути використані модифікації методів 1 – 8. Особливу увагу слід приділити розробленню стохастичних варіантів реалізації цих методів.

Для розв'язування задачі розфарбування випадкового графа, побудованого на основі онтологічної підтримки проектів, нами пропонується застосування стохастичного ігрового методу, який має властивості самонавчання та адаптації до невизначеностей.

3 МАТЕРІАЛИ І МЕТОДИ

З кожною вершиною (проектом) графа пов'яжемо гравця, чисті стратегії якого визначають палітру пронумерованих кольорів $X_i = \{x_{i,1}, x_{i,2}, \dots, x_{i,N}\}$, де $x_{i,j}$ – номер кольору; N – кількість елементів палітри кольорів, яка обмежується значенням $N=L$, що є необхідним для розфарбування повнозв'язного графа.

Вибір чистих стратегій $x_i(t) \in X_i$ здійснюється гравцями випадково і незалежно у моменти часу $t=1,2,\dots$. У зв'язку з відмовами гравців, на кожному кроці гри визначається випадкова реалізація графа $G(t) = (\Gamma(t), E(t)) \subseteq G$, у якому змінюється лише

розмітка вершин $\Gamma(t) = \{\Gamma_i(t) | i = 1..L\}$ множинами агентів

$$\Gamma_i(t) = \{A_{i,1}, A_{i,2}, \dots, A_{i,g}\} \setminus \{A_{i,k}(t) | k = 1..J, l \leq g\}$$

та відповідні зв'язки $E(t)$ між вершинами. Тут $\{A_{i,k}(t)\}$ – агенти, що відмовили у момент часу t . Кількість вершин L і відповідних до них проектів Π_i залишаються незмінними. Гравці не володіють інформацією про поточну реалізацію графа в цілому. Кожному гравцю відомий тільки їх локальний підграф – множина сусідніх вершин, що приєднані ребрами до контрольованої гравцем вершини графа.

Нехай $K_i^{loc}(t) = |E_i(t)|$ – кількість ребер i -ї вершини випадкового графа у момент часу t . Тоді i -й гравець приймає участь у грі, якщо йому відповідає неізолювана вершина, тобто $K_i^{loc}(t) \geq 1$.

Після завершення вибору чистих стратегій усіма гравцями кожен з них обчислює значення поточного програшу, як середню кількість однакових кольорів сусідніх вершин графа:

$$\xi_i(t) = \left(K_i^{loc}(t)\right)^{-1} \sum_{j \in D_i(t)} \chi(x_i(t) = x_j(t)), \quad (1)$$

де $\chi() \in \{0,1\}$ – індикаторна функція події; $D_i(t) = \{index(e_{i,j}(t))\}$ – множина номерів сусідніх вершин для i -ї вершини випадкового графа. Сусідніми до вершини i є вершини випадкового графа, що мають з нею безпосередній зв'язок у момент часу t . Очевидно, що $|D_i(t)| = K_i^{loc}(t)$.

Гравці оцінюють свої дії у ході гри за допомогою поточних значень функцій середніх програшів (або втрат):

$$\Xi_i(t) = t^{-1} \sum_{\tau=1}^t \xi_i(\tau), \quad i = 1..L. \quad (2)$$

Хід стохастичної гри в цілому можна контролювати за допомогою системної функції середніх програшів:

Стратегія поведінки кожного гравця повинна бути спрямована на мінімізацію власних функцій середніх програшів (на мінімізацію співпадань кольорів сусідніх вершин графа):

$$\overline{\lim}_{t \rightarrow \infty} \Xi_i(t) \rightarrow \min_{\{x_i(t)\}}, \quad i = 1..L. \quad (4)$$

Розв'язки задачі багатокритеріальної оптимізації (4) необхідно шукати у множинах точок колективної оптимальності, наприклад, Слейтера, Неша, Парето або інших [37]. Найчастіше у задачах без обміну

поточною інформацією про стратегії, стани та програші гравців (або з мінімально потрібним обміном) використовується критерій рівноваги за Нешем, суть якого полягає у наступному. У точці рівноваги за Нешем кожному гравцю не вигідно змінювати свою стратегію, якщо усі інші гравці притримуються точки рівноваги:

$$\overline{\lim}_{t \rightarrow \infty} \left[\Xi_i(t, \{x^{D_i}(t)\}) - \Xi_i(t, \{\tilde{x}^{D_i}(t)\}) \right] \leq 0, \quad i = 1..L. \quad (5)$$

У (5) використано такі позначення: D_i – локальна множина сусідніх гравців (або вершин графа) для i -го гравця (або вершини); $X^{D_i} = \otimes_{j \in D_i} X_j$ – простір комбінованих чистих стратегій гравців з локальної множини D_i ; $X^{D_i} \subseteq X$; $X = \otimes_{i=1..L} X_i$ – простір комбінованих чистих стратегій усієї множини гравців; \otimes – операція декартового добутку; $x^{D_i} \in X^{D_i}$; \tilde{x}_i – відхилення від точки рівноваги стратегія i -го гравця; $\tilde{x}^{D_i} = x^{D_i} \setminus x_i + \tilde{x}_i \in X^{D_i}$ – комбінована стратегія сусідніх гравців з підмножини D_i після заміни стратегії i -го гравця; $x_i, \tilde{x}_i \in X_i$.

Отже, спостерігаючи випадкові поточні програші (1), гравці повинні навчитися вибирати фарби $x_i(t) \in X_i$ із палітри кольорів X^i так, щоб сформована послідовність варіантів $\{x_i(t)\}$ забезпечила виконання мети (4) в асимптотиці часу $t \rightarrow \infty$. На практиці кількість кроків стохастичної гри обмежується деяким максимальним значенням або досягненням правильного розфарбовування графа.

Для генерування послідовностей стратегій $\{x_i(t) | i = 1..L\}$, $t = 1, 2, \dots$, які забезпечують виконання критеріїв (4), побудуємо імовірнісний механізм на основі змішаних стратегій гравців $\{p_i(t) | i = 1..L\}$. Змішана стратегія $p_i(t) = (p_{i,1}(t), p_{i,2}(t), \dots, p_{i,N}(t))$ складається з умовних імовірностей вибору чистих стратегій:

$$p_{i,j}(t) = P\{x_i(t) = x_{i,j} | u_i(\tau), \xi_i(\tau), \tau = 1, 2, \dots, t-1\}, \quad j = 1..N,$$

де $\{x_i(\tau)\}$ – передісторія чистих стратегій, вибраних гравцем з номером i ; $\{\xi_i(\tau)\}$ – передісторія отриманих за це програшів.

Для формування послідовностей з потрібними властивостями змішані стратегії на кожному кроці гри змінюються за рекурентним методом [38]:

$$p_i(t+1) = \pi_{\epsilon(t+1)}^N \{p_i(t) - \gamma(t)R(p_i(t), x_i(t), \xi_i(t))\}, \quad (6)$$

де $\pi_{\varepsilon_{t+1}}^N$ – проєктор на одиничний ε -симплекс $S_\varepsilon^N \subseteq S^N \subset R^N$ (тут верхній індекс не є показником степеня, а вказує на кількість вимірів простору дійсних чисел); $\gamma(t)$ – монотонно спадна послідовність невід’ємних величин, яка регулює величину кроку методу; R – крок методу; $\varepsilon(t)$ – монотонно спадна послідовність невід’ємних величин, яка регулює швидкість розширення ε -симплексу.

Координати точок одиничного симплексу є нормовані так, що їх сума дорівнює 1:

$$S^N = \left\{ p \mid \sum_{j=1}^N p_j = 1; p_j \geq 0 \ (j=1..N) \right\}.$$

Одиничний ε -симплекс є компактною підмножиною одиничного симплексу:

$$S_\varepsilon^N = \left\{ p_i \mid p_i \in S^N; p_{i,j} \geq \varepsilon \ (j=1..N) \right\},$$

$$\varepsilon \in (0, 1/N), p_i(t) \in S_\varepsilon^N.$$

Рекурентний метод (6) потрібно побудувати так, щоб при виборі стратегії $x_{i,j}(t)$ відповідна імовірність $p_{i,j}(t)$ зменшилася пропорційно величині поточного програшу $\xi_i(t)$. Решта елементів змішаної стратегії не змінюється, або зростає пропорційно $\xi_i(t)$. Тобто метод повинен підсилювати імовірності вибору більш вдалих стратегій, корисних для виконання критеріїв мінімізації середніх програшів гравців (4). Метод з такими властивостями називається адаптивним або самонавчальним.

Після обчислення нових значень векторів змішаних стратегій відбувається їх проєктування на ε -симплекс S_ε^N . Оператор проєктування $\pi_{\varepsilon_{t+1}}^N$ задовольняє умови: $\|p_i - \pi_{\varepsilon}^N \{q_i\}\| \leq \|p_i - q_i\|$;

$$\pi_{\varepsilon}^N \{q_i\} \in S_\varepsilon^N \ i=1..L, \forall p_i \in S_\varepsilon^N, \forall q_i \in R^N.$$

Проєктування на розширюваний ε -симплекс забезпечує виконання умови $p_{i,j}(t) \geq \varepsilon(t), j=1..N$, яка необхідна для повноти статистичної інформації щодо вибору чистих стратегій.

Побудову рекурентних методів виду (6) виконаємо методом стохастичної апроксимації [39, 40]. Припустимо, що математичні сподівання випадкових величин $M\{\xi_i(x,t)\} = v_i(x)$ відомі для всіх комбінованих стратегій $x \in X = \otimes_{i=1..L} X_i$.

Визначимо полілінійну функцію середніх програшів матричної гри:

$$V_i(p^{D_i}) = \sum_{x^{D_i} \in X^{D_i}} v_i(x^{D_i}) \prod_{j \in D_i; x_j \in X^{D_i}} p_j(x_j), \quad (7)$$

де D_i – локальна множина сусідніх гравців; $X^{D_i} = \otimes_{j \in D_i} X_j$ – простір комбінованих чистих стратегій гравців з локальної множини D_i ; $x^{D_i} \in X^{D_i}$ – одна із комбінованих локальних стратегій гравців; $p^{D_i} \in S^{D_i} = \prod_{j \in D_i} S_j^N$; $p_i \in S^N$.

Метою матричної гри є мінімізація функцій середніх програшів (7):

$$V_i(p^{D_i}) \rightarrow \min_{p_i}, \ i=1..L. \quad (8)$$

Розв’язок матричної гри у точці рівноваги за Нешем у змішаних стратегіях визначається такою умовою:

$$V_i(p_*^{D_i}) - V_i(p_*^{D_i \setminus \{i\}}, p_i) \leq 0, \ i=1..L,$$

де $V_i(p_*^{D_i})$ – функція середніх програшів, визначена у точці Неша $p_*^{D_i} \in S^{D_i}$ на локальному симплексі $S^{D_i} = \prod_{j \in D_i} S^N$ комбінованих змішаних стратегій

гравців з множини D_i сусідніх гравців; $V_i(p_*^{D_i}, p_i)$ – функція середніх програшів, визначена на симплексі S^{D_i} для будь-якого відхилення змішаної стратегії i -го гравця від точки Неша.

Згідно із теорією стохастичної апроксимації [38–40], для мінімізації системи функцій (8) визначимо вектор руху R рекурентного методу (6) так, щоб його математичне сподівання було градієнтом функції середніх програшів (7):

$$M\{R(p_i(t), x_i(t), \xi_i(t))\} = \nabla_{p_i} V_i(p^{D_i}).$$

Враховуючи, що

$$\nabla_{p_i} V_i = M \left\{ \frac{\xi_i(t)}{e^T(x_i(t)) p_i(t)} e(x_i(t)) \mid p_i(t) = p_i \right\},$$

де $e(x_i(t))$ – одиничний вектор-індикатор вибору чистої стратегії $x_i(t) \in X_i$, $e^T(x_i(t))$ – транспонований вектор, отримаємо градієнтний рекурентний метод розв’язування ігрової задачі:

$$p_i(t+1) = \pi_{\varepsilon(t+1)}^N \left\{ p_i(t) - \gamma(t) \frac{\xi_i(t)e(x_i(t))}{e^T(x_i(t))p_i(t)} \right\}. \quad (9)$$

Інші рекурентні методи можна отримати з умови доповняльної нежорсткості [41, 42], яка виконується для точок рівноваги за Нешем у повністю змішаних стратегіях:

$$\nabla_{p_i} V_i = V_i e_N, \quad i=1..L, \quad (10)$$

де e_N – вектор, що складається з N одиниць. Умова доповняльної нежорсткості описує незалежність функцій середніх втрат гравців від їх власних змішаних стратегій у точці Неша. Як би не змінювалася змішана стратегія $p_i(t)$ гравця з номером i ($i=1..L$) на одиничному симплексі, коли усі інші гравці притримуються своїх стратегій у точці Неша, значення функції вигрешів V_i залишається постійним.

Оскільки

$$\nabla_{p_i} V_i - V_i e_N = M \left\{ \xi_i(t) \left[\frac{e(x_i(t))}{e^T(x_i(t))p_i(t)} - e_N \right] \middle| p_i(t) = p_i \right\},$$

то методом стохастичної апроксимації отримаємо рекурентний метод доповняльної нежорсткості:

$$p_i(t+1) = \pi_{\varepsilon(t+1)}^N \left\{ p_i(t) - \gamma(t) \xi_i(t) \left[\frac{e(x_i(t))}{e^T(x_i(t))p_i(t)} - e_N \right] \right\}. \quad (11)$$

Для врахування розв'язків на межі одиничного симплексу виконаємо зважування векторної умови (10) елементами вектора p_i :

$$\text{diag}(p_i)[V_i e_N - \nabla V_i] = 0, \quad (12)$$

де $\text{diag}(p_i)$ – квадратна діагональна матриця порядку N , складена з елементів вектора p_i .

Враховуючи, що

$$\text{diag}(p_i)(V_i e_N - \nabla_{p_i} V_i) = M \left\{ \xi_i(t) [p_i(t) - e(x_i(t))] \middle| p_i(t) = p_i \right\},$$

методом стохастичної апроксимації отримаємо рекурентний метод зваженої доповняльної нежорсткості (12):

$$p_i(t+1) = \pi_{\varepsilon(t+1)}^N \left\{ p_i(t) - \gamma(t) \xi_i(t) [e(x_i(t)) - p_i(t)] \right\}. \quad (13)$$

Завдяки такій динамічній реорганізації змішаних стратегій на основі опрацювання поточних програшів, методи (9), (11), (13) забезпечують адаптивний вибір чистих стратегій у часі.

© Кравець П. О., Литвин В. В., Висоцька В. А., 2022
DOI 10.15588/1607-3274-2022-1-14

Параметри $\gamma(t)$ та $\varepsilon(t)$ є монотонно спадними послідовностями невід'ємних величин і використовуються для керування збіжністю рекурентних методів. Ці параметри можна обчислити так:

$$\gamma(t) = \gamma(0)t^{-\alpha}, \quad \varepsilon(t) = \varepsilon(0)t^{-\beta}, \quad (14)$$

де $\gamma_0, \alpha, \beta > 0$, $\varepsilon(0) \in (0, N^{-1})$. Збіжність змішаних стратегій $p_i(t)$ $i=1..L$ до оптимальних значень з імовірністю 1 або у середньоквадратичному визначається співвідношеннями параметрів γ_t та ε_t , які повинні задовольняти фундаментальні умови стохастичної апроксимації [38–40].

Працездатність (у сенсі виконання критеріїв (4)) рекурентних алгоритмів забезпечується виконанням умови псевдоградієнтності вектора R відносно функції Ляпунова $\Delta(p)$ [38]:

$$\langle M \{ R \{ x_i(t), p_i(t), \xi_i(t) \} \middle| p_i(t) = p_i \}, \nabla_{p_i} (\Delta(p)) \rangle \geq 0,$$

де $\langle \cdot, \cdot \rangle$ – скалярний добуток векторів в евклідовому

просторі; $p_i \in S^N$; $p \in S = \prod_{i=1}^L S_i^N$.

Функція Ляпунова Δ повинна бути диференційованою по p_i , $i=1..L$, мати корені в точках асимптотичної оптимальності $\Delta(p^*) = 0$, бути знакододатною $\Delta(p) > 0$ на комбінованому одиничному симплексі $\forall p \in S$, $p \neq p^*$. Для оптимізації функції середніх вигрешів на системі одиничних симплексів можна прийняти $\Delta(t) = \sum_{i=1}^L \|p_i(t) - p_i^*(t)\|$, де $p_i^*(t)$ – асимптотично-оптимальний розв'язок у змішаних стратегіях для i -го гравця.

Для розглянутих рекурентних методів функція Ляпунова $\Delta(t)$ може бути визначена як похибка умови доповняльної нежорсткості (квадрат норми різниці змішаних стратегій):

$$\Delta(t) = \sum_{i=1}^L \|p_i(t) - \tilde{p}_i(t)\|^2,$$

де $\tilde{p}_i(t) = \text{diag}(p_i(t))(\nabla_{p_i(t)} V_i(t))/V_i(t)$ – зважена змішана стратегія i -го гравця, для якої виконується умова доповняльної нежорсткості.

Середньоквадратичну швидкість збіжності рекурентних методів можна оцінити асимптотичним методом моментів Чжуна [38]:

$$\overline{\lim}_{n \rightarrow \infty} n^0 M \{ \Delta(t) \} = \vartheta, \quad (15)$$

де θ – параметр порядку, ϑ – величина швидкості збіжності. Більшим значенням θ та меншим ϑ відповідає більша швидкість збіжності ігрового методу.

У знакододатних середовищах, для яких $V_i(p^{D_i}) > 0$ на системі одиничних симплексів, теоретичний порядок середньоквадратичної збіжності методів (9) та (11) становить $\theta = \min(1 + \beta - \alpha, \alpha - \beta)$ при обмеженнях на параметри $\alpha \in (0, 1]$; $0 < \beta < \alpha$. Теоретичний порядок середньоквадратичної швидкості збіжності методу (13) дорівнює $\theta = \min(1 + \beta - \alpha, \alpha)$ при обмеженнях $\alpha \in (0, 1]$; $\beta > 0$ [42].

Вибір чистих стратегій (кольорів вершин) $x_{i,k}(t)$, $i = 1..L$ здійснюється гравцями випадково на основі змішаних стратегій $p_i(t) = (p_{i,1}(t), p_{i,2}(t), \dots, p_{i,N}(t))$:

$$k = \arg \left(\min_{k=1..N} \sum_{j=1}^k p_{i,j}(t) > \omega \right) \in \{1..N\}, \quad (16)$$

де $\omega \in [0, 1]$ – дійсне випадкове число з рівномірним розподілом.

Стохастична гра розпочинається з ненавчених змішаних стратегій зі значеннями елементів $p_{i,j}(0) = 1/N$, де $j = 1..N$. На протязі наступних моментів часу динаміка векторів змішаних стратегій визначається за одним із марковських рекурентних методів. Рекурентні методи (9), (11), (13) забезпечують адаптацію стратегій гравців як до зміни реалізацій випадкового графа так і до обчислених на їх основі апіорі невідомих поточних втрат.

Отже, в моменти часу $t = 1, 2, \dots$ кожен гравець на основі змішаної стратегії $p_i(t)$ вибирає чисту стратегію $x_i(t)$ (16) і до моменту часу $t+1$ отримує поточний програш $\xi_i(t)$ (1), після чого обчислює змішану стратегію $p_i(t+1)$ згідно одного із методів (9), (11), (13).

Кольори вершин випадкового графа визначаються як заокруглене до цілого значення математичне сподівання можливих кольорів, обчислене для останнього кроку стохастичної гри:

$$\bar{x}_i(t) = \text{int} \left(\sum_{i=1}^N p_i(t) x_i(t) \right), \quad i = 1..L. \quad (17)$$

Алгоритм стохастичної гри.

Кроки 1–2 визначають ініціалізацію даних та виконують підготовчі дії, а кроки 3–11 реалізують стохастичну гру розфарбовування вершин випадкового графа.

Крок 1. Задати початкові значення параметрів:

$t = 0$ – початковий момент часу; $L, K, N = L$ – кількість чистих стратегій гравців (кількість кольорів палітри фарб);

$\Omega = \{O_1, O_2, \dots, O_m\}$ – бібліотека онтологій;

$\Pi_i = \{O_{i,1}, O_{i,2}, \dots, O_{i,r}\} \subseteq \Omega$, $i = 1..L$ – набори онтологічних знань або компетенцій, необхідних для виконання проектів;

$A_k = \{O_{k,1}, O_{k,2}, \dots, O_{k,s}\} \subseteq \Omega$, $k = 1..K$ – набори онтологій, що визначають здібності агентів;

$q_{i,k}$, $i = 1..L$, $k = 1..K$ – імовірності участі агентів у виконанні проектів;

$U_i = \{u_{i,1}, u_{i,2}, \dots, u_{i,N}\}$, $i = 1..L$ – вектори чистих стратегій гравців;

$p_i(0) = ((1/N)_j | j = 1..N)$, $i = 1..L$ – початкові значення змішаних стратегій гравців;

$\gamma > 0$ – параметр кроку навчання;

$\alpha \in (0, 1]$ – коефіцієнт порядку кроку навчання;

ε – параметр ε -симплекса;

$\beta > 0$ – коефіцієнт порядку розширення ε -симплекса;

$\lambda \in [0, 1]$ – ваговий коефіцієнт;

t_{max} – максимальна кількість кроків методу.

Крок 2. Виконати підготовчі дії:

2.1. Виконати покриття проектів онтологіями, залучивши до виконання проектів відповідних агентів.

2.2. Побудувати граф, вершини якого позначають проекти (команди агентів), а ребра з'єднують ті вершини (проекти), для виконання яких залучені однакові агенти. Сформувані початкову матрицю суміжностей вершин графа.

2.3. З кожною вершиною графа пов'язати гравців, які вибирають варіанти поточних кольорів вершин графа.

Крок 3. Визначити поточний склад команд агентів, залучених до виконання проектів з імовірністю $q_{i,k}$ та виконати нову розмітку вершин графа.

Крок 4. Визначити поточну матрицю суміжностей вершин графа.

Крок 5. Вибрати чисті стратегії (кольори вершин графа) $x_i(t) \in X_i$ гравців $i = 1..L$ згідно з (16).

Крок 6. Обчислити значення поточних програшів $\xi_i(t)$, $i = 1..L$ згідно з (1).

Крок 7. Обчислити параметри $\gamma(t)$ та $\varepsilon(t)$ згідно з (14).

Крок 8. Обчислити елементи векторів змішаних стратегій $p_i(t)$, $i = 1..L$ згідно з (13).

Крок 9. Обчислити поточні значення функцій середніх програшів $\Xi_i(t)$ (2) кожного гравця і на їх основі обчислити системну функцію середніх програшів $\Xi(t)$ (3) стохастичної гри розфарбовування графа.

Крок 10. Задати наступний момент часу $t := t + 1$.

Крок 11. Якщо $t < t_{\max}$, то перейти на крок 3, інакше – на крок 12.

Крок 12. Обчислити усереднені значення кольорів $\bar{x}_i(t)$, $i=1..L$ для вершин графа згідно з (17). Кінець гри.

4 ЕКСПЕРИМЕНТИ

Нехай задані:

1) бібліотека онтологій $\Omega = \{O_1, O_2, O_3, O_4, O_5\}$;

2) компетенції, необхідні для виконання проектів $\Pi_1 = \{O_1, O_3, O_4\}$, $\Pi_2 = \{O_2, O_3, O_5\}$,

$\Pi_3 = \{O_1, O_4, O_5\}$, $\Pi_4 = \{O_1, O_3, O_5\}$;

3) здібності агентів $A_1 = \{O_1, O_4\}$, $A_2 = \{O_2, O_3\}$, $A_3 = \{O_1, O_5\}$, $A_4 = \{O_3, O_4\}$, які можуть бути залучені для виконання проектів.

Виходячи з цих даних, можливі такі покриття проектів агентами: $\Pi_1 = \{A_1, A_2\}$, $\Pi_2 = \{A_2, A_3\}$, $\Pi_3 = \{A_1, A_3\}$, $\Pi_4 = \{A_3, A_4\}$. Дійсно, для заданих проектів справедливі наступні відношення покриття онтологій:

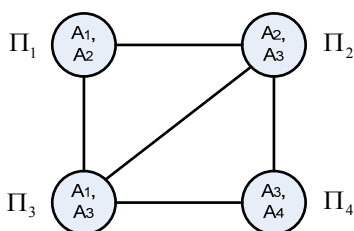
$\Pi_1 : A_1 \cup A_2 = \{O_1, O_2, O_3, O_4\} \supseteq \{O_1, O_3, O_4\}$;

$\Pi_2 : A_2 \cup A_3 = \{O_1, O_2, O_3, O_5\} \supseteq \{O_2, O_3, O_5\}$;

$\Pi_3 : A_1 \cup A_3 = \{O_1, O_4, O_5\} \supseteq \{O_1, O_4, O_5\}$;

$\Pi_4 : A_3 \cup A_4 = \{O_1, O_3, O_4, O_5\} \supseteq \{O_1, O_3, O_5\}$.

На рис. 1 зображено граф, вершинами якого є проекти Π_i , а ребрами – зв'язки між тими проектами, для виконання яких планується залучити одних і тих же агентів. У вершинах графа наведено список агентів A_k , $k=1..K_i$, залучених до виконання проектів. З кожною вершиною (проектом) пов'язується гравець, який виконує ходи стохастичної гри щодо вибору поточного кольору вершини, залежно від кольорів з'єднаних ребрами сусідніх вершин графа.



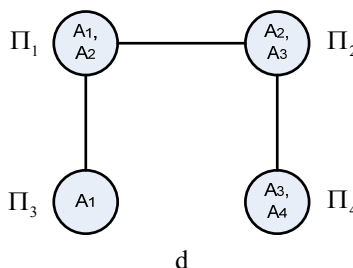
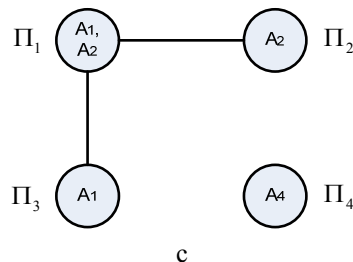
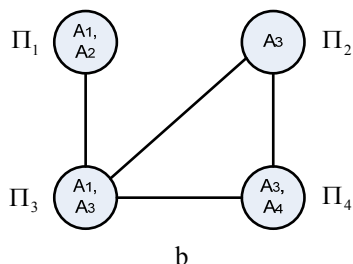
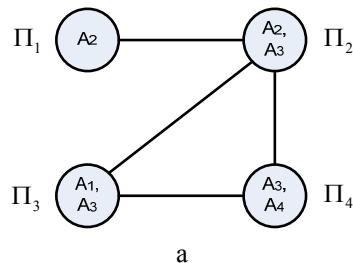
Рисунки 1 – Граф залежностей проектів від виконавців

Відмови агентів призводять до зміни складу команд виконавців проектів і до відповідної зміни зв'язків між вершинами графа. У результаті, замість детермінованого графа, отримаємо випадковий граф, декілька можливих реалізацій якого зображено на рис. 2.

На рис. 2а зображено реалізацію випадкового графа для випадку, коли відмовив агент A1 проекту Π1. Така ж структура графа буде у випадку відмови агента A1 проекту Π3, або відмови агента A1 для обох

проектів Π1 та Π3. Результатом цього є втрата зв'язку між проектами Π1 та Π3.

Відмова агента A2 проекту Π2 призведе до реалізації випадкового графа, зображеної на рис. 2б. Аналогічний результат отримаємо у випадку відмови агента A2 проекту Π1, або відмови агента A2 проектів Π1 та Π2.



Рисунки 2 – Реалізації випадкового графа

Випадок відмови агента A3, залученого до виконання проектів Π2, Π3 та Π4, зображено на рис. 2с. На цьому рисунку видно, що відмови агентів можуть призвести до порушення зв'язності графа проектів.

Реалізація графа, зображеного на рис. 2д, отримана у результаті відмови агента A3 проекту Π3.

Для заданого випадкового графа необхідно визначити такий порядок виконання проектів, для якого можлива участь кожного агента є послідовною у часі.

5 РЕЗУЛЬТАТИ

Розв'язування цієї задачі виконаємо стохастичним ігровим методом (13) розфарбовування випадкового графа. Для цього задамо такі параметри стохастичної гри: $L = 4$; $N = L$; $K = 4$; $\gamma_0 = 1$; $\alpha = 0,01$; $\varepsilon_0 = 0,999N^{-1}$; $\beta = 2$.

Вплив імовірностей $q_k = q$, $i = 1..K$ участі агентів у виконанні проектів на збіжність стохастичної гри зображено на рис. 3 у логарифмічному масштабі.

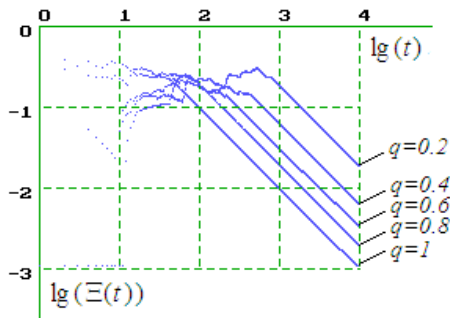


Рисунок 3 – Залежність системної функції середніх втрат від імовірностей участі агентів у виконанні проектів

Параметр θ порядку швидкості збіжності (15) ігрового методу визначається тангенсом кута, утвореного лінійною апроксимацією графіка системної функції середніх втрат та віссю часу. Як видно на рис. 3, середній порядок швидкості збіжності ігрового методу є наближеним до 1. Він практично не змінюється для різних імовірностей участі агентів у виконанні проектів. Із зменшенням значення цих імовірностей лише зростає час навчання стохастичної гри.

На рис. 4 зображено апроксимовану показникову залежність середньої кількості кроків \bar{t} , необхідних для розфарбовування випадкового графа, від імовірностей $q_k = q$, $i = 1..K$ участі агентів у гри.

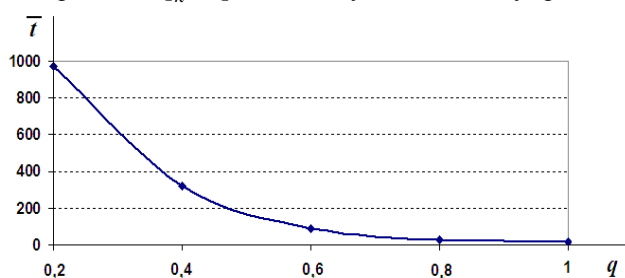


Рисунок 4 – Середня кількість кроків навчання стохастичної гри

Зростання імовірності $q \rightarrow 1$ участі агентів у виконанні проектів призводить до збільшення подібності реалізацій випадкового графа $G(t)$ до заданого детермінованого графа G :

$$\delta(t) = \mathfrak{M}(G(t), G) \rightarrow 0,$$

де $\delta(t) \in [0,1]$ – міра близькості графів у моменти часу $t = 1, 2, \dots$ [43]. Результатом цього є зменшення кількості кроків навчання стохастичної гри. Для абсолютно надійних агентів ($q = 1$) потрібно 10–20 кроків стохастичної гри для формування правильного розфарбовування зображеного на рис. 1 графа.

Розв'язком стохастичної гри є зображений на рис. 5 розфарбований граф, отриманий для імовірностей $q_k = q = 0,8$, $i = 1..K$ участі агентів у виконанні проектів.

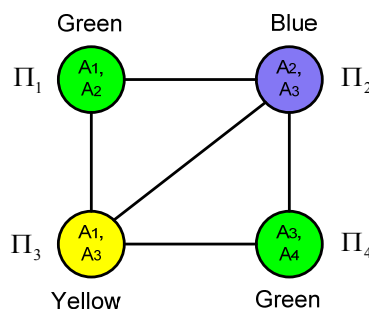


Рисунок 5 – Розфарбований граф проектів

Проекти, що відповідають вершинам з однаковим кольором, можуть виконуватись одночасно (паралельно). Дві послідовності виконання проектів із шести можливих зображено на рис. 6.

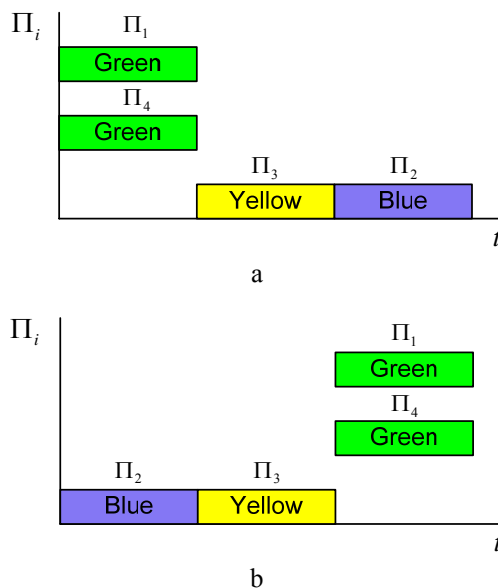


Рисунок 6 – Послідовності виконання проектів

Як видно на рис. 6а та рис. 6б, проекти Π_1 та Π_4 можуть виконуватись одночасно, а проекти Π_2 та Π_3 – послідовно з цими паралельними проектами. Застосування алгоритму розфарбовування графа визначає лише порядок виконання проектів без врахування їх, можливо, різної тривалості.

6 ОБГОВОРЕННЯ

В умовах апіорної невизначеності випадкового графа проектів важливою є максимізація імовірностей покриття проектів.

Нехай $Z_i = 2^{K_i}$ – повний простір комбінованих станів агентів, залучених до виконання i -го проекту ($i=1..L$), де K_i – кількість таких агентів. Значення стану $s_{i,j}=1$ сигналізує про участь $j \Leftrightarrow k$ -го агента, а значення $s_{i,j}=0$ – про його відмову від участі у виконанні проекту Π_i . Тут операція \Leftrightarrow виконує взаємно однозначне відображення послідовних номерів $j=1..K_i$ агентів A_j проекту Π_i та реальних номерів агентів A_k . Тоді імовірність покриття проекту $p_{cov}(\Pi_i)$ визначається як сума імовірностей тих комбінованих станів $(s_{i,1}s_{i,2}...s_{i,K_i})$, для яких індивідуальні стани агентів дорівнюють 1 і об'єднані онтології цих агентів покривають даний проект.

Наприклад, для зображеного на рис. 1 графа кожен проект Π_i , $i=1..L$ має такий простір станів для двох залучених для його виконання агентів:

$$Z_i = \{(s_{i,1}s_{i,2}) \mid s_{i,j} \in \{0,1\}, j=1,2\} = \{00,01,10,11\}.$$

Очевидно, що імовірність реалізації усіх комбінованих станів дорівнює 1: $(1-q_{i,1})(1-q_{i,2}) + (1-q_{i,1})q_{i,2} + q_{i,1}(1-q_{i,2}) + q_{i,1}q_{i,2} = 1$. Тут перший індекс позначає номер проекту, а другий – номер агента.

Враховуючи склад множин онтологій, імовірність покриття проекту, наприклад Π_1 , визначається так: $p_{cov}(\Pi_1) = q_{1,1}q_{1,2}$, де $\Pi_1 = \{A_1, A_2\}$. Якщо $q_{1,k} = q$, $k=1,2$, то матимемо квадратичну залежність імовірності покриття $p_{cov}(\Pi_1) = q^2$ проекту від імовірності відмов агентів.

Імовірність покриття можна підвищити залученням у проект надлишкових агентів, онтологій яких хоча б частково входять у множину онтологій проекту. Для прикладу введемо у проект Π_1 додаткового агента A_4 , що призведе до появи у графі нового зв'язку між вершинами Π_1 та Π_4 . У результаті цього простір станів агентів матиме вигляд:

$$Z_1 = \{(s_{1,1}s_{1,2}s_{1,3}) \mid s_{1,j} \in \{0,1\}, j=1..3\} = \{000,001,010,011,100,101,110,111\}.$$

З урахуванням нового набору онтологій імовірність покриття проекту $\Pi_1 = \{A_1, A_2, A_4\}$ тепер буде дорівнювати:

$$p'_{cov}(\Pi_1) = q_{1,1}(1-q_{1,2})q_{1,4} + q_{1,1}q_{1,2}(1-q_{1,4}) + q_{1,1}q_{1,2}q_{1,4}.$$

Легко перевірити, що $p'_{cov}(\Pi_1) \geq p_{cov}(\Pi_1)$. В окремому випадку для однакових значень імовірностей участі агентів у виконанні проекту маємо $p'_{cov}(\Pi_1) = 2q^2(1-q) + q^3 \geq q^2$.

Графіки імовірностей $p_{cov}(\Pi_1)$ та $p'_{cov}(\Pi_1)$ покриття проекту Π_1 для $q_{1,k} = q \in [0,1]$, $k=1..K$ наведено на рис. 7.

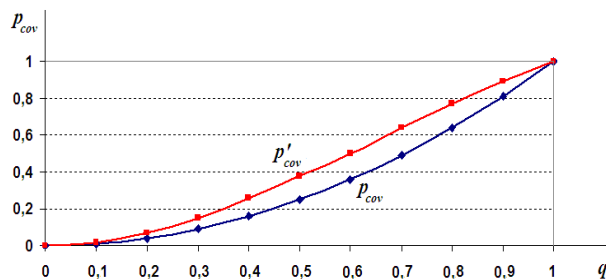


Рисунок 7 – Імовірності покриття проектів

Надлишкове покриття проекту по-різному впливає на збіжність стохастичної гри, залежно від варіантів залучення агентів для виконання проектів. Типові реалізації функції усереднених у часі програшів для проекту Π_1 подані на рис. 8 для різних імовірностей покриття. Графік 1 отримано для $p_{cov}(\Pi_1) = 0.64$, а графік 2 – для $p'_{cov}(\Pi_1) = 0.768$, обчислених для $q_{1,k} = q = 0.8$, $k=1..K$.

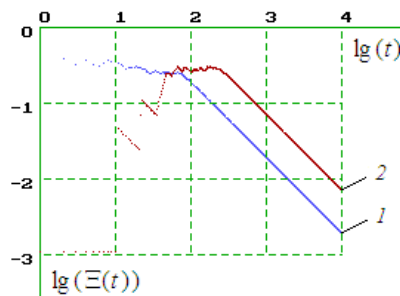


Рисунок 8 – Вплив надлишкового покриття проекту на збіжність стохастичної гри

Як видно на рис. 7 надлишкове покриття проектів через залучення додаткових агентів може забезпечити зростання імовірності покриття затребуваних онтологій проектів наявними онтологіями агентів. Однак, як показано на рис. 8, для випадкових графів це може призвести до погіршення показників збіжності ігрового методу за рахунок залежності проектів від одних і тих же агентів.

Результатом залучення однакових додаткових агентів є зростання зв'язності графа проектів, який у випадку відмов агентів розпадається на більшу кількість підграфів, у ньому частіше з'являються ізольовані вершини, що послаблює умови збіжності стохастичної гри розфарбовування випадкового графа. Збільшення порядку графа, що еквівалентно збільшенню кількості проектів, також призводить до сповільнення збіжності стохастичної гри.

Залучення надлишкових агентів до виконання проектів можна обмежити додатковим критерієм мінімізації вартості виконання проектів.

ВИСНОВКИ

За результатами проведеного дослідження можна зробити такі висновки:

1) розв'язано задачу планування послідовності виконання проектів на основі самонавчального стохастичного ігрового методу розфарбовування графа;

2) завдяки своїм адаптивним властивостям, стохастичний ігровий метод можна застосувати для розфарбовування випадкового графа, який враховує імовірності участі агентів у виконанні проектів;

3) результатом навчання стохастичної гри є асимптотично правильно розфарбований випадковий граф, який дозволяє визначити послідовність призначення персоналу для виконання проектів;

4) збіжність стохастичного ігрового методу забезпечується збалансованим співвідношенням його параметрів при дотриманні фундаментальних обмежень стохастичної апроксимації;

5) зростання порядку графа, зв'язності графа, імовірностей відмов агентів призводять до зростання кількості кроків, необхідних для збіжності стохастичної гри розфарбовування графа;

6) метод стохастичної гри розфарбовування графа можна застосувати для розв'язування подібних задач, що формулюються в умовах неповної інформації, наприклад, складання різноманітних розкладів, розпаралелювання алгоритмів, класифікації, кластеризації даних та інших;

7) розглянута стохастична гра має також самостійне значення як модель глобальної самоорганізації станів розподіленої системи, проявом якої є правильне розфарбування випадкового графа в умовах невизначеності на основі опрацювання локально зібраних даних.

ПОДЯКИ

Роботу виконано в рамках держбюджетних тем «Методи та засоби функціонування систем підтримки прийняття рішень на основі онтологій» (ID:839 2017-05-15 09:20:01 (2459-315)) та «Система підтримки прийняття рішень розпізнавання мультиспектральних образів на основі технологій машинного навчання та онтологічного підходу» (ID:0120U102203). Дослідження провадилось в межах спільних наукових досліджень кафедри інформаційних систем та мереж НУ «Львівська політехніка» на тему «Дослідження, розроблення і впровадження інтелектуальних розподілених інформаційних технологій та систем на основі ресурсів баз даних, сховищ даних, просторів даних та знань з метою прискорення процесів формування сучасного інформаційного суспільства». Наукові дослідження провадилися також в рамках ініціативної тематики досліджень кафедри ІСМ НУ «Львівська політехніка» на тему «Розроблення

інтелектуальних розподілених систем на основі онтологічного підходу з метою інтеграції інформаційних ресурсів».

ЛІТЕРАТУРА / ЛИТЕРАТУРА

1. Virtual Communities: Concepts, Methodologies, Tools and Applications / Information Resources Management Association (USA). – Vol. 1–4. – Hershey : IGI Global, 2011. – 2930 p. DOI: 10.4018/978-1-60960-100-3.
2. Heagney J. Fundamentals of Project Management / J. Heagney. – HarperCollins Focus, 2018. – 240 p.
3. Asdre K. The harmonious coloring problem is NP-complete for interval and permutation graphs / K. Asdre, K. Ioannidou, S. D. Nikolopoulos // Discret Applied Mathematics. – 2007. – Vol. 155. – P. 2377–2382.
4. NP-completeness of local colorings of graphs / Z. Li, E. Zhu, Z. Shao, J. Xu // Information Processing Letters. – 2018. – Vol. 130. – P. 25–29.
5. Keet C. M. An Introduction to Ontology Engineering, v.1.5 / C. M. Keet. – University of Cape Town, South Africa. – 2020. – 306 p. – Access mode: <https://people.cs.uct.ac.za/~mkeet/0Ebook/>.
6. Кравець П. О. Ігрова модель онтологічної підтримки проектів / П. О. Кравець, В. В. Литвин, В. А. Висоцька // Radio Electronics, Computer Science, Control. – 2021. – Vol. 1, No. 1. – P. 172–183. DOI: 10.15588/1607-3274-2021-1-17.
7. Chen B.-S. Stochastic Game Strategies and their Applications / B.-S. Chen. – CRC Press, 2019. – 610 p.
8. Panik M. J. Linear Programming and Resource Allocation Modeling / M. J. Panik. – Wiley, 2018. – 448 p.
9. Markis S. Allocation of Manufacturing Tasks to Humans and Robots / S. Markis // Cooperating Robots for Flexible Manufacturing. – Springer, 2021. – P. 373–380.
10. Wang Z. Machine learning-based intermittent equipment scheduling model for flexible production process / Z. Wang, Y. Man // Application of Artificial Intelligence in Process Systems Engineering. – 2021. – P. 473–495.
11. Yang G. Resource allocation algorithm and job scheduling of virtual manufacturing workshop / G. Yang, S. Chen // Academic Journal of Manufacturing Engineering. – 2020. – Vol. 18, No. 2. – P. 155–161.
12. Yeganeh F.T. A multi-objective optimization approach to project scheduling with resiliency criteria under uncertain activity duration / F. T. Yeganeh, S. H. Zegordi // Annals of Operations Research. – 2020. – Vol. 285, No. 1. – P. 161–196.
13. Liu W. Production scheduling and equipment matching of flexible workshops based on multi-objective and multi-process hybrid optimization algorithm / W. Liu // Academic Journal of Manufacturing Engineering. – 2020. – Vol. 18, No. 4. – P. 158–163.
14. Катренко А. В. Дослідження операцій : підручник / А. В. Катренко. – Львів : Магнолія Плюс, 2006. – 549 p.
15. The Reseach of Task Assignment Based on Ant Colony Algorithm / [Z. Wang, S. Li, Y. Wang, S. Li] // International Conference on Mechatronics and Automation. – IEEE Publisher, 2009. – P. 2334–2339. DOI: 10.1109/ICMA.2009.5246570.
16. Sahu A. Solving the Assignment Problem using Genetic Algorithm and Simulated Annealing / A. Sahu, R. Tapadar // IAENG International Journal of Applied Mathematics, IJAM_36_1_7. – 2007. – Vol. 36:1. – P. 1–4.
17. Zais M. A graph coloring approach to the deployment shedaling and unit assignment problem / M. Zais, M. Laguna // Journal of Sheduling. – 2016. – Vol. 19. – P. 73–90.
18. Zhu A. A neural network approach to dynamic task assignment of multirobots / A. Zhu, S. X. Yang // IEEE transactions on neural networks. – 2006. – Vol. 17, No. 5. – P. 1278–1287.

19. Wu S. S. Heuristic algorithm for task assignment and scheduling in a processor network / S. S. Wu, D. Sweeting // *Parallel Computing*. – 1994. – Vol. 20, Issue 1. – P. 1–14.
20. Chartrand G. Chromatic Graph Theory / G. Chartrand, P. Zhang. – Chapman and Hall/CRC, 2019. – 525 p.
21. Saoub K. R. Graph Theory. An Introduction to Proofs, Algorithms, and Applications / K. R. Saoub. – Chapman and Hall/CRC, 2021. – 437 p.
22. Monasson R. On the Analysis of Backtrack Procedures for the Colouring of Random Graphs / R. Monasson // *Lect. Notes Phys.* – 2004. – Vol. 650. – P. 235–254.
23. Christofides N. Graph theory: an algorithmic approach / N. Christofides. – New York : Academic Press, 1975. – 400 p.
24. Bincy A. K. Graph Coloring and its Real Time Applications an Overview / A. K. Bincy, B. J. Presitha // *International Journal of Mathematics And its Applications*. – 2017. – Vol. 5, Issue 4-F. – P. 845–849.
25. Denysenko O. Overview of graph coloring methods and algorithms / O. Denysenko // *Міжнародний мультидисциплінарний науковий журнал «Л'ОГОС. Мистецтво наукової думки»*. – 2019. – No. 7. – P. 27–32.
26. Lima A. M. Exact Algorithms for the Graph Coloring Problem / A. M. Lima, R. Carmo // *Revista de Informatica Teórica e Aplicada*. – RITA. – 2018. – Vol. 25, No. 04. – P. 57–73.
27. Gupta S. Greedy Graph Coloring Algorithm Based on Depth First Search / S. Gupta, D. P. Singh // *International Journal on Emerging Technologies*. – 2020. – Vol. 11, No. 2. – P. 854–862.
28. A genetic algorithm for total graph coloring / [A. Dey, A. Agarwal, P. Dixit et al.] // *Journal of Intelligent & Fuzzy Systems*. – 2019. – Vol. 37, No. 6. – P. 7831–7838.
29. Philipsen W. J. M. Graph coloring using neural networks / W. J. M. Philipsen, L. Stok // *IEEE International Symposium on Circuits and Systems*. – 1991. – P. 1597–1600.
30. A chaotic binary salp swarm algorithm for solving the graph coloring problem / [Y. Meraihi, A. Ramdane-Cherif, M. Mahseur, D. Achelia] // *International Symposium on Modelling and Implementation of Complex Systems*. – 2018. – P. 106–118.
31. Dowsland K. A. An improved ant colony optimization heuristic for graph colouring / K. A. Dowsland, J. M. Thompson // *Discrete Applied Mathematics*. – 2008. – Vol. 156, No. 3. – P. 313–324.
32. Blum A. Multiagent Graph Coloring: Pareto Efficiency, Fairness and Individual Rationality / A. Blum, J. S. Rosenschein // *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence*. – 2008. – P. 24–29.
33. Panagopoulou P. N. A Game Theoretic Approach for Efficient Graph Coloring / P. N. Panagopoulou, P. G. Spirakis // In: Hong S. H., Nagamochi H., Fukunada T. (eds). *Algorithm and Computation. ISAAC 2008. Lecture Notes in Computer Science*. – Springer, Berlin, Heidelberg, 2008. – Vol. 5369. DOI: 10.1007/978-3-540-92182-0_19.
34. Кравець П. О. Ігрова модель хроматичного розфарбовування графів / П. О. Кравець // *Комп'ютерні системи проектування. Теорія і практика: Вісник НУ «Львівська політехніка»*. – 2008. – № 626. – С. 63–74.
35. Райгородский А. М. Модели случайных графов / А. М. Райгородский. – Москва : МЦНМО, 2011. – 136 с.
36. Frieze A. Introduction to random graphs / A. Frieze, M. Karoński. – Cambridge University Press, 2016. – 478 p.
37. Ungureanu V. Pareto-Nash-Stackelberg Game and Control Theory: Intelligent Paradigms and Applications / V. Ungureanu. – Springer, 2018. – 343 p.
38. Назин А. В. Адаптивный выбор вариантов: Рекуррентные алгоритмы / А. В. Назин, А. С. Позняк. – Москва : Наука, 1986. – 288 с.
39. Kushner H. Stochastic Approximation and Recursive Algorithms and Applications / H. Kushner, G. G. Yin. – Springer Science & Business Media, 2013. – 417 p.
40. Benveniste A. Adaptive Algorithms and Stochastic Approximations / A. Benveniste, M. Metivier, P. Priouret. – Springer Science & Business Media, 2012. – 365 p.
41. Neogy S. K. Mathematical Programming and Game Theory / S. K. Neogy, R. V. Vapat, D. Dubey. – Springer, 2018. – 226 p.
42. Кравець П. О. Ігрова самоорганізація системи агентів з індивідуальним оцінюванням стратегій / П. О. Кравець // *Комп'ютерні системи та мережі: Вісник НУ «Львівська політехніка»*. – 2005. – № 546. – С. 75–85.
43. Білова Т. Г. Метод оцінки ступеню структурної близькості зв'язних неорієнтованих графів / Т. Г. Білова, І. О. Побіженко // *Обробка інформації в складних технічних системах*. – 2017. – Вип. 1, № 47. – С. 9–12. – DOI: 10.30748/soi.2017.147.02.

Стаття надійшла до редакції 11.10.2021.
Після доробки 08.12.2021.

УДК 004.[852+94]; 519.837.3

МОДЕЛИРОВАНИЕ ИГРОВОЙ ЗАДАЧИ НАЗНАЧЕНИЯ ПЕРСОНАЛА ДЛЯ ВЫПОЛНЕНИЯ ИТ-ПРОЕКТОВ НА ОСНОВЕ ОНТОЛОГИЙ

Кравець П. А. – канд. техн. наук, доцент, доцент кафедры «Информационные системы и сети», Национальный университет «Львовская политехника», Украина.

Литвин В. В. – д-р техн. наук, профессор, заведующий кафедры «Информационные системы и сети», Национальный университет «Львовская политехника», Украина.

Висоцька В. А. – канд. техн. наук, доцент, доцент кафедры «Информационные системы и сети», Национальный университет «Львовская политехника», Украина.

АННОТАЦИЯ

Актуальность. В этой статье описано решение игровой задачи назначения персонала для работы над проектами на основе онтологического подхода. Суть задачи состоит в следующем. Существует потребность в создании команд для выполнения нескольких проектов. Каждый проект задается набором необходимых онтологических знаний. Для выполнения проектов менеджеры привлекают квалифицированных специалистов (агентов), способности которых также задаются наборами онтологий. Состав команд должен быть таким, чтобы объединенные онтологии их агентов покрывали множества онтологий соответствующих проектов. Каждый агент с определенными вероятностями может принять последовательное участие в выполнении нескольких проектов. Одновременная работа агента над различными проектами не допускается. Необходимо определить порядок выполнения проектов и соответствующий ему порядок назначения персонала.

Целью исследования является разработка математической модели стохастической игры, рекуррентных марковских методов для ее решения, алгоритмического и программного обеспечения, проведение компьютерного эксперимента, анализ результатов и выработкой рекомендаций по их практическому применению.

Метод. Для планирования выполнения проектов использовано стохастический игровой алгоритм раскраски неориентированного случайного графа. Для этого количество вершин графа принято равным количеству проектов. Ребрами соединены те вершины графа проектов, для выполнения которых привлечено одного и того же агента. С учетом восстановительных отказов агентов связи между вершинами графа динамически изменяются. Необходимо достичь правильной раскраски случайного графа. Тогда проекты с одинаково раскрашенными вершинами графа могут быть выполнены параллельно, а проекты с различными цветами вершин – последовательно.

Результаты. В статье построена математическая модель стохастической игры и обучаемый марковский метод для ее решения. Каждая вершина графа контролируется игроком. Чистыми стратегиями игрока являются элементы палитры цветов. После выбора цвета своей вершины каждый игрок вычисляет текущий проигрыш как относительное количество одинаковых цветов в локальном множестве соседних игроков. Цель игроков заключается в минимизации функций средних проигрышей. Марковский рекуррентный метод обеспечивает адаптивный выбор цветов вершин случайного графа на основе динамических векторов смешанных стратегий, значения которых зависят от текущих проигрышей игроков. Результатом стохастической игры является асимптотически правильно раскрашенный случайный граф, когда каждому ребру начального детерминированного графа будут соответствовать в среднем разные цвета вершин.

Выводы. Проведен компьютерный эксперимент, который подтвердил сходимость стохастической игры для задачи раскраски случайного графа. Это дало возможность определить порядок назначения персонала для выполнения проектов.

КЛЮЧЕВЫЕ СЛОВА: проект, агент, онтология, назначение персонала, раскраска случайного графа, стохастическая игра, марковский рекуррентный метод, адаптация, самообучение.

UDC 004.[852+94]; 519.837.3

MODELLING GAME TASK OF ASSIGNING STAFF TO PERFORM IT-PROJECTS BASED ON ONTOLOGIES

Kravets P. – PhD, Associate Professor of Information Systems and Networks Department, Lviv Polytechnic National University, Lviv, Ukraine.

Lytvyn V. – Dr. Sc., Professor, Head of Information Systems and Networks Department, Lviv Polytechnic National University, Lviv, Ukraine.

Vysotska V. – PhD, Associate Professor of Information Systems and Networks Department, Lviv Polytechnic National University, Lviv, Ukraine.

ABSTRACT

Context. This article describes how to solve the game problem of assigning staff to work on projects based on an ontological approach. The essence of the problem is this. There is a need to create teams to carry out several projects. Each project is defined by a set of necessary ontological knowledge. To implement projects, managers invite qualified specialists (agents), whose abilities are also defined by sets of ontologies. The composition of the teams should be such that the combined ontologies of their agents cover the set of ontologies of the respective projects. Each agent with a certain probability can take part in the implementation of several projects. Simultaneous work of the agent on different projects is not allowed. It is necessary to determine the order of project implementation and the corresponding order of personnel appointment.

Objective of the study is to develop a mathematical model of stochastic game, recurrent Markov methods for its solution, algorithmic and software, computer experiment, analysis of results and development of recommendations for their practical application.

Method. A stochastic game algorithm for coloring an undirected random graph was used to plan project execution. To do this, the number of vertices of the graph is taken equal to the number of projects. The edges of the project graph for which the same agent is invited are connected by edges. Due to the recovery failures of agents, the connections between the vertices of the graph change dynamically. It is necessary to achieve the correct coloring of the random graph. Then projects with the same colored vertices of the graph can be executed in parallel, and projects with different colors of vertices – in series.

Results. The article builds a mathematical model of a stochastic game and a self-learning Markov method for its solution. Each vertex of the graph is controlled by the player. The player's pure strategies are the elements of the color palette. After selecting the color of their own top, each player calculates the current loss as a relative number of identical colors in the local set of neighboring players. The goal of the players is to minimize the functions of average losses. The Markov recurrent method provides an adaptive choice of colors for the vertices of a random graph based on dynamic vectors of mixed strategies, the values of which depend on the current losses of players. The result of a stochastic game is an asymptotically correctly colored random graph, when each edge of the initial deterministic graph will correspond on average to different colors of vertices.

Conclusions. A computer experiment was performed, which confirmed the convergence of the stochastic game for the problem of coloring a random graph. This made it possible to determine the procedure for appointing staff to implement projects.

KEYWORDS: project, agent, ontology, staff assigning, coloring of random graph, stochastic game, Markovian recursive method, adaptation, self-learning.

REFERENCES

1. Virtual Communities: Concepts, Methodologies, Tools and Applications, *Information Resources Management Association (USA)*, Vol. 1–4. Hershey, IGI Global, 2011, 2930 p. DOI: 10.4018/978-1-60960-100-3.
2. Heagney J. *Fundamentals of Project Management*. HarperCollins Focus, 2018, 240 p.
3. Asdre K., Ioannidou K., Nikolopoulos S. D. The harmonious coloring problem is NP-complete for interval and permutation graphs, *Discret Applied Mathematics*, 2007, Vol. 155, pp. 2377–2382.
4. Li Z., Zhu E., Shao Z., Xu J. NP-completeness of local colorings of graphs, *Information Processing Letters*, 2018, Vol. 130, pp. 25–29.

5. Keet C. M. An Introduction to Ontology Engineering, v.1.5. University of Cape Town, South Africa, 2020, 306 p. Access mode: <https://people.cs.uct.ac.za/~mkeet/0Ebook/>.
6. Kravets P., Lytvyn V., Vysotska V. Game model of ontological project support (in ukrainian), *Radio Electronics, Computer Science, Control*, 2021, Vol. 1, No. 1, pp. 172–183. DOI: 10.15588/1607-3274-2021-1-17.
7. Chen B.-S. Stochastic Game Strategies and their Applications. CRC Press, 2019, 610 p.
8. Panik M. J. Linear Programming and Resource Allocation Modeling. Wiley, 2018, 448 p.
9. Markis S. Allocation of Manufacturing Tasks to Humans and Robots, *Cooperating Robots for Flexible Manufacturing*. Springer, 2021, pp. 373–380.
10. Wang Z., Man Y. Machine learning-based intermittent equipment scheduling model for flexible production process, *Application of Artificial Intelligence in Process Systems Engineering*, 2021, pp. 473–495.
11. Yang G., Chen S. Resource allocation algorithm and job scheduling of virtual manufacturing workshop, *Academic Journal of Manufacturing Engineering*, 2020, Vol. 18, No. 2, pp. 155–161.
12. Yeganeh F. T., Zegordi S. H. A multi-objective optimization approach to project scheduling with resiliency criteria under uncertain activity duration, *Annals of Operations Research*, 2020, Vol. 285, No. 1, pp. 161–196.
13. Liu W. Production scheduling and equipment matching of flexible workshops based on multi-objective and multi-process hybrid optimization algorithm, *Academic Journal of Manufacturing Engineering*, 2020, Vol. 18, No. 4, pp. 158–163.
14. Katrenko A.V. Operations research. Manual (in ukrainian). Lviv, Magnolia Plus, 2006, 549 p.
15. Wang Z., Li S., Wang Y., Li S. The Research of Task Assignment Based on Ant Colony Algorithm, *International Conference on Mechatronics and Automation, IEEE Publisher*, 2009, pp. 2334–2339. DOI: 10.1109/ICMA.2009.5246570.
16. Sahu A., Tapadar R. Solving the Assignment Problem using Genetic Algorithm and Simulated Annealing, *IAENG International Journal of Applied Mathematics, IJAM_36_1_7*, 2007, Vol. 36:1, pp. 1–4.
17. Zais M., Laguna M. A graph coloring approach to the deployment scheduling and unit assignment problem, *Journal of Scheduling*, 2016, Vol. 19, pp. 73–90.
18. Zhu A., Yang S. X. A neural network approach to dynamic task assignment of multirobots, *IEEE transactions on neural networks*, 2006, Vol. 17, No. 5, pp. 1278–1287.
19. Wu S. S., Sweeting D. Heuristic algorithm for task assignment and scheduling in a processor network, *Parallel Computing*, 1994, Vol. 20, Issue 1, pp. 1–14.
20. Chartrand G., Zhang P. Chromatic Graph Theory. Chapman and Hall/CRC, 2019, 525 p.
21. Saoub K. R. Graph Theory. An Introduction to Proofs, Algorithms, and Applications. Chapman and Hall/CRC, 2021, 437 p.
22. Monasson R. On the Analysis of Backtrack Procedures for the Colouring of Random Graphs, *Lect. Notes Phys*, 2004, Vol. 650, pp. 235–254.
23. Christofides N. Graph theory: an algorithmic approach / N. Christofides. New York, Academic Press, 1975, 400 p.
24. Bincy A. K., Presitha B. J. Graph Coloring and its Real Time Applications an Overview, *International Journal of Mathematics And its Applications*, 2017, Vol. 5, Issue 4-F, pp. 845–849.
25. Denysenko O. Overview of graph coloring methods and algorithms, *International Multidisciplinary Science Journal* «Λ'ΟΓΟΣ». *The art of scientific thought*, 2019, No. 7, pp. 27–32.
26. Lima A. M., Carmo R. Exact Algorithms for the Graph Coloring Problem, *Revista de Informatica Teórica e Aplicada, RITA*, 2018, Vol. 25, No. 04, pp. 57–73.
27. Gupta S., Singh D. P. Greedy Graph Coloring Algorithm Based on Depth First Search, *International Journal on Emerging Technologies*, 2020, Vol. 11, No. 2, pp. 854–862.
28. Dey A., Agarwal A., Dixit P., Long H. V., Werner F., Pal T., Son L. H. A genetic algorithm for total graph coloring, *Journal of Intelligent & Fuzzy Systems*, 2019, Vol. 37, No. 6, pp. 7831–7838.
29. Philipsen W. J. M., Stok L. Graph coloring using neural networks, *IEEE International Symposium on Circuits and Systems*, 1991, pp. 1597–1600.
30. Meraihi Y., Ramdane-Cherif A., Mahseur M., Achelia D. A chaotic binary salp swarm algorithm for solving the graph coloring problem, *International Symposium on Modelling and Implementation of Complex Systems*, 2018, pp. 106–118.
31. Dowland K. A., Thompson J. M. An improved ant colony optimization heuristic for graph colouring, *Discrete Applied Mathematics*, 2008, Vol. 156, No. 3, pp. 313–324.
32. Blum A., Rosenschein J. S. Multiagent Graph Coloring: Pareto Efficiency, Fairness and Individual Rationality, *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence*, 2008, pp. 24–29.
33. Panagopoulou P. N., Spirakis P.G. A Game Theoretic Approach for Efficient Graph Coloring, In: Hong S.H., Nagamochi H., Fukunada T. (eds). *Algorithm and Computation. ISAAC 2008. Lecture Notes in Computer Science*. Springer, Berlin, Heidelberg, 2008, Vol. 5369, DOI: 10.1007/978-3-540-92182-0_19.
34. Kravets P. O. Game model of chromatic coloring of graphs (in ukrainian), *Computer Design Systems. Theory and Practice: Bulletin of the National University of Lviv Polytechnic*, 2008, No. 626, pp. 63–74.
35. Raigorodskii A. M. Models of random graphs (in russian). Moscow, Moscow Center for Continuous Mathematical Education (MCCME), 2011, 136 p.
36. Frieze A., Karoński M. Introduction to random graphs. Cambridge University Press, 2016, 478 p.
37. Ungureanu V. Pareto-Nash-Stackelberg Game and Control Theory: Intelligent Paradigms and Applications. Springer, 2018, 343 p.
38. Nazin A. V., Poznyak A. S. Adaptive Choice of Variants: Recurrence Algorithms (in russian). Moscow, Science, 1986, 288 p.
39. Kushner H., Yin G. G. Stochastic Approximation and Recursive Algorithms and Applications. Springer Science & Business Media, 2013, 417 p.
40. Benveniste A., Metivier M., Priouret P. Adaptive Algorithms and Stochastic Approximations. Springer Science & Business Media, 2012, 365 p.
41. Neogy S. K., Bapat R. B., Dubey D. Mathematical Programming and Game Theory. Springer, 2018, 226 p.
42. Kravets P. A. Game self-organization of agents system with individual estimation of strategies (in ukrainian), *Computer systems and networks: Bulletin of the National University of Lviv Polytechnic*, 2005, No. 546, pp. 75–85.
43. Bilova T. G., Pobezhenko I. A. Method of estimation the degree of structural proximity of bound non-oriented graphs (in ukrainian), *Processing of information in complex technical systems*, 2017, Issue 1, No. 47, pp. 9–12. DOI: 10.30748/soi.2017.147.02.

THE RELIABILITY IMPROVING OF COMPUTER SYSTEM ELEMENTS WITH USING MODULAR ENCODING

Freyman V. I. – Dr. Sc., Professor of Department “Automatics and Telemechanics”, Perm National Research Polytechnic University, Perm, Russia.

ABSTRACT

Context. Computing systems are implemented in many industries and economies of the modern world. The quality indicators of the systems in which they are used depend on the reliability of their work. The reliability of a computing system consists of the reliability of the construction and functioning of its elements. It is not always possible to ensure reliability in the design by choosing a high-quality element base, structural redundancy, or other well-known methods. Therefore, important and critical elements of computing systems are protected by built-in control schemes. They allow you to detect errors that occur when performing basic data operations. An effective way of constructing such circuits is to use actions on the remainder of the division of the operands by a selected module or by several modules (modular coding). Especially the task of choosing the most accurate and least redundant means of control is relevant for a wide range of basic elements of modern computing systems.

Objective. The aim of the work is research and development of recommendations on the use of modular coding to improve the reliability of the functioning of elements of modern computing systems in various hardware and software basis.

Methods. A method for numerical control of the correctness of performing basic arithmetic and logical operations by computing devices is selected and analyzed. On its basis, a schematic model of a computing system was built and verified in the MatLab Simulink environment, which uses modular coding as a means of ensuring the reliability of the functioning of elements. The analysis of the probabilistic characteristics of decision-making is carried out, estimates of the probability of an erroneous decision-making are given. A software implementation of the simulation algorithm in the Visual Basic for Applications environment has been created, which made it possible to plot the dependence of reliability indicators on coding parameters.

Results. A schematic model of a computing system has been developed. It allows study various combinations of faults in the functioning of elements and errors in their operations. An algorithm for simulating all kinds of malfunctions and errors in the functioning of elements of computing systems when they perform basic operations is implemented in software. The qualitative dependences of the probabilistic characteristics of reliability on the coding parameters are determined. Based on the analysis of the characteristics obtained, conclusions are drawn and practical recommendations are given on the use of modular coding in the elements of computing systems in order to achieve the specified reliability indicators.

Conclusions. To improve the reliability of the functioning of the elements of computing systems, it is effective to use built-in control schemes using modular coding. Taking into account the recommendations for choosing the parameters of the codes will ensure the required reliability with minimal redundancy of circuits and the computational complexity of the calculation algorithms.

KEYWORDS: computing system, element, reliability, diagnostics, modular coding, module, deduction, reliability indicators, error detection probability, simulation, circuit, built-in control.

ABBREVIATIONS

AES is an Advanced Encryption Standard;
ASIC is an Application Specific Integrated Circuits;
CAS is a Chemical Abstracts Service;
CS is a Computing System;
ECS is an Element of Computing System;
FPGA is a Field-Programmable Gate Array;
IDEA is an International Data Encryption Algorithm;
RCS is a Residual Class System;
RSA is a public key cryptographic algorithm (authors are Rivest, Shamir, Adleman);
SoC is a System on Crystal.

NOMENCLATURE

A is a first operand;
 B is a second operand;
 C is a result of operation;
 f is a total function;
 i is an index of information module;
 j is an index of control module;
 k is a number of control modules;
 k_j is a modulo;
 m is a number of information modules;
 n is a total number of modules;

N is a amount of non-found errors;
 P is a probability of error detection;
 P_c is a working range;
 P_{nf} is an error probability;
 $P(P_{nf}/P_j)$ is a conditional probability of missing an error;
 p is a probability of error at one digit;
 pc is a module for modulo calculation;
 p_j is a value of control module with index j ;
 pk_j is a control module with index j ;
 pm_i is a information module with index i ;
 r_a is a remainder (modulo) after dividing the number A by the module p ;
 r_b is a remainder (modulo) after dividing the number B by the module p ;
 r_c is a remainder (modulo) after dividing the number C by the module p ;
 r'_c is a result of control by the module p ;
 R is a redundancy rate;
 R_1 is a first redundancy rate;
 R_{II} is a second redundancy rate.

INTRODUCTION

CS play an important role in solving specialized problems in many areas of industry and economy of the modern world. Most of the quality indicators of the processes they serve depend on the reliability of their work.

Reliability is determined by the following main components: failure-free operation, maintainability, durability and storability [1]. One of the main ways to ensure reliability is the use of built-in control schemes using modular coding [2].

Modular (arithmetic) codes detect and correct errors arising from the action of noise during the storage and processing of information in the elements of computing systems, as well as during the transfer of data between them. The main problem here is the economical use of redundancy to achieve the required reliability (noise immunity) of data transmission over a noisy channel or when they are stored on some medium. Knowledge of the theory and practice of coding is necessary for developers of both hardware and software of modern computing systems [3].

In the process of calculations, there is a constant transfer and transformation of information in the memory of computing devices. Thus, when designing an ECS, it is necessary to provide for both error detection and correction measures. This function is assigned to the control system. Control system – a set of hardware and software methods and tools that ensure the determination of the correct operation of the device as a whole or its individual units, as well as automatic correction of detected errors. There are the following types of computational errors arising [4]:

- due to errors in the initial data;
- due to methodological errors;
- due to malfunctions in the operation of devices.

Therefore, important and urgent tasks are a reasonable choice of parameters and an effective hardware and software implementation of modular coding methods in ECS.

The object of study is modular coding methodology in ECS control schemes.

The subject of the study is development of methods for assessing the probabilistic characteristics of modular codes and recommendations for their implementation in the selected hardware-software basis.

The purpose of the work is increasing the reliability of computing systems elements with using a modular coding in the built-in control circuits.

1 PROBLEM STATEMENT

The basis of CS is computers (information converters). Let us single out two directions of ensuring reliability – improving reliability during their operation and increasing the reliability of information transfer between them [5].

The first direction is the choice of effective circuit solutions, materials, production technologies, etc. But this does not always help, therefore, when using them in

critical infrastructure facilities (aviation, space, security, energy, etc.), it is necessary to provide additional control of the correct functioning. An effective solution is to control the execution of operations (modulo a prime number or their product – modular coding). This is especially true for the implementation of your own basic elements (with your own production of integrated circuits of various degrees of integration) and in programmable logic devices in a configurable hardware-software basis (FPGA, ASIC, SoC, etc.) [6].

The second direction is redundant coding. Advantages of modular codes: simple and, as a consequence, less resource-intensive (the number of operations, the length of the operands, etc.) encoding and decoding algorithms. This is especially true when implementing ESCs under resource constraints such as fieldbus programmable logic controllers. Also, when the transmission medium is noisy, it is usually required to use complex methods of error-correcting coding (Reed-Solomon codes, convolutional codes, turbo codes and their cascading), which have such complex decoding algorithms that it is difficult and sometimes unrealistic to implement them in ECS [7].

Mathematical formulation of the problem: the length of the informational part of the message m is known. It is necessary to determine the dependence of the probabilistic indicator of reliability on the number and values of control modules:

$$P = f(k, p_j), j \in [1; k]. \quad (1)$$

Also, an important characteristic for choosing the parameters of the modular code is the dependence of the probabilistic indicator of reliability on redundancy:

$$P = f(R). \quad (2)$$

The task is to select the number and values of modules for better reliability (with a fixed redundancy) and less redundancy (with a fixed value of the reliability indicator).

2 REVIEW OF THE LITERATURE

The basis of modular coding is number theory (Fermat, Bernoulli, Leibniz, Euler, Gauss) [8]. It is based on the comparison of two integers modulo a natural number – a mathematical operation that allows you to answer the question of whether two selected integers, when divided by one number, give the same remainders. Arithmetic operations with the remainder of numbers in a fixed modulus form modular arithmetic or modular arithmetic, which is widely used in mathematics, computer science, and cryptography [9].

Work on the use of redundant coding to improve the reliability of computing devices began in the middle of the 20th century. The introduction of redundancy at the hardware and / or software levels has been recognized as the main way to build reliable computing systems [10]. In particular, modular coding is used in many built-in control schemes in the structure of computing devices [11].

Redundant coding is used at all levels of interaction between elements of computing systems. Depending on this, different coding methods are used – both relatively simple (parity check [6], codes for some combinations [12], modular codes [13]), and more complex (binary cyclic codes [14], nonbinary Reed codes -Solomon [15], non-block convolutional codes [16], soft decision codes [17], packet error correction codes [18], etc.). The redundancy and computational complexity of the coding and decoding algorithms are determined based on the requirements for the reliability of the functioning of a particular computing system or its device [19–21].

To determine errors when entering an international bank account number, a comparison modulo 97 is used. This allows you to detect arithmetic errors of a sufficiently high frequency.

In cryptography, comparisons can be found in public-key systems using, for example, the RSA algorithm or the Diffie-Hellman protocol. Also, modular arithmetic provides finite fields, over which elliptic curves are then drawn, and is used in various symmetric key protocols (AES, IDEA).

In chemistry, the last digit in the CAS serial number is the checksum value, which is calculated by adding the last digit of the number multiplied by 1, the second digit from the right multiplied by 2, the third digit multiplied by three, and so on up to the first digit from the left, ending with the remainder of division by 10.

It should be noted that the literature does not describe ways of justifying the choice of a module when controlling logical operations (usually 3 is used in practice, but it is possible that other modules or a composite module are more efficient) [18]. There are no publications on the assessment of probabilistic indicators of reliability when deciding on the frequency and place of arithmetic errors when controlling operations in computing devices.

Also, the results of studies of the application of modular coding to ensure reliability in data transmission, to compare efficiency with other error-correcting codes

are not presented. This explains the relevance of this work.

3 MATERIALS AND METHODS

Consider arithmetic codes using numerical control modulo a prime number, which is widely used in computing systems to control all basic arithmetic and logical operations [3]. These include addition, subtraction, multiplication, logical addition (disjunction), logical multiplication (conjunction), negation of equivalence (addition mod 2), inversion, shift operations of various kinds (arithmetic and logical left-right shift, cyclic shift). A generalized block diagram that implements numerical control in modulus is shown in Fig. 1.

The essence of control is as follows. Let us denote the main controlled operation on operands A and B by $*$, and the residues of operands A and B by r_a and r_b respectively: $A \equiv r_a \pmod{pc}$; $B \equiv r_b \pmod{pc}$.

The process of forming a deduction (remainder) is called the convolution of a number. In parallel with the main operation $*$ on operands A and B a certain operation is performed in the controlling device # on the operands r_a and r_b . Then the result of the operation C is collapsing, and value r_c is compares with result r'_c received in the control device. The equality of r_c and r'_c indicates the correct performance of the operation.

Taking into account that the bit depth of the deduction is much less than the dimension of the operand, it can be expected that the total complexity of the controlling device and convolution devices will be less than the complexity of the device duplicating the main (controlled) device.

The objects of control are devices that implement the above operations: adders, counters, shifters, arithmetic devices, etc.

Let us give a number of mathematical operations on residues in the numerical method of controlling the basic operations for converting information in computing devices.

This type of control is based on the use of identities known from the theory of numbers for the comparability

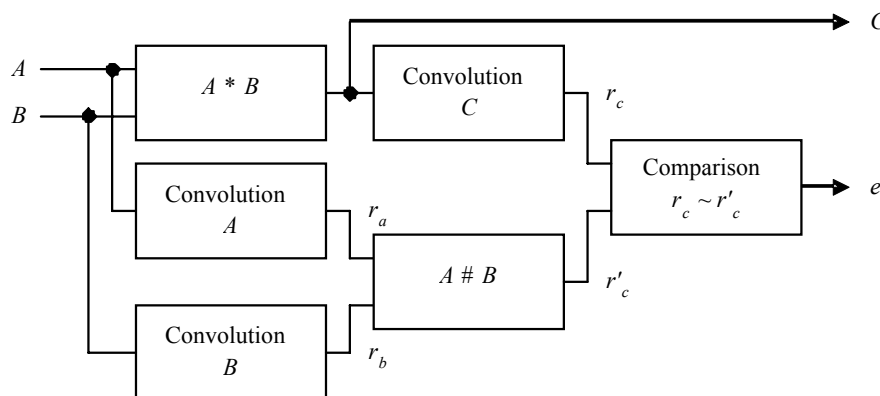


Figure 1 – Element structure of the developed test diagnostics system

of the sum (product) of numbers with the sum (product) of the remainders of the same numbers by some modulus:

$$\sum_{i=1}^n A_i \equiv \sum_{i=1}^n r_{a_i} \pmod{pc}; \quad (3)$$

$$\prod_{i=1}^n A_i \equiv \prod_{i=1}^n r_{a_i} \pmod{pc}. \quad (4)$$

There are some formulas for control the basic arithmetical and logical operations of operands A and B and the result C :

$$\text{addition: } r'_c \equiv (r_a + r_b) \pmod{pc}; \quad (5)$$

$$\text{multiplication: } r'_c \equiv (r_a \cdot r_b) \pmod{pc}; \quad (6)$$

$$\text{disjunction: } r'_c \equiv (r_a + r_b - r_{(A \& B)}) \pmod{pc}; \quad (7)$$

$$\text{conjunction: } r'_c \equiv (r_a + r_b - r_{(A+B)}) \pmod{pc}. \quad (8)$$

A more difficult and more accurate control method is the error detection method that works with numbers in RSC. Let us define a series of positive coprime numbers p_1, p_2, \dots, p_n , which we call the bases of the number system in the residual classes. Let $p_1 < p_2 < \dots < p_n$. To represent the number A in the selected RCS we find the set of the smallest residuals r_i ($i = 1, 2, \dots, n$) for $A \equiv r_i \pmod{p_i}$.

Thus, each number will be uniquely represented by its own set of residuals: $A = (r_1, r_2, \dots, r_{n-1}, r_n)$. The range of positive integers at RCS is limited to:

$$Pc = p_1 p_2 \dots p_n. \quad (9)$$

The range of numbers $[0, Pc-1]$ will be called "working". All operands and results within this range will be called «correct» and outside of it will be called "incorrect".

For the code provided in RCS to be able to detect and correct arithmetic errors, redundancy must be introduced. The main method for constructing redundant RCSs is to expand the original system by adding modules $p_{m+1}, p_{m+2}, \dots, p_{m+k}$ that pairwise simple with each other and with the modules of the original system. Thus, in the representation of a number in RCS, one can distinguish m information modules and k redundant modules, total n modules. This means that a number in RCS can be represented through the totality of residues (modulus) for all declared modules:

$$A = \{r_{m1}, r_{m2}, \dots, r_{mi}, \dots, r_{mm}, r_{k1}, r_{k2}, \dots, r_{kj}, \dots, r_{kk}\}; \quad (10)$$

$$i \in [1; m]; j \in [1; k].$$

Redundancy can be estimated as the ratio of the number of redundant modules to the total number of modules (R_I) or through the bit representation of the lengths of the corresponding parts of the code (R_{II}):

$$R_I = \frac{k}{m+k} = \frac{k}{n}; \quad (11)$$

$$R_{II} = \frac{\sum_{j=1}^k [\log_2 k_j]}{\sum_{i=1}^m [\log_2 m_i] + \sum_{j=1}^k [\log_2 k_j]}. \quad (12)$$

An analytical model [3] was used to determine the reliability characteristics, a schematic model (in the MatLab Simulink package) and a software model (in the Visual Basic for Applications environment) were developed.

Analytical model. We will assume that errors in each of the digits of the number code appear independently of each other and obey the binomial probability distribution. Then the probability of occurrence of errors of the j -th multiplicity in an n -bit number:

$$P(j, n) = P(j) = \binom{n}{j} p^j (1-p)^{n-j}. \quad (13)$$

Determine the probability of not detecting an error:

$$P_{nf} = 1 - P = \sum_{j=1}^n P(j, n) P\left(\frac{P_{nf}}{P_j}\right). \quad (14)$$

This formula is universal, that is, it is valid for any control method.

Schematic model. The block diagram of the model in Matlab Simulink [22], which implements the control of arithmetic operations (multiplication, addition, subtraction), is shown in Fig. 2. The values of the operands A and B are entered, as well as the operating mode of the device. The elements «Display» illustrates the results of the operation and control the correctness of its execution. Distortion can be set in the block to define variants of detectable and undetectable errors.

The block diagram of the model in Matlab Simulink, which implements the control of logical operations (conjunction, disjunction, summation modulo two), is shown in Fig. 3.

To study the properties of modular coding in the system of residual classes a block diagram of the model in MatLab Simulink was developed (Fig. 4). The main components are implemented in the model:

- source of information;
- encoder;
- information transmission channel;
- source of interference;
- decoder;
- the recipient of the information.

In the encoder and decoder (Fig. 5), the number and values of information and redundant modules are set, and the conversion to RCS is also implemented. In the communication channel model, distortions are introduced into the transmitted message. An error detection procedure is implemented in the decoder. It is based on the inverse transformation of the number representation in RCS and checking if it is within the working range.

The constructed models make it possible to carry out experiments on introducing arbitrary distortions and analyzing the detecting ability of a code with given characteristics.

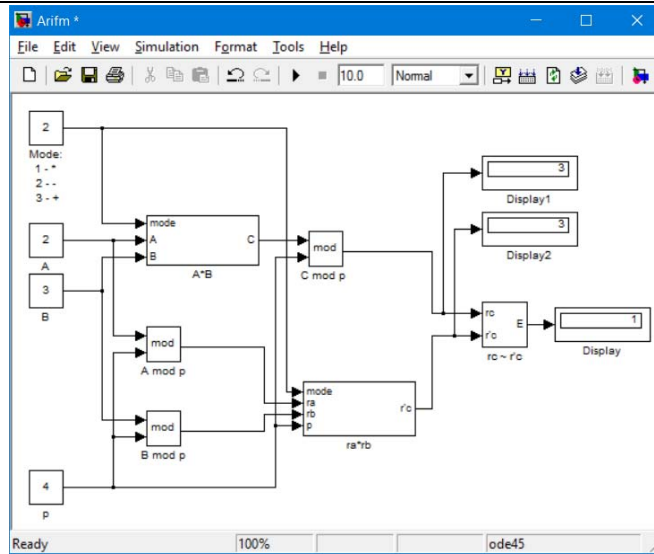


Figure 2 – The model for arithmetical operation control

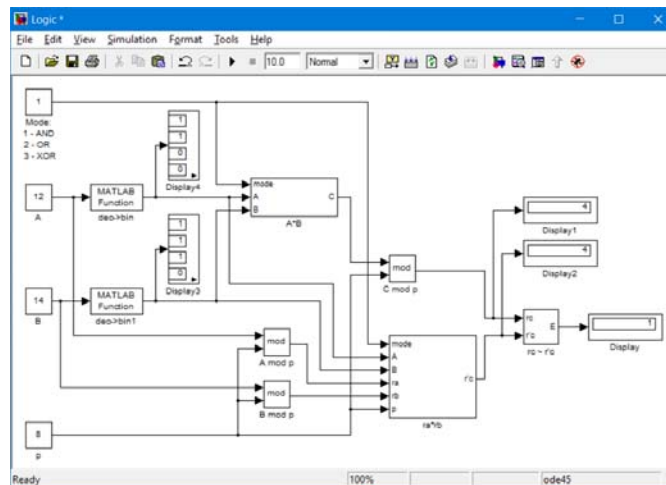


Figure 3 – The model for logical operation control

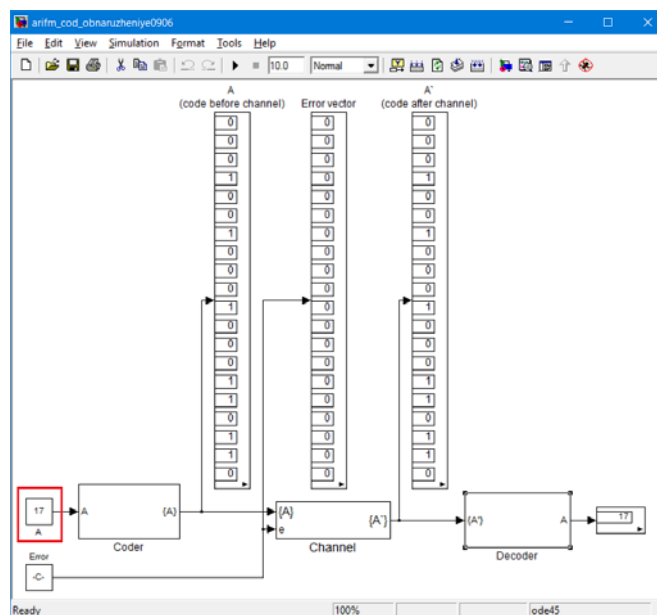


Figure 4 – The model for computing system with encoding with RCS

Programming model. To study the dependences of reliability indicators on code parameters, a program model was developed in the Visual Basic for Applications language for Microsoft Excel (Fig. 6). It allows you to set the type of the checked operation, the required parameters of the code and to carry out simulation modeling of the full set of faults. As a result, the number of detected and undetected errors is calculated. The software simulation algorithm is shown in Fig. 7.

The number of undetected errors $N \in [N_{\min}; N_{\max}]$ (div – whole division operation):

$$N_{\min} = (2^m \text{ div } pc); N_{\max} = (2^m \text{ div } pc) + 1. \quad (15)$$

The probability of undetected errors:

$$P_{nf} = N_{\max} / 2^m = (2^m \text{ div } p + 1) / 2^m. \quad (16)$$

4 EXPERIMENTS

As a result of the study of the analytical model (14), a dependence was built (Fig. 8). When analyzing distortions, we assume that errors of different signs are equally probable and in the i -th bit of the binary positional number are equal $\pm 2^i$.

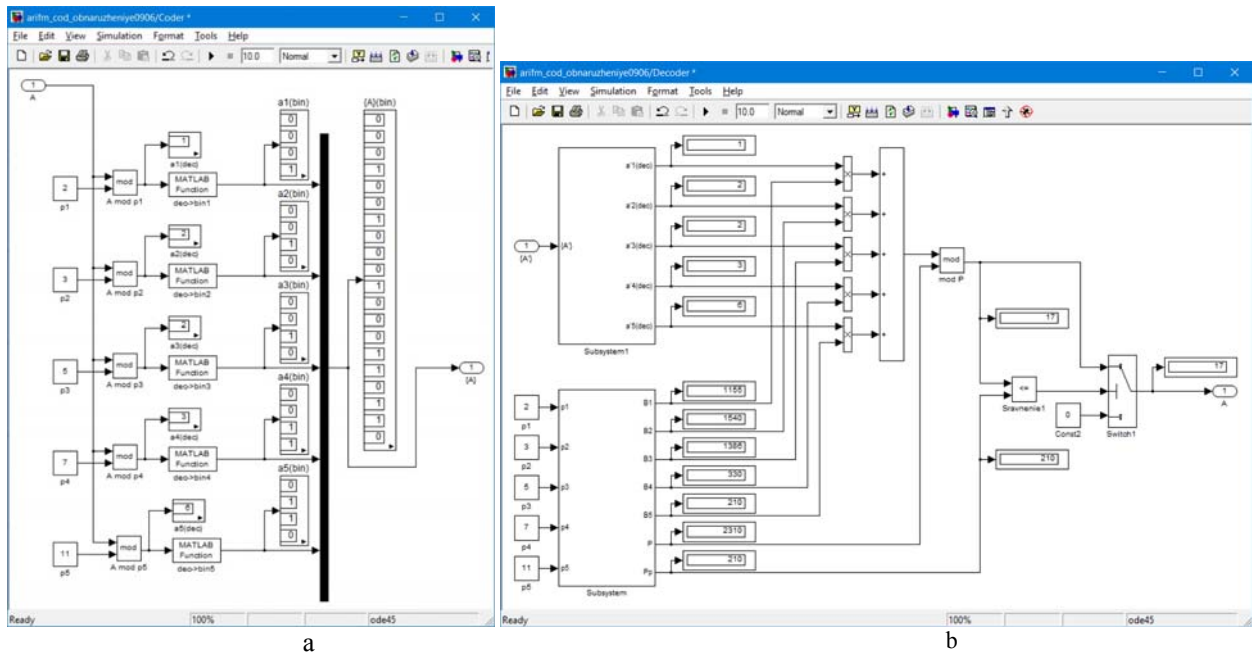


Figure 5 – Encoder (a) and decoder (b) of model RCS

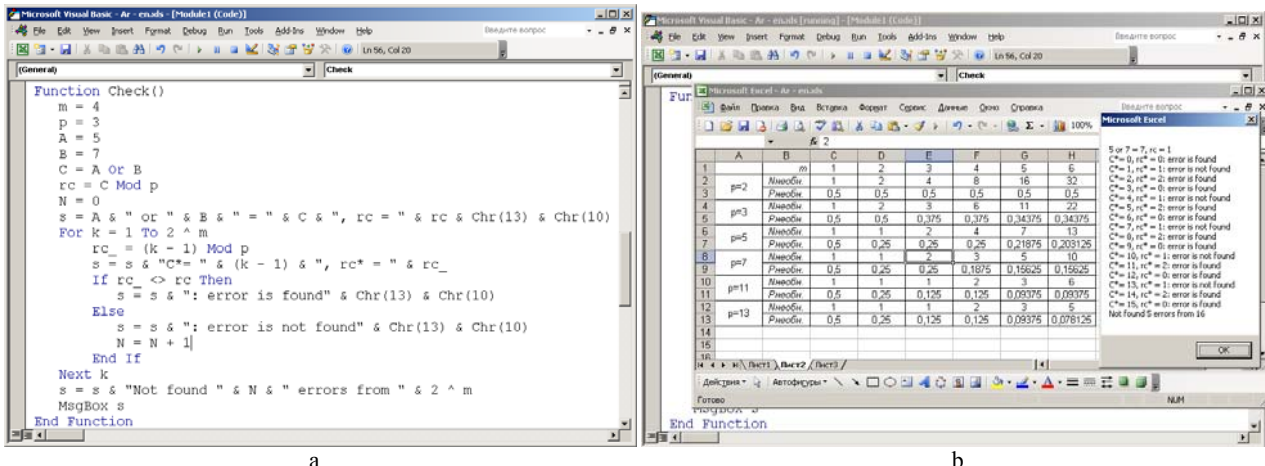


Figure 6 – The main program module (a) and the example of simulation results (b)

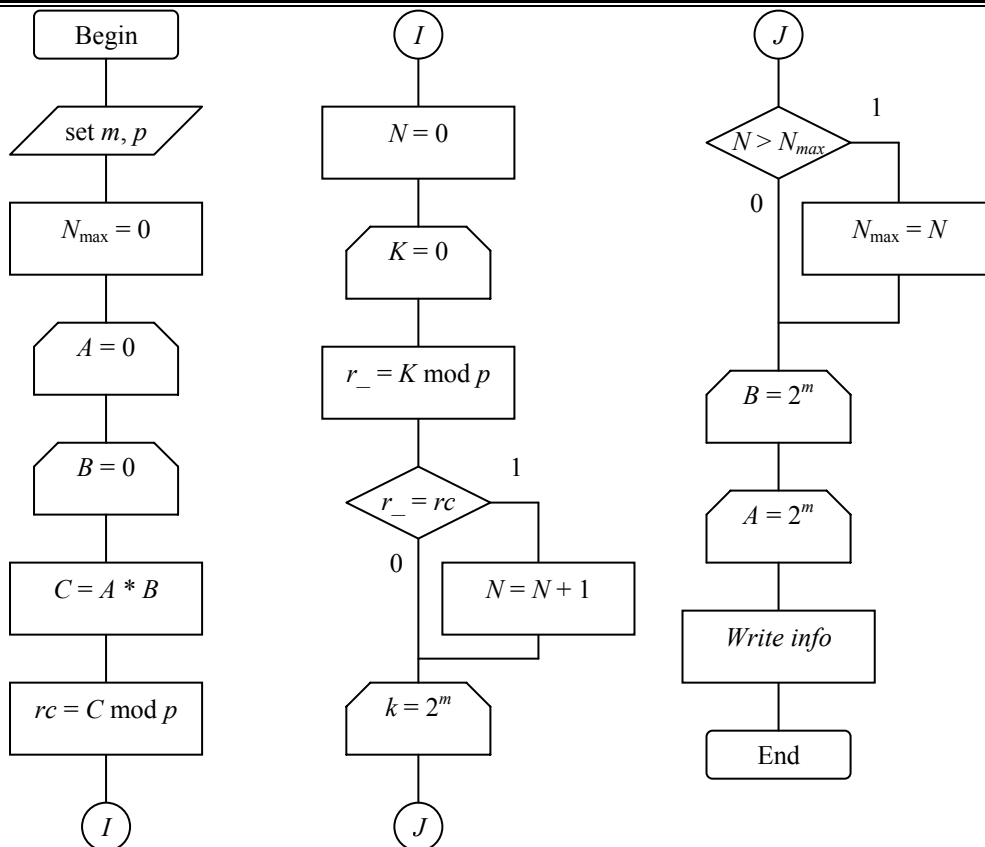


Figure 7 – Algorithm of program simulation

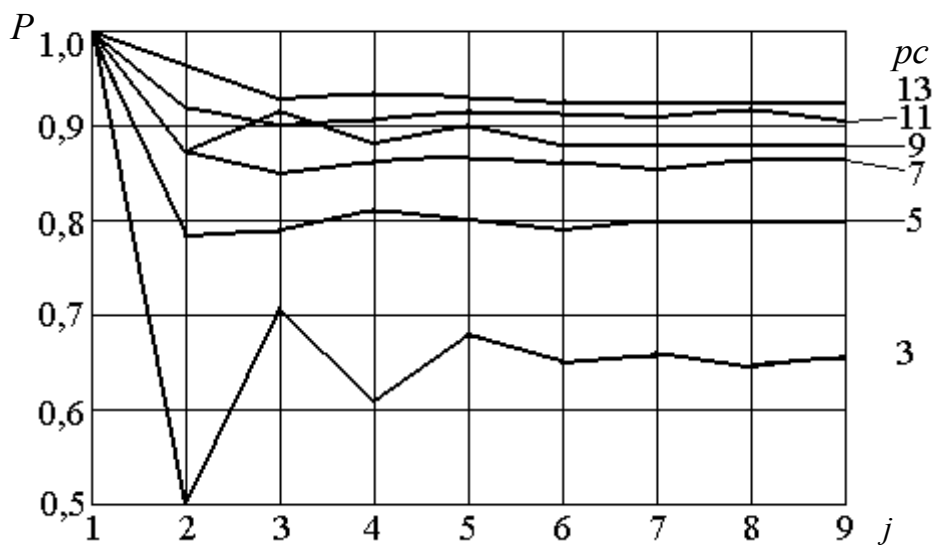


Figure 8 – Dependence of the conditional probability of error detection P on their multiplicity j for different modules pc

Fig. 9 shows the dependences of the error detection probability obtained from the results of the study of the software model:

– on the bit depth of the checked number for different modules p ;

– from the value of the modulus p for different values of the digit capacity of the checked number.

Fig. 10 shows the results of circuit simulation for errors in one and two digits of the number representation in RCS.

CONCLUSIONS

This article presents the results of the application of modular coding in solving problems of increasing the reliability of the functioning of the elements of computing systems. An approach to the implementation of control over the correctness of the execution of arithmetic and logical operations in computing devices has been studied. The application of modular coding to a data transmission channel both inside and between computing devices is considered. Analytical, schematic and software models of the investigated control, coding and decoding processes have been built.

The scientific novelty of the results is: an approach to assessing the dependence of reliability indicators on the control method and coding parameters is proposed. The dependences of the probabilistic characteristics on the coding parameters are constructed. Qualitative conclusions are made about their direction and the results of their influence on the main indicators of reliability.

The practical significance the results of the work consists in recommendations on the use of the obtained dependences and characteristics when choosing the method and parameters of modular coding. This has the prospect of being used both when implemented in computing devices:

- in the hardware basis (at the level of logical elements);
- in the software basis (microcontrollers, microprocessors);
- in hardware and software basis (FPGA).

Prospects for further research are prospects for the use of modular coding in information transmission systems as an effective way to correct transmission errors.

ACKNOWLEDGEMENTS

The presented studies were carried out in continuation of research and development work carried out by the Department of Automatics and Telemechanics together with leading domestic developers of radio electronic equipment (Perm, Russia). The results obtained are intended for the hardware and software implementation of computing devices for reliable interaction of the modules of the designed equipment along intrablock highways, as well as through local and transport channels of information transmission systems. The author expresses deep gratitude to his teacher and long-term work supervisor, professor Efim Lvovich Kon, for introducing him to the scientific topics under consideration.

REFERENCES

1. State Standard 27.002–1989. Industrial product Dependability. General concepts. Terms and definitions. Moscow, Standartinform Publ., 1990. 24 p. (In Russian).
2. Jha N. K., Wang, S. J. Design and synthesis of self-checking VLSI circuits, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2013, Vol. 12 (6), pp. 878–887. DOI: 10.1109/43.229762.
3. Kon E. L., Freyman V. I. The theory of telecommunications. The noise stability data transmission within information and control and telecommunication systems: models, algorithms, structures. Perm, Perm State Technical University, 2007, 317 p.
4. Fink L. M. Signals, noise, errors. Radio and communication, 1984, 256 p.
5. Freyman V., Bezukladnikov I. Research and application of noise stability providing methods at information and control systems, *2017 IEEE Conference of Russian young researchers in electrical and electronic engineering : 1–3 February 2017 : proceedings*. Saint-Petersburg, Saint-Petersburg Electrotechnical University «LETI», 2017, pp. 831–837.
6. Kubalik P., Kubatova H. Parity Codes Used for On-Line Testing in FPGA, *Acta Polytechnica*, 2005, Vol. 45, No. 6, pp. 53–59.
7. Sklar B. Digital communications. Fundamentals and applications : second edition. New Jersey, Prentice Hall, 2001, 1079 p.
8. Hazewinkel M. Encyclopedia of Mathematics. Springer, 2001, 488 p.
9. Chiranjeevi G. N., Kulkarni S. Pipeline Architecture for $N=K*2L$ Bit Modular ALU: Case Study between Current Generation Computing and Vedic Computing, *Convergence in Technology (I2CT), IEEE 6th International Conference, Mumbai, India, 02–04 April 2021*. Mumbai, IEEE, 2021, pp. 1–4. DOI: 10.33130/AJCT.2021v07i01.007.
10. Kumar A. A., Makur A. Improved coding-theoretic and subspace-based decoding algorithms for a wider class of DCT and DST codes, *IEEE Transactions on Signal Processing*, 2010, Vol. 58, Issue 2, pp. 695–708.
11. Zeng H., Li W., Chen T., Nan L. An Efficient Module Arithmetic Logic Unit in Dual Field for Internet of Things Applications, *Application Specific Integrated Circuits (ASIC) : IEEE 14th International Conference, Kunming, China, 26–29 October 2021, proceedings*. Kunming, IEEE, 2021, pp. 1–4. DOI: 10.1109/ASICON52560.2021.9620531.
12. Plantard T. Efficient Word Size Modular Arithmetic, *Computer Arithmetic (ARITH) : IEEE 28th Symposium, Lyngby, Denmark, 14–16 June 2021 : proceedings*. Lyngby, IEEE, 2021, pp. 139–139. DOI: 10.1109/ARITH51176.2021.00036.
13. Ahmadifar H., Jaberipur G. Improved modulo- $(2n \pm 3)$ multipliers, *Computer Architecture and Digital Systems (CADS), 17th CSI International Symposium, Tehran, 30–31 October 2013. Tehran, IEEE, 2013, pp. 31–35. 10.1109/CADS.2013.6714234.*
14. Blahut R. E. Theory and practice of error control codes. Massachusetts, Addison-Wesley Publishing Company Incorporated, 1986, 576 p.
15. Freyman V. Research of the Reed-Solomon codes characteristic for realization within control systems devices, *Radio electronics, Computer science, Control*, 2019, Vol. 3 (50), pp. 143–151. DOI: 10.15588/1607-3274-2019-3-16.
16. Viterbi A. J., Omura J. K. Principles of digital communication and coding. New York, McGraw-Hill, 2009, 584 p.
17. Freyman V. I. Methods and algorithms of soft decoding for signals within information transmission channels between control systems elements, *Radio Electronics, Computer Science, Control*, 2018, Vol. 4, pp. 226–235. DOI: 10.15588/1607–3274–2018–4–22.
18. Freyman V. Analysis of the transmission reliability between control systems devices when errors are packaged, *Radio electronics, Computer science, Control*, 2019, Vol. 4 (51), pp. 223–231. DOI: 10.15588/1607-3274-2019-4-21.
19. Morelos-Zaragoza R. The art of error correcting. Malden, Wiley, 2006, 269 p.
20. Bhargava K. Efficient implementation of error correction coding in a communication system by using VHDL, *VSRD*

International Journal of Electrical, Electronics and Communication Engineering, 2012, Vol. 2 (6), pp. 359–365.
21. Efanov D., Sapozhnikov V., Sapozhnikov V. Two-Modulus Codes with Summation of On-Data Bits for Technical Diagnostics of Discrete Systems, *Automatic Control and*

Computer Sciences, 2018, Vol. 52, pp. 1–12.
DOI:10.3103/S0146411618010029

22. MATLAB Documentation [Electronic resource]. Access mode: <http://www.mathworks.com/help/matlab/>.

Received 06.12.2021.

Accepted 30.12.2021.

УДК 621.391:004.052

ПІДВИЩЕННЯ НАДІЙНОСТІ ЕЛЕМЕНТІВ ОБЧИСЛЮВАЛЬНИХ СИСТЕМ З ВИКОРИСТАННЯМ МОДУЛЯРНОГО КОДУВАННЯ

Фрейман В. І. – д-р техн. наук, професор кафедри «Автоматика та телемеханіка», Пермський національний дослідницький політехнічний університет, м. Перм, Росія.

АНОТАЦІЯ

Актуальність. Обчислювальні системи запроваджено у багато галузей промисловості та економіки сучасного світу. Від надійності їхньої роботи залежать якісні показники тих систем, у яких вони використовуються. Надійність обчислювальної системи складається з надійності побудови та функціонування її елементів. Не завжди можна забезпечити надійність під час проектування вибором якісної елементної бази, структурною надмірністю або іншими відомими способами. Тому важливі та відповідальні елементи обчислювальних систем захищають схемами вбудованого контролю. Вони дозволяють виявляти помилки, які відбуваються під час виконання основних операцій із даними. Ефективним способом побудови таких схем є застосування дій над залишками від розподілу операндів за вибраним модулем або за декількома модулями (модулярне кодування). Особливо завдання вибору найточнішого та найменш надлишкового засобу контролю актуальне для широкого спектру базисних елементів сучасних обчислювальних систем.

Мета роботи. Метою роботи є проведення досліджень та розробка рекомендацій щодо застосування модулярного кодування для підвищення надійності функціонування елементів сучасних обчислювальних систем у різному апаратурно-програмному базисі.

Методи. Вибрано та проаналізовано спосіб числового контролю правильності виконання основних арифметичних та логічних операцій обчислювальними пристроями. На його основі було побудовано та верифіковано схемотехнічну модель обчислювальної системи в середовищі MatLab Simulink, що використовує модулярне кодування як засіб забезпечення надійності функціонування елементів. Проведено аналіз імовірнісних характеристик ухвалення рішення, дано оцінки ймовірності помилкового ухвалення рішення. Створено програмну реалізацію алгоритму імітаційного моделювання у середовищі Visual Basic for Applications, яка дозволила побудувати залежності показників надійності від параметрів кодування.

Результати. Розроблено схемотехнічну модель обчислювальної системи, яка дозволяє досліджувати різні поєднання несправностей функціонування елементів та помилок у виконанні ними операцій. Програмно реалізований алгоритм імітаційного моделювання різноманітних несправностей та помилок функціонування елементів обчислювальних систем під час виконання ними базисних операцій. Визначено якісні залежності ймовірнісних характеристик надійності від параметрів кодування. На підставі аналізу отриманих характеристик зроблено висновки та надано практичні рекомендації щодо застосування модулярного кодування в елементах обчислювальних систем з метою досягнення ними заданих показників надійності.

Висновки. Для підвищення надійності функціонування елементів обчислювальних систем є ефективним застосування схем вбудованого контролю з використанням модулярного кодування. Облік рекомендацій щодо вибору параметрів кодів дозволить забезпечити необхідну надійність за мінімальною надмірністю схем та обчислювальної складності алгоритмів розрахунку.

КЛЮЧОВІ СЛОВА: обчислювальна система, елемент, надійність, діагностика, модулярне кодування, модуль, відрахування, показники надійності, ймовірність виявлення помилки, моделювання, схема, вбудований контроль.

УДК 621.391:004.052

ПОВЫШЕНИЕ НАДЕЖНОСТИ ЭЛЕМЕНТОВ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ С ИСПОЛЬЗОВАНИЕМ МОДУЛЯРНОГО КОДИРОВАНИЯ

Фрейман В. И. – д-р техн. наук, профессор кафедры «Автоматика и телемеханика», Пермский национальный исследовательский политехнический университет, г. Пермь, Россия.

АННОТАЦИЯ

Актуальность. Вычислительные системы внедрены во многие отрасли промышленности и экономики современного мира. От надежности их работы зависят качественные показатели тех систем, в которых они используются. Надежность вычислительной системы складывается из надежности построения и функционирования ее элементов. Не всегда можно обеспечить надежность при проектировании выбором качественной элементной базы, структурной избыточностью или другими известными способами. Поэтому важные и ответственные элементы вычислительных систем защищают схемами встроенного контроля. Они позволяют обнаруживать ошибки, которые происходят при выполнении основных операций с данными. Эффективным способом построения таких схем является применение действий над остатками от деления операндов по выбранному модулю или по нескольким модулям (модулярное кодирование). Особенно задача выбора наиболее точного и наименее избыточного средства контроля актуальна для широкого спектра базисных элементов современных вычислительных систем.

Цель работы. Целью работы является проведение исследований и разработка рекомендаций по применению модулярного кодирования для повышения надежности функционирования элементов современных вычислительных систем в различном аппаратурно-программном базисе.

Методы. Выбран и проанализирован способ числового контроля правильности выполнения основных арифметических и логических операций вычислительными устройствами. На его основе была построена и верифицирована схемотехническая модель вычислительной системы в среде MatLab Simulink, использующей модулярное кодирование как средство обеспечения надежности функционирования элементов. Проведен анализ вероятностных характеристик принятия решения, даны оценки вероятности ошибочного принятия решения. Создана программная реализация алгоритма имитационного моделирования в среде Visual Basic for Applications, которая позволила построить зависимости показателей надежности от параметров кодирования.

Результаты. Разработана схемотехническая модель вычислительной системы, которая позволяет исследовать различные сочетания неисправностей функционирования элементов и ошибок в выполнении ими операций. Программно реализован алгоритм имитационного моделирования всевозможных неисправностей и ошибок функционирования элементов вычислительных систем при выполнении ими базисных операций. Определены качественные зависимости вероятностных характеристик надежности от параметров кодирования. На основании анализа полученных характеристик сделаны выводы и даны практические рекомендации по применению модулярного кодирования в элементах вычислительных систем с целью достижения ими заданных показателей надежности.

Выводы. Для повышения надежности функционирования элементов вычислительных систем эффективно применение схем встроенного контроля с использованием модулярного кодирования. Учет рекомендаций по выбору параметров кодов позволит обеспечить требуемую надежность при минимальной избыточности схем и вычислительной сложности алгоритмов расчета.

КЛЮЧЕВЫЕ СЛОВА: вычислительная система, элемент, надежность, диагностика, модулярное кодирование, модуль, вычет, показатели надежности, вероятность обнаружения ошибки, моделирование, схема, встроенный контроль.

ЛИТЕРАТУРА / LITERATURA

1. State Standard 27.002–1989. Industrial product Dependability. General concepts. Terms and definitions. – Moscow, Standartinform Publ., 1990. 24 p. (In Russian).
2. Jha N. K., Design and synthesis of self-checking VLSI circuits / N. K. Jha, S. J. Wang // *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*. – 2013. – Vol. 12 (6). – P. 878–887. DOI: 10.1109/43.229762.
3. Kon E. L. The theory of telecommunications. The noise stability data transmission within information and control and telecommunication systems: models, algorithms, structures / E. L. Kon, V. I. Freyman. – Perm : Perm State Technical University, 2007. – 317 p.
4. Fink L. M. Signals, noise, errors / L. M. Fink. – Radio and communication, 1984. – 256 p.
5. Freyman V. Research and application of noise stability providing methods at information and control systems / V. Freyman, I. Bezukladnikov // 2017 IEEE Conference of Russian young researchers in electrical and electronic engineering : 1–3 February 2017 : proceedings. – Saint-Petersburg : Saint-Petersburg Electrotechnical University «LETI», 2017. – P. 831–837.
6. Kubalik P. Parity Codes Used for On-Line Testing in FPGA / P. Kubalik, H. Kubatova // *Acta Polytechnica*. – 2005. – Vol. 45, No. 6. – P. 53–59.
7. Sklar B. Digital communications. Fundamentals and applications : second edition / B. Sklar. – New Jersey : Prentice Hall, 2001. – 1079 p.
8. Hazewinkel M. Encyclopedia of Mathematics / M. Hazewinkel. – Springer, 2001. – 488 p.
9. Chiranjeevi G. N. Pipeline Architecture for $N=K*2L$ Bit Modular ALU: Case Study between Current Generation Computing and Vedic Computing / G. N. Chiranjeevi, S. Kulkarni // *Convergence in Technology (I2CT) : IEEE 6th International Conference, Mumbai, India, 02–04 April 2021*. – Mumbai : IEEE, 2021. – P. 1–4. DOI: 10.33130/AJCT.2021v07i01.007.
10. Kumar A. A. Improved coding-theoretic and subspace-based decoding algorithms for a wider class of DCT and DST codes / A. A. Kumar, A. Makur // *IEEE Transactions on Signal Processing*. – 2010. – Vol. 58, Issue 2. – P. 695–708.
11. An Efficient Module Arithmetic Logic Unit in Dual Field for Internet of Things Applications / H. Zeng, W. Li, T. Chen, L. Nan // *Application Specific Integrated Circuits (ASIC) : IEEE 14th International Conference, Kunming, China, 26–29 October 2021 : proceedings*. – Kunming : IEEE, 2021. – P. 1–4. DOI: 10.1109/ASICON52560.2021.9620531.
12. Plantard T. Efficient Word Size Modular Arithmetic / T. Plantard // *Computer Arithmetic (ARITH) : IEEE 28th Symposium, Lyngby, Denmark, 14–16 June 2021 : proceedings*. – Lyngby : IEEE, 2021. – P. 139–139. DOI: 10.1109/ARITH51176.2021.00036.
13. Ahmadifar H., Jaberipur G. Improved modulo- $(2n \pm 3)$ multipliers / H. Ahmadifar, G. Jaberipur // *Computer Architecture and Digital Systems (CADS) : 17th CSI International Symposium, Tehran, 30–31 October 2013*. – Tehran : IEEE, 2013. – P. 31–35. DOI: 10.1109/CADS.2013.6714234.
14. Blahut R. E. Theory and practice of error control codes / R. E. Blahut. – Massachusetts : Addison-Wesley Publishing Company Incorporated, 1986. – 576 p.
15. Freyman V. Research of the Reed-Solomon codes characteristic for realization within control systems devices / V. Freyman // *Radio Electronics, Computer Science, Control*. – 2019. – Vol. 3 (50). – P. 143–151. DOI: 10.15588/1607-3274-2019-3-16.
16. Viterbi A. J. Principles of digital communication and coding / A. J. Viterbi, J. K. Omura. – New York : McGraw-Hill, 2009. – 584 p.
17. Freyman V. I. Methods and algorithms of soft decoding for signals within information transmission channels between control systems elements / V. I. Freyman // *Radio Electronics, Computer Science, Control*. – 2018. – Vol. 4. – P. 226–235. DOI: 10.15588/1607-3274-2018-4-22.
18. Freyman V. Analysis of the transmission reliability between control systems devices when errors are packaged / V. Freyman // *Radio Electronics, Computer Science, Control*. – 2019. – Vol. 4 (51). – P. 223–231. DOI: 10.15588/1607-3274-2019-4-21.
19. Morelos-Zaragoza R. The art of error correcting / R. Morelos-Zaragoza. – Malden : Wiley, 2006. – 269 p.
20. Bhargava K. Efficient implementation of error correction coding in a communication system by using VHDL / K. Bhargava // *VSRD International Journal of Electrical, Electronics and Communication Engineering*. – 2012. – Vol. 2 (6). – P. 359–365.
21. Efanov D. Two-Modulus Codes with Summation of On-Data Bits for Technical Diagnostics of Discrete Systems / D. Efanov, V. Sapozhnikov, V. Sapozhnikov // *Automatic Control and Computer Sciences*. – 2018. – Vol. 52. – P. 1–12. DOI: 10.3103/S0146411618010029
22. MATLAB Documentation [Electronic resource]. – Access mode: <http://www.mathworks.com/help/matlab/>.

УПРАВЛІННЯ У ТЕХНІЧНИХ СИСТЕМАХ

CONTROL IN TECHNICAL SYSTEMS

УПРАВЛЕНИЕ В ТЕХНИЧЕСКИХ СИСТЕМАХ

UDC 004.986:681.5

DOMAIN ONTOLOGY DEVELOPMENT FOR CONDITION MONITORING SYSTEM OF INDUSTRIAL CONTROL EQUIPMENT AND DEVICES

Vlasenko L. O. – PhD, Associate Professor of the Department of Software Engineering and Cyber Security, Kyiv National University of Trade and Economics, Kyiv, Ukraine.

Lutska N. M. – PhD, Associate Professor of the Department of Automation and Computer Technologies of Control Systems, National University of Food Technologies, Kyiv, Ukraine.

Zaiets N. A. – Dr. Sc., Professor of the Department of Department of Automation and Robotic Systems, National University of Life and Environmental Sciences of Ukraine, Kyiv, Ukraine.

Shyshak A. V. – Postgraduate student of the Department of Automation and Computer Technologies of Control Systems, National University of Food Technologies, Kyiv, Ukraine.

Savchuk O. V. – PhD, Senior lecturer of the Department of Power Supply named after Sinkov V., National University of Life and Environmental Sciences of Ukraine, Kyiv, Ukraine.

ABSTRACT

Context. Modern intelligent systems of failure identification of control equipment and devices in food industry are based on a complexation of approaches implemented on various methods and algorithms. The feature of such systems is that within them operates a large amount of heterogeneous data and knowledge that are difficult to combine. The use of ontologies of different levels in the system development process solves this problem.

Objective. Domain ontology development for equipment condition monitoring system as a basis for designing intelligent decision support system with ontology knowledge base.

Methods. There are different ontology development approaches. They may differ in the quantity of levels and types of ontologies or be a combination of subject and problem domains ontologies depending on the complexity of the problem and the chosen ontology development method. This paper represents two levels of the three-level ontology being developed for intelligent condition monitoring system of control equipment and devices. The upper level is represented by top-level ontology Basic Formal Ontology (BFO) which provides systematization of the meta-level, including temporal part. International standards and technical reports such as IEC 62890, ISO 55000, ISA 95, ISA 106, IEC 62264, ISO 10303-242: 2020 are considered in the development process of the second ontology level – Domain ontology.

Results. The article provides Domain ontology for equipment condition monitoring system in food industry. The developed Domain ontology systematizes, structures engineering knowledge and uses BFO which provides a set of basic elements at the meta-level. They set the values of the following entities: type of production, methods of failure identification, causes, failures, events, equipment, etc. The developed Domain ontology has semantic cross-links. A fragment of the Domain ontology relationships for the “Control equipment” subclass of “Equipment” class is also presented in the paper.

Conclusions. The developed ontology can be used to analyze the knowledge base on the causes, locations and types of failures and their identification methods. The developed ontology is a basis for application ontology development.

KEYWORDS: top-level ontology, BFO, domain ontology, failure, control equipment and devices.

ABBREVIATIONS

BFO is a Basic Formal Ontology;
PC is a personal computer;
IDSS is an intelligent decision support system;
SPAN is an ontology for occurrents;
SNAP is an ontology for continuants.

NOMENCLATURE

O_n is an extended domain ontology;
 R is a set of relations specified for classes;
 O^{SP} is a SPAN ontology;
 O^{SN} is a SNAP ontology;
 O^P is a Process ontology;
 O^{PB} is a Process Boundary ontology;
 O^{TR} is a Temporal Region ontology;

O^{SR} is a Spatial Region ontology;
 O^{St} is a State ontology;
 O^{BD} is a Breakdown moment ontology;
 O^{Tm} is a Time ontology;
 O^{SR} is a Spatiotemporal Region ontology;
 O^{Pl} is an ontology of Technological process life cycle;
 O^{El} is an ontology of Equipment life cycle stages;
 O^{IndC} is an Independent continuant ontology;
 O^{DC} is a Dependent continuant ontology;
 O^{GDC} is a Generally dependent continuant;
 O^{SDC} is a Specifically dependent continuant ontology;
 O^{ME} is a Material Entity ontology;
 O^{IME} is an Immaterial Entity ontology;
 O^v is a Variables ontology;
 O^q is a Quality ontology;
 O^{Er} is an Ensuring reliability ontology;
 O^d is a Disposition ontology;
 O^{Sp} is a Spot ontology;
 O^r is a Role ontology;
 O^f is a Function ontology;
 O^C is a Cause ontology;
 O^{Ev} is an Event ontology;
 O^M is a Methods ontology;
 O^{TF} is a Fault type ontology;
 J is a set of criteria for assessing ontology quality;
 Vr is a verification assessment of Domain ontology;
 Vl is a validation assessment of Domain ontology;
 W is a concordance coefficient.

INTRODUCTION

Modern effective IDSS is a set of interconnected subsystems implemented on different methods and algorithms, depending on the purpose of the system. Usually, the basis of IDSS is the production monitoring subsystem. The production monitoring subsystem can have different purposes: monitoring of process stability, equipment condition monitoring, monitoring of other production or economic indicators and their combinations. The quality of production monitoring subsystem operation directly affects the accuracy, correctness, timeliness and reliability of the recommendations that IDSS being developed provides.

Combination of functional and structural parts of the system is a serious problem of the implementation of the intelligent decision support systems. The authors do not know international standards that describe the algorithms and procedures for integrating parts of IDSS for different functional purposes as well as general unified models and tools for developing intelligent systems and standardized coordination mechanisms. This leads to overlapping and duplication of some solutions within the entire manufacturing enterprise or its parts during realization of comprehensive automation of the manufacturing enterprise at different levels. Overlapping and duplication lead to additional financial costs for implementing different solutions of the same problem. In addition, it can lead to collisions, additional time and resource losses and quality deterioration of the manufacturing operation.

In addition, the main problems of intelligent system implementation include the need to convert data formats to provide the interaction between IDSS parts. The developers of complex intelligent systems emphasize the need to create a digital twin of the enterprise or its parts to effectively solve these problems. The digital twin will solve the problem regarding overlapping of system solutions, identification of weak points, modernization, improving and expanding the system.

The problem of heterogeneity and diversity of data within IDSS, their complexity, inconsistency and absence of pattern does not exist for ontologies because they combine any data and knowledge. Therefore, it is advisable to use ontologies of different levels during developing the system.

The object of study is a process of combining the knowledge and data of manufacturing enterprise to provide condition monitoring of control equipment and devices in food industry.

The subject of study is Domain ontology for condition monitoring system of industrial control equipment and devices.

The purpose of the work is Domain ontology development for condition monitoring system of control equipment and devices of industrial enterprise, in particular, in the food industry, which systematizes and conceptualizes knowledge, objects and processes for many related tasks and is based on BFO.

1 PROBLEM STATEMENT

Formally, the BFO model is described by a tuple (1):

$$On = \langle O^{SP}, O^{SN}, R \rangle. \quad (1)$$

Model (1) should include only those concepts used for the developed Domain ontology. For the BFO model, the set of relations R is limited to "is_a".

SPAN ontology model O^{SP} is described by a tuple (2):

$$O^{SP} = \langle O^P, O^{PB}, O^{TR}, O^{SR} \rangle. \quad (2)$$

SNAP ontology model O^{SN} is described by a tuple (3):

$$O^{SN} = \langle O^{IndC}, O^{DC} (O^{GDC}, O^{SDC}) \rangle. \quad (3)$$

Tuples (2) and (3) at the level of Domain ontology should describe the concepts and their relationships that provide monitoring of industrial control equipment and devices in accordance with existing international standards in the industrial automation domain.

Domain ontology should be focused on the following tasks: qualitative representation of domain knowledge; systematization and structuring of information; formalization of engineering knowledge and management of effective research of domain knowledge; application in the solution development of individual problems in IDSS.

2 REVIEW OF THE LITERATURE

Ontology has been developed to represent and describe domain knowledge understandable to machines

(PCs) and humans. Ontologies are used in many fields of computer science such as software development [1], knowledge engineering [2], semantic networking, information retrieval [3], etc.

Metadata such as domain description, purpose of ontologies, are used to describe the context of ontologies and to integrate them. However, these metadata may not always be available in ontologies, and it is a significant problem. There are different ways to integrate ontologies. One option is based on automatic domain identification and presented in [4], the other option is based on OWL – in [5, 6]. Integration of top-level, domain and application ontologies is not considered in the article.

There are many different approaches to increase production efficiency. Most of them improve the technological process. However, there is an objective need to use equipment diagnostics subsystems. Because the repair time and equipment downtime directly depend on the complexity of the failure. Downtime reduction leads to increased profits and reduced losses.

Classification of failures and reasons of their occurrence [7], accurate selection of methods for diagnosing equipment conditions significantly reduce the time to find the failure and the place of its occurrence, and, consequently, increase the efficiency of response to the event. In addition, an effective diagnostic system provides timely preventive maintenance that also increase productivity. Today, there are alternative solutions regarding the use of ontologies in the development of failures searching systems in various fields: software development, construction, automotive industry, services [8, 9], etc. For food industry it is necessary to consider its specifics as well as existing international industry standards.

To develop IDSS with ontological database and equipment condition monitoring subsystem it is necessary to develop top-level ontology and domain ontology at the first stage. The next step is developing of application ontology.

The decision support system for equipment condition monitoring includes different quantity of heterogeneous information. It comes from technological process, automation system, equipment condition monitoring subsystem and other subsystems. For convenience, the production monitoring system was divided into three parts: technological equipment condition monitoring, control equipment and devices condition monitoring, electrical equipment condition monitoring.

3 MATERIALS AND METHODS

Any ontology are terms and its meanings used in specific domain, relationships between terms in hierarchical structure and their definitions. Therefore, Domain ontology should be developed correctly to implement application ontology. To build Domain ontology, it is necessary to structure the system of concepts at the semantic level by providing the basic concepts and properties of automated control systems, intelligent systems, technological objects of food industry and hardware. This is a basis for integration of information and data coming from different

decentralized sources throughout the life cycle of intelligent equipment condition monitoring system. Large volumes of heterogeneous data are at all stages of equipment and system life cycle.

The authors have experience in building ontologies for technical applications. The results are presented in [10]. The feature of the ontologies given in [10] is the approach of Ukrainian academician A. V. Palagin [11] taken as a basis. This approach has undeniable advantages such as task and method orientation, but significant disadvantage is the lack of standardization. Therefore, Basic Formal Ontology is chosen among existing top-level ontologies to solve defined tasks [12].

The advantages of BFO are current development of the standard [13] and version BFO 2.2 which is OWL-compliant. BFO is based on the monohierarchy principle. According to monohierarchy principle, types and subtypes taxonomies of specific classes in compatible ontologies have hierarchical structure. The simplicity of the BFO structure implies the relative simplicity of the formulations, which leads to the fact that the node of the universal graph has only one parent node with the relation “is_a”. This principle avoids errors in the development of top-level ontology. It should be noted that all of the mentioned above provides significant technical advantages in further applied implementation.

At the upper level, BFO is divided into two ontologies SNAP and SPAN [14], which correspond to two categories of individuals. They do not overlap and correspond to space and time: continuant and occurrent. Continuant are continuous entities that include objects, attributes, and locations. Occurrent includes entities that contain processes and time domains. They occur, deploy, and evolve over time. The feature of continuant is integral existence at any time unlike occurrent consists of parts that change over time, they have a beginning, middle and end.

Domain ontology is developed to build application ontology that will search for type of failure, its location and causes. Domain ontology is usually needed to build a knowledge model that describes multi-party project development process.

The developed BFO-based Domain ontology for food industry productions, which contains concepts in structured, systematic and flexible format, is presented below. The selected concepts of the domain correspond to the respective concepts in ISA 106, ISA 95 (IEC 62264) and other industry standards. In addition, appropriate hierarchies are maintained. This provides an openness of domain ontology and ability to make changes and extensions of it to present specific knowledge of the research domain. Domain ontology for the equipment condition monitoring system is based on knowledge and coordination between different industries, disciplines and subsystems.

The following steps were performed during Domain ontology development for equipment condition monitoring system to structure data and knowledge and select the appropriate classes:

1. A three-level ontology consisting of top-level ontology, domain ontology and application ontology is chosen to implement the tasks.

2. BFO is chosen as a top-level ontology. Temporal part allows to take into account the dynamics of ontologies, their classes and entities. It is possible to use version BFO 2.0 on OWL. BFO-individuals allow to clearly structure and specify domain knowledge at the level of domain ontology.

3. Domain ontology is divided into two ontologies according to BFO: SNAP and SPAN. The advantage of this approach is a separate selection of invariant entities-objects and entities-processes with the corresponding time domains and labels for domain.

4. Physical model described in ISA 106, ISA 88 and role-based equipment hierarchy model described in ISA 95 and features of equipment in food industry are considered to properly select appropriate classes and establish correct semantic relationships between objects and designing processes.

5. The quality control of the developed ontology is obligatory stage. Today, there is no standard for this procedure, but it is possible to use several methods [15].

BFO entities are specified by domain ontology subclasses to describe and define the basic domain concepts of equipment condition monitoring.

SPAN ontology is described by tuple (4):

$$O^{SP} = \langle O^{St}, O^{BD}, O^{Tm}, O^{SR}(O^{Pl}, O^{El}) \rangle. \quad (4)$$

The ontology of occurrent domain consists of four entities: State O^{St} corresponds to equipment condition (operative condition, fault state, after-fault state); Breakdown moment O^{BD} stores a set of timestamps that correspond to the moment of equipment failure; Temporal Region at Domain ontology level corresponds to time part Time O^{Tm} that describes the period of time during which the event occurs; Spatiotemporal Region O^{SR} divided into two parts – O^{Pl} corresponds to stages of technological process life cycle and O^{El} corresponds to stages of equipment life cycle.

Combination of SPAN ontology and Domain ontology for entities that evolve over time is shown in Fig. 1. BFO entities are above the line and Domain ontology entities are below the line. The subclasses of which the corresponding class consists are given in the form of a list.

Continuant consists of Independent continuant and Dependent continuant according to the documentation. Independent continuant is an entity that can exist independently or be a part of another entity. Dependent continuant exists due to another object or as a part of another object. Independent continuant is divided into two entities: Material Entity and abstract Immaterial Entity.

Domain ontology for SNAP consists of two parts, which are described by the following tuples (5) and (6). For Independent continuant (Fig. 2):

$$O^{IndC} = \langle O^{ME}, O^{IME} \rangle, \quad (5)$$

O^{ME} – Material Entity ontology at Domain ontology level includes data about the structure of manufacturing enterprise, technological objects, material batch, production equipment.

Immaterial Entity O^{IME} is a class of location of technological equipment – production units and lines. Spatial Region provides management of 2D and 3D drawings and models. It gives an opportunity to find a place of failure and device which failed with the indication of its location on the corresponding drawing during documentation development using specific software. It will greatly facilitate staff work and reduce the time for communication between services of the enterprise.

Dependent continuant (Fig. 3) is described by tuple (6):

$$O^{DC} = \langle O^{GDC}(O^v), O^{SDC}(O^q(O^{Er}), O^d(O^{Sp}, O^l), O^r(O^C, O^{Ev}, O^M, O^{TF})) \rangle, \quad (6)$$

O^v – variables ontology at Domain ontology level which includes subclasses of technological, information vari-

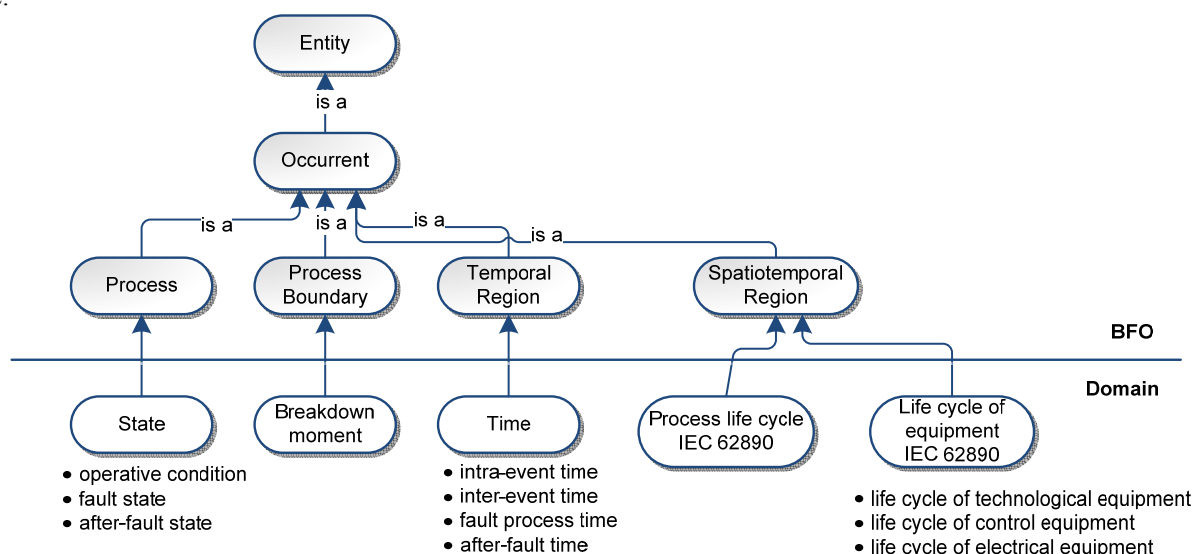


Figure 1 – SPAN and Domain ontology

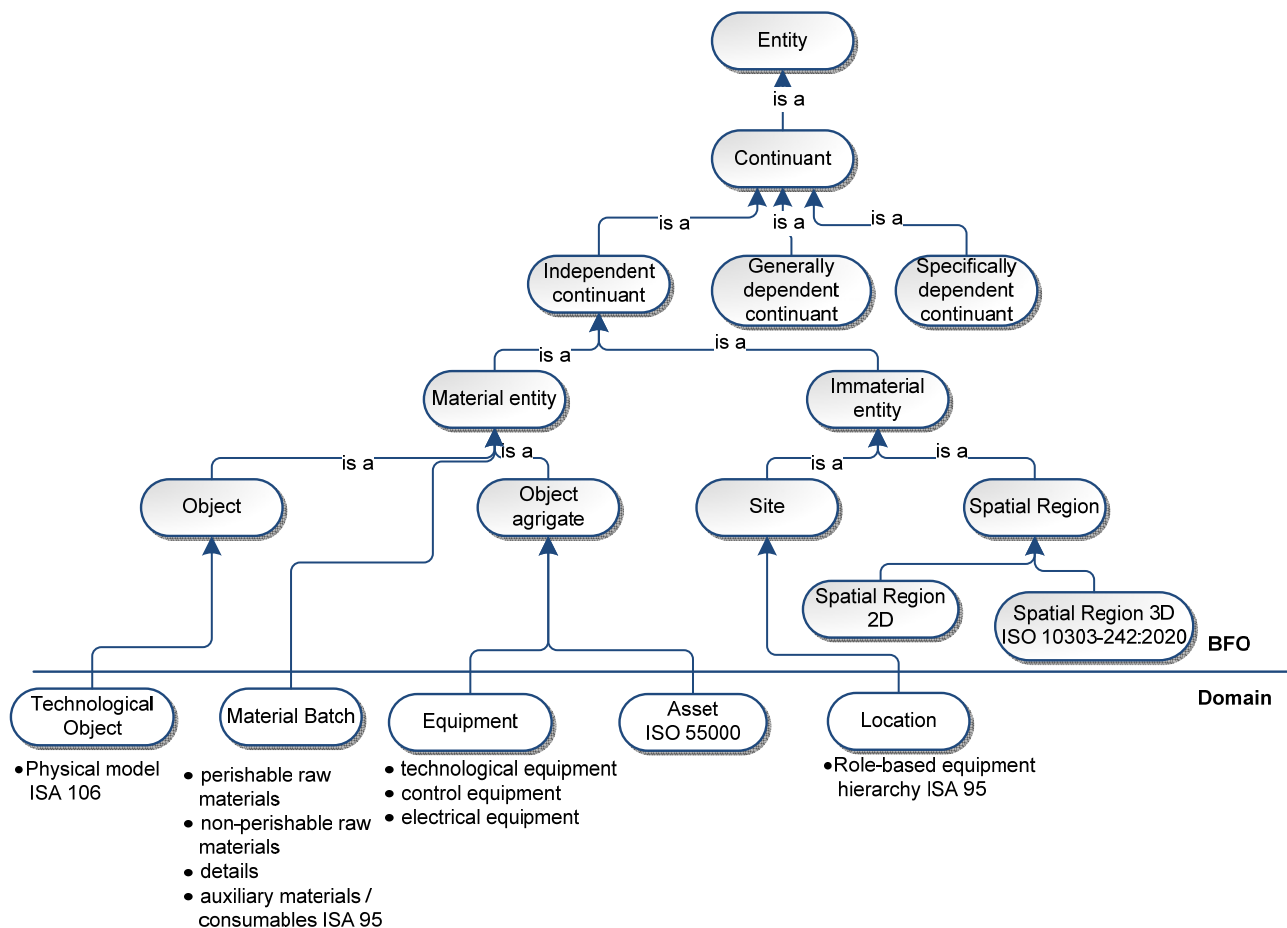


Figure 2 – SNAP and Domain ontology (Independent Continuant)

ables and quality indicators of equipment condition; O^q – an entity of Ensuring reliability which describes the indicators of equipment reliability; O^d – Disposition ontology which consists of equipment locations and failures locations; O^r – Role ontology classifies information about failures causes (Cause), events of two types (Event): Repair and Maintenance of equipment, subjective and objective Methods for equipment condition diagnostics and types of failures; O^f – Function ontology corresponds to functions performed within the developed system.

4 EXPERIMENTS

Experimental research was performed to verify the effectiveness of the developed Domain Ontology for condition monitoring system of control equipment and devices for data and knowledge structuring on the example of general types of equipment failures belonging to the Control equipment class at the sugar plant for two seasons of its operation.

The ontology quality is assessed comprehensively by developers and future users. The assessment is based on

the ontology verification and validation results because the authors do not know united approach that describe a procedure to make a technical conclusion regarding the correspondence of the ontology content to the system at each stage of its life cycle.

All requirements are met. The assessment is performed according to the set of criteria, which are described by tuple (7):

$$J = \langle Vr, Vl \rangle. \quad (7)$$

Validity Vl actually indicates the correctness of the syntactic coding of the ontology specification, which is checked using OWL validator URL: mowlpower.cs.man.ac.uk:8080/validator/. The absence of logical contradictions in the ontology ensures its compatibility. If there are contradictions, the ontology is incompatible and as a result, any formal model cannot satisfy ontology axioms.

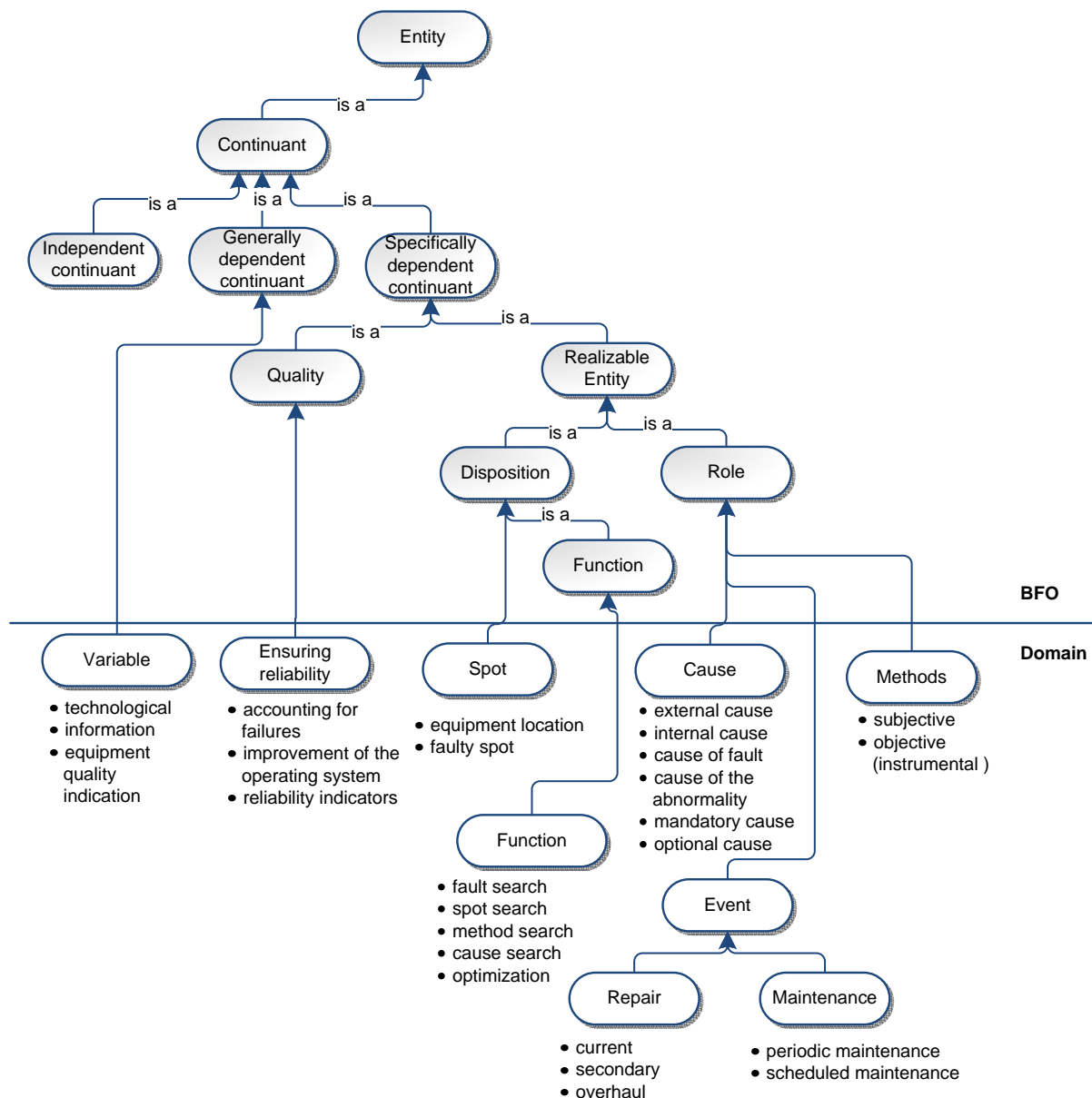


Figure 3 – SNAP and Domain ontology (Dependent Continuant)

Verification assessment V_r indicates the formal correctness of the developed Domain ontology according to syntactic validity and the absence of logical contradictions. Evaluation of Domain Ontology semantic correctness is usually based on “gold standard”. “Gold standard” is an evaluation method based on a specific task, specific sources of knowledge or expert assessments. Whereas the authors did not find a suitable basic ontology or mathematical model for “gold standard” evaluation for the studied domain, assessments based on specific tasks and sources of knowledge are quite subjective and may not reflect the real effect. The set of assessments V_r for evaluation of the effectiveness of the developed solutions to provide experts was chosen. These include the following criteria:

Accessibility (V_{r1}). This criterion characterizes the ability to access the ontology: hosting in open repositories,

payment conditions, support of global search, general ontology support.

Completeness and adaptability (V_{r2}). This criterion measures diversity of the domain ontological base and its completeness and possibility to adapt it to the specific problem.

Openness (V_{r3}). This criterion characterizes the possibility of refinement, addition of the necessary terms, relations, axioms, rules, etc.

Quality (V_{r4}). This criterion shows whether the developing ontology remains consistent, correct, whether there is a feedback mechanism to track problems.

Standardization (V_{r5}). This criterion characterizes the compliance of the ontology with the existing international standard, or affiliation to the standard.

5 RESULTS

The developed Domain ontology has semantic cross-links. A fragment of the Domain ontology relationships for the “Control equipment” subclass of “Equipment” class is presented in Fig. 4.

One of the ontology queries is “What are the typical failures specific to a particular class of equipment (such as motors)?”. The answer is a subset of instances of the class Fault type (Fig. 4). Or on the query: “What are the typical causes of failure for the appropriate location and type of equipment?”, a user gets a subset of instances of the Cause class. Because Control equipment class connects to Equipment location class which determines the location of technical devices through “hasLocation” semantic property. Similarly, the relations “hasState”, “ServiceOf” are used to describe the state and metrological service of equipment. “hasFault” and “CharacteristicOf” relationships associate equipment with the corresponding types and causes of failures. Types and causes of failures are also related by their relationship CallOf and hasCause. Equipment states are included in the

equipment life cycle stage provided by the appropriate “PartOf” relationship.

Fig. 4 does not completely reflect all the relationships that exist in the domain, in particular, the axioms of classes and instances.

Ontology metrics for BFO and Domain ontology levels are shown in the table 1.

Table 1 – Ontology metrics

Ontology metric	Quantity
Class count	42
SubClass	45
Relation	84

The quantity of used Relations is shown in the table 2. From the given tables it is possible to draw a conclusion that implementation of the developed ontologies significantly saves time for development and description of separate ontologies and provides compactness of ontology and variability of decisions.

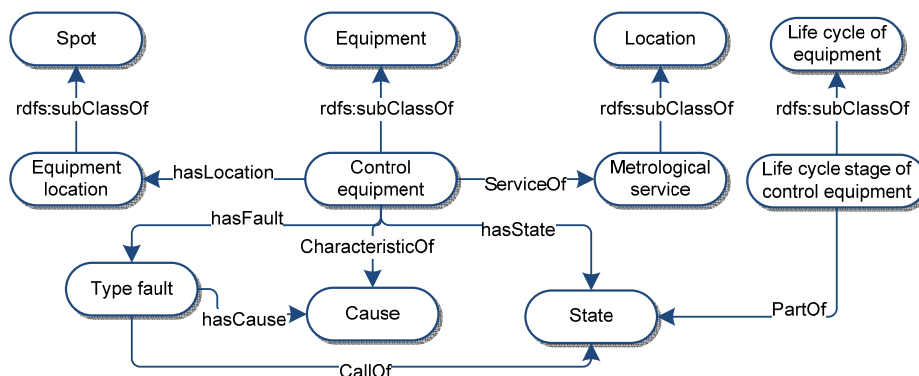


Figure 4 – A fragment of the relationships between Domain ontology classes

Table 2 – The quantity of Relations by type

Relation	Quantity
is a	20
has part	17
has function	7
has property	21
has standard	11
has state	3
has location	5
other relation	18

6 DISCUSSION

Experts conducted the testing based on inquiries about the types, types of failures, their location, etc. A certain list of criteria was evaluated according to the results of the testing. The expert assessment was conducted by a group of 7 experts, selected according to the general requirements for their qualification among specialists and future users. Expertise was conducted in the form of a group assessment with simultaneous analysis of the competence of experts.

The results of expert assessments are shown in the diagram in Fig. 5. The average number of queries of each expert is 48, and the uniqueness of queries is 21%.

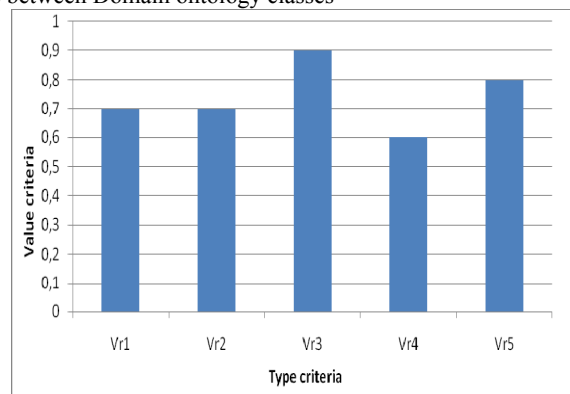


Figure 5 – The results of expert assessments

The concordance coefficient in the group $W = 0.927$, which indicates a high consistency of experts’ opinions. Therefore, the developed Domain ontology is semantically successful, although it requires a slight increase in the assessment of the quality criterion (Vr_4).

It should be highlighted that the adaptability of the developed ontology is limited due to the use of not only general and standard ontologies, but also self-developed ontologies and its specific domain orientation limited by failures in food industry.

The result of consistency assessment is positive. Domain ontology does not contain critical problems: cyclic hierarchies of classes, redundant axioms, logically incompatible definitions of concepts and relations. The ontology is also successfully tested for the accuracy and completeness of the domain for the developed IDSS with the equipment condition monitoring subsystem by stakeholders. The adequacy of compliance with the basic conceptual model of the developed system, the correctness of definitions, concepts and relationships are confirmed.

CONCLUSIONS

As a result of this work, Domain ontology for condition monitoring system of control equipment and devices is developed based on BFO. Domain ontology systematizes and formalizes domain knowledge and data and is the basis for defining individual enterprise tasks implemented by IDSS based on application ontology.

The scientific novelty. For the first time Domain ontology was developed for condition monitoring system of control equipment and devices, based on modern industrial standards, taking into account the features of failures of control devices, which allowed to formulate and present domain knowledge and data and systematize the domain for further development of an intelligent decision support system for condition monitoring.

The practical significance. The developed ontology identifies the objects and processes of equipment condition monitoring domain such as states and stages of the equipment life cycle, causes and locations of failures, etc. which provide a basis for collecting the necessary information in one place and a quick search for data about failures and their causes. Because of semantic relations between objects and processes as well as considerations about physical model described in ISA 106, ISA 88 and role-based equipment hierarchy model described in ISA 95 and features of equipment in food industry, developed Domain ontology, which consist of 171 entities, provides necessary logical chains for timely failure identification, prevention of serious failures and downtime reduction. All of the above will define the new management of the equipment condition monitoring, which will lead to an economic effect.

Prospects for further research. The developed Domain ontology is the basis for the development of an applied ontology of actual production tasks which are related to the construction of industrial intelligent knowledge bases and ontological repositories.

REFERENCES

1. Alrumaih H., Mirza A., Alsalamah H., Domain Ontology for Requirements Classification in Requirements Engineering Context, *IEEE Access*, 2020, Vol. 8, pp. 89899–89908. DOI: 10.1109/ACCESS.2020.2993838
2. Alterovitz G., Xiang M., Hill DP., Lomax J., Liu J., Cherkassky M., Dreyfuss J., Mungall C. et al. Ontology engi-

- neering, *Nature Biotechnology*, 2010, Vol. 28(2), pp. 128–130. DOI: 10.1038/nbt0210-128
3. Sharma S., Kumar A., Rana V., Ontology Based Informational Retrieval System on the Semantic Web: Semantic Web Mining, *2017 International Conference on Next Generation Computing and Information Systems (ICNGCIS)*, 2017, pp. 35–37. DOI: 10.1109/ICNGCIS.2017.21
4. Wu D., Hakansson A. A method of identifying ontology domain, *Procedia Computer Science*, 2014, Vol. 35, pp. 504–513. DOI: 10.1016/j.procs.2014.08.131
5. Valra A., Madeddu D., Chiappetti J., Farina D. The BIM Management System: A Common Data Environment Using Linked Data to Support the Efficient Renovation in Buildings, *Proceedings*, 2020, Vol. 65(1):18. DOI: 10.3390/proceedings2020065018
6. Mohd Ali M., Rai R., Otte J. N., Smith B. A Product Life Cycle Ontology for Additive Manufacturing, *Computers in Industry*, 2019, Vol. 105, pp. 191–203. DOI: 10.1016/j.compind.2018.12.007
7. Kitamura Y., Mizoguchi R. An Ontological Analysis of Fault Process and Category of Faults. *Proc. 10th International Workshop on Principles of Diagnosis (DX-99)*, 1999, pp. 118–128.
8. Voisin A., Medina-Oliva G., Monnin M., Leger J., Jung B. Fault Diagnosis System Based on Ontology for Fleet Case Reused. *Ebrahimipour V., Yacout S. (eds) Ontology Modeling in Physical Asset Integrity Management*, Springer. Cham, 2015, pp. 133–169. DOI: https://doi.org/10.1007/978-3-319-15326-1_5
9. Kitamura Y., Sano T., Namba K., Mizoguchi R. A functional concept ontology and its application to automatic identification of functional structures, *Advanced Engineering Informatics*, 2002, Vol. 16(2), pp. 145–163. DOI: 10.1016/S1474-0346(02)00006-X
10. Lutska N., Vlasenko L., Zaiets N., Shtepa V. Vasant P., Zelinka I., Weber GW. (eds) Ontological Aspects of Developing Robust Control Systems for Technological Objects. *Intelligent Computing and Optimization. ICO 2020. Advances in Intelligent Systems and Computing*, 2021, Vol. 1324, pp. 1252–1261. DOI: 10.1007/978-3-030-68154-8_107
11. Palagin A., Petrenko N. System-ontological analysis of the subject area, *Control systems and machines*, 2019, Issue 3 (281), pp. 23–37. DOI: 10.15407/csc.2019.03.023
12. Arp R., Smith B. Function, Role, and Disposition in Basic Formal Ontology, *Nature Precedings*, 2008, Vol. 1941.1, pp. 1–4. DOI: 10.1038/npre.2008.1941.1
13. Smith B. et al. Basic Formal Ontology 2.0. Specification and user's guide, 2015. URL: <https://confluence.ihtsdotools.org>
14. Grenon P., Smith B. SNAP and SPAN: Towards Dynamic Spatial Ontology, *Spatial cognition and computation*, 2004, Vol. 4 (1), pp. 69–104. DOI: 10.1207/s15427633scc0401_5
15. Mazzola L., Kapahnke P., Vujic M., Klusch M. CDM-Core: A Manufacturing Domain Ontology in OWL2 for Production and Maintenance, *Proceedings of the 8th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2016)*, 2016, Vol. 2, pp. 136–143. DOI: 10.5220/0006056301360143

Received 25.09.2021.

Accepted 08.11.2021.

УДК 004.986:681.5

РОЗРОБКА ОНТОЛОГІЇ ПРЕДМЕТНОЇ ОБЛАСТІ ДЛЯ СИСТЕМИ МОНІТОРИНГУ СТАНУ ТЕХНІЧНИХ ЗАСОБІВ АВТОМАТИЗАЦІЇ

Власенко Л. О. – канд. техн. наук, доцент, доцент кафедри інженерії програмного забезпечення та кібербезпеки, Київський національний торговельно-економічний університет, Київ, Україна.

Луцька Н. М. – канд. техн. наук, доцент, доцент кафедри автоматизації та комп'ютерних технологій систем управління, Національний університет харчових технологій, Київ, Україна.

Засць Н. А. – д-р техн. наук, професор, професор кафедри автоматики та робототехнічних систем ім. академіка Мартиненка І. І., Національний університет біоресурсів і природокористування України, Київ, Україна.

Шишак А. – аспірант, асистент кафедри автоматизації та комп'ютерних технологій систем управління, Національний університет харчових технологій, Київ, Україна.

Савчук О. В. – канд. техн. наук, старший викладач кафедри електропостачання імені Синькова В. М., Національний університет біоресурсів і природокористування України, Київ, Україна

АНОТАЦІЯ

Актуальність. Сучасні інтелектуальні системи ідентифікації поломок технічних засобів автоматизації харчової промисловості будуються на комплексуванні підходів, що реалізовані на різних методах і алгоритмах. Особливістю таких систем є те, що в їх межах функціонує велика кількість різнорідних даних і знань, які важко поєднати між собою. Використання онтологій різних рівнів при розробці системи вирішує дану проблему.

Мета. Побудова онтології предметної області для моніторингу стану обладнання, що є основою для проектування інтелектуальної системи підтримки прийняття рішень з онтологічною базою знань.

Метод. Існують різні підходи до побудови онтологій. В залежності від складності задачі і обраного способу проектування онтологій вони можуть відрізнятися кількістю рівнів та видів онтологій або являти собою поєднання онтологій предметної та проблемної областей. В даній роботі сформована трирівнева онтологія, що проектується для інтелектуальної системи моніторингу стану технічних засобів автоматизації. Верхній рівень представлено онтологією верхнього рівня Basic Formal Ontology, що забезпечує систематизацію метарівня, включаючи темпоральну частину. При проектуванні другого рівня – онтологій предметної області, враховано міжнародні стандарти: IEC 62890, ISO 55000, ISA 106, IEC 62264, ISO 10303-242:2020.

Результати. В статті наведено онтологію предметної області для системи моніторингу стану обладнання харчової галузі, яка систематизує та структурує інженерні знання. Розроблена онтологія предметної області використовує Basic Formal Ontology, яка надає набір базових елементів на метарівні. Вони задають значення виділених сутностей: вид виробництва, методи визначення поломок, причини, поломки, події, обладнання тощо. Онтологія розробленої предметної області має введені семантичні перехресні зв'язки. Також в роботі наведено фрагмент відношень в онтології предметної області системи для підкласу Технічні засоби автоматизації класу Обладнання.

Висновки. Дана онтологія може бути використана для аналізу бази знань за причинами, місцями та видами поломок і методами їх визначення, та є основою для розробки прикладної онтології.

КЛЮЧОВІ СЛОВА: онтологія верхнього рівня, BFO, онтологія предметної області, поломка, технічні засоби автоматизації.

УДК 004.986:681.5

РАЗРАБОТКА ОНТОЛОГИИ ПРЕДМЕТНОЙ ОБЛАСТИ ДЛЯ СИСТЕМЫ МОНИТОРИНГА СОСТОЯНИЯ ТЕХНИЧЕСКИХ СРЕДСТВ АВТОМАТИЗАЦИИ

Власенко Л. А. – канд. техн. наук, доцент, доцент кафедры инженерии программного обеспечения и кибербезопасности, Киевский национальный торгово-экономический университет, Киев, Украина.

Луцкая Н. Н. – канд. техн. наук, доцент, доцент кафедры автоматизации и компьютерных технологий систем управления, Национальный университет пищевых технологий, Киев, Украина.

Заец Н. А. – д-р техн. наук, профессор, профессор кафедры автоматики и робототехнических систем им. академика Мартыненко И. И., Национальный университет биоресурсов и природопользования Украины, Киев, Украина.

Шишак А. В. – аспирант, ассистент кафедры автоматизации и компьютерных технологий систем управления, Национальный университет пищевых технологий, Киев, Украина.

Савчук О. В. – канд. техн. наук, старший преподаватель кафедры электроснабжения имени Синькова В. М., Национальный университет биоресурсов и природопользования Украины, Киев, Украина.

АННОТАЦИЯ

Актуальность. Современные интеллектуальные системы идентификации поломок технических средств автоматизации пищевой промышленности строятся на комплексировании подходов, реализованных на различных методах и алгоритмах. Особенностью таких систем является то, что в их пределах функционирует большое количество разнородных данных и знаний, которые трудно совместить между собой. Использование онтологий различных уровней при разработке системы решает данную проблему.

Цель. Построение онтологии предметной области для мониторинга состояния оборудования, является основой для проектирования интеллектуальной системы поддержки принятия решений с онтологической базой знаний.

Метод. Существуют различные подходы к построению онтологий. В зависимости от сложности задачи и выбранного способа проектирования онтологий они могут отличаться количеством уровней и видов онтологий или представлять собой сочетание онтологий предметной и проблемной областей. В данной работе сформирована трехуровневая проектируемая онтология для интеллектуальной системы мониторинга состояния технических средств автоматизации. Верхний уровень

представлен онтологией верхнего уровня Basic Formal Ontology, что обеспечивает систематизацию метауровне, включая темпоральную часть. При проектировании второго уровня – онтологии предметной области, учтены международные стандарты: IEC 62890, ISO 55000, ISA 106, IEC 62264, ISO 10303-242:2020.

Результаты. В статье показана онтология предметной области для системы мониторинга состояния оборудования пищевой отрасли, которая систематизирует и структурирует инженерные знания. Разработанная онтология предметной области использует Basic Formal Ontology, которая предоставляет набор базовых элементов на метауровне. Они задают значения выделенных сущностей: вид производства, методы определения поломок, причины, поломки, события, оборудования. Онтология разработанной предметной области включает семантические перекрестные связи. В качестве примера, приведен фрагмент отношений в онтологии предметной области системы для подкласса Технические средства автоматизации класса Оборудование.

Выводы. Данная онтология может быть использована для анализа базы знаний по причинам, местам и видам поломок и методам их определения, и является основой для разработки прикладной онтологии.

КЛЮЧЕВЫЕ СЛОВА: онтология верхнего уровня, BFO, онтология предметной области, поломка, технические средства автоматизации.

ЛИТЕРАТУРА / LITERATURA

1. Alrumaih H. Domain Ontology for Requirements Classification in Requirements Engineering Context / H. Alrumaih, A. Mirza, H. Alsalamah // Access. – IEEE. – 2020. – Vol. 8. – P. 89899–89908. DOI: 10.1109/ACCESS.2020.2993838
2. Alterovitz G. Ontology engineering / [M. Xiang, DP. Hil et al.] // Nature Biotechnology. – 2010. – Vol. 28 (2). – P. 128–130. DOI: 10.1038/nbt0210-128
3. Sharma S. Ontology Based Informational Retrieval System on the Semantic Web: Semantic Web Mining / S. Sharma, A. Kumar, V. Rana // 2017 International Conference on Next Generation Computing and Information Systems (ICNGCIS). – 2017. – P. 35–37. DOI: 10.1109/ICNGCIS.2017.21
4. Wu D. A method of identifying ontology domain / D. Wu, A. Hakansson // Procedia Computer Science. – 2014. – Vol. 35. – P. 504–513. DOI: 10.1016/j.procs.2014.08.131
5. Valra A. The BIM Management System: A Common Data Environment Using Linked Data to Support the Efficient Renovation in Buildings / [D. Madeddu, J. Chiappetti, D. Farina] // Proceedings. – 2020. – Vol. 65(1):18. DOI: 10.3390/proceedings2020065018
6. A Product Life Cycle Ontology for Additive Manufacturing / [M. Mohd Ali, R. Rai, J. N. Otte, B. Smith] // Computers in Industry. – 2019. – Vol. 105. – P. 191–203. DOI: 10.1016/j.compind.2018.12.007
7. Kitamura Y. An Ontological Analysis of Fault Process and Category of Faults / Y. Kitamura, R. Mizoguchi // Proc. 10th International Workshop on Principles of Diagnosis (DX-99). – 1999. – P. 118–128.
8. Fault Diagnosis System Based on Ontology for Fleet Case Reused / [G. Medina-Oliva, M. Monnin J. Leger, B. Iung] // Ontology Modeling in Physical Asset Integrity Management. – Springer, Cham, 2015. – P. 133–169. DOI: https://doi.org/10.1007/978-3-319-15326-1_5
9. A functional concept ontology and its application to automatic identification of functional structures / [Y. Kitamura, T. Sano, K. Namba, R. Mizoguchi] // Advanced Engineering Informatics. – Elsevier Ltd, 2002. – Vol. 16(2). – P. 145–163. DOI: 10.1016/S1474-0346(02)00006-X
10. Ontological Aspects of Developing Robust Control Systems for Technological Objects / N. Lutskaya, L. Vlasenko, N. Zaiets, V. Shtepa, Vasant P., Zelinka I., Weber GW. (eds) // Intelligent Computing and Optimization. ICO 2020. Advances in Intelligent Systems and Computing. Springer, Cham. – 2021. – Vol. 1324. – P. 1252–1261. DOI: 10.1007/978-3-030-68154-8_107
11. Palagin A. System-ontological analysis of the subject area / A. Palagin, N. Petrenko // Control systems and machines. – 2009. – Issue 3 (281). – P. 23–37. DOI: 10.15407/csc.2019.03.023
12. Arp R. Function, Role, and Disposition in Basic Formal Ontology / R. Arp, B. Smith // Nature Precedings. – 2008. – Vol. 1941.1. – P. 1–4. DOI: 10.1038/npre.2008.1941.1
13. Smith B. Basic Formal Ontology 2.0. Specification and user's guide / [B. Smith et al.]. – 2015. <https://confluence.ihtsdotools.org>
14. Grenon P. SNAP and SPAN: Towards Dynamic Spatial Ontology / P. Grenon, B. Smith // Spatial cognition and computation. – 2004. – Volume 4 (1). – P. 69–104. DOI: 10.1207/s15427633scc0401_5
15. CDM-Core: A Manufacturing Domain Ontology in OWL2 for Production and Maintenance / [L. Mazzola, P. Kapahnke, M. Vujic, M. Klusch] // Proceedings of the 8th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management. – KEOD, (IC3K 2016). Porto, Portugal. – 2016. – Vol. 2. – P. 136–143. DOI: 10.5220/0006056301360143

DECISION-MAKING AT EVOLUTIONARY SEARCH DURING LIMITED NUMBER OF FUZZY EXPERIMENTS WITH MULTIPLE CRITERIA

Irodov V. F. – Dr. Sc., Professor, Professor of the Department of Information Technology, Dnipro Technological University “Step”, Dnipro, Ukraine.

Shaptala M. V. – PhD, Associate Professor, Rector of University, Dnipro Technological University “Step”, Dnipro, Ukraine.

Dudkin K. V. – PhD, Director of Limited Liability Company “KV-Automation”, Kiev, Ukraine.

Shaptala D. E. – PhD, Associate Professor of the Department of Information Technology, Dnipro Technological University “Step”, Dnipro, Ukraine.

Chirin D. A. – Postgraduate student of the Department of System Analysis and Modeling in Heat and Gas Supply, Pridneprovsk State Academy of Civil Engineering and Architecture, Dnipro, Ukraine.

ABSTRACT

Context. The mechanism of decision-making during limited number of fuzzy experiments with multiple criteria are considered. The investigation object is process decision-making for project or control in complex systems with multiple criteria.

Objective. It is necessary to determine optimal (most preferred) parameters of the systems with multiple criteria. It is no the mathematical model of the system, there is limited number of fuzzy experiments only.

Method. experimental study of a process with several criteria (functions) depending on its parameters; the use of expert fuzzy evaluation to build a matrix of preferences for individual implementations; building a function of choosing preferred solutions based on a preference matrix by constructing a mathematical model of preference recognition, formulation and solving the problem of generalized mathematical programming as the final step in building the selection mechanism. The decision-making mechanism depends on the expert assessment procedure when comparing a limited set of results with each other, as well as on the statement of conditions when solving the problem of generalized mathematical programming. Comparison of a finite number of fuzzy experiments is convenient for expert evaluation. Presentation of the final choice as a result of solving the problem of generalized mathematical programming is convenient for using such a mechanism in automatic control systems already without human intervention. The proposed scheme of decision-making during limited number of fuzzy experiments has been applied to decision-making of project management for pellet burner.

Results. Experimental decision-making fuzzy results are presented in the presence of several criteria for a pellet burner of a tubular heater, which confirm the acceptability of the developed decision-making mechanism. It was proposed the new scheme for constructing a selection mechanism for decision-making in systems with several criteria where there is a sample of fuzzy experimental results.

Conclusions. The scheme of decision-making is includes the solving the generalized mathematical programming as the final step in building the selection mechanism. For the problem of generalized mathematical programming it may be applied the evolutionary search with choice function in form of preference or in form of lock.

KEYWORDS: decision-making, multiple criteria, fuzzy experiments, function of choosing, evolutionary search.

NOMENCLATURE

a_{1i}, a_{2i} are the choice function parameters;

x is a set of inlet system parameters;

x^j is a scalar parameters (continuous or discrete);

Ω is a set of admissible parameters;

z is a set of outlet system functions (parameters);

z^f is a one from output parameters;

R_S is a fuzzy binary choice relation;

R_G is a fuzzy generation relation;

$S(X)$ is a selection function;

$G(X)$ is a generation function;

$G_H(X)$ is a set of new solutions;

B_{ob} is a table of experimental results;

B is a matching matrix of experimental results;

$\Gamma(x)$ is a choice function;

N_{ob} is a number of experiments;

N_b is a number of branches for evolutionary search;

N_E is a number of new solutions;

N_{op} is a number of preferred solutions;

S_b is a burner area;

S_{pr} is an useful area for primary air;

Lp is a primary air flow;

L is a total air flow;

S, Sp are the square parameters of burner;

Y_A, Y_{CO}, Y_{NO} are the outlet parameters of burner;

X is a subset of parameters;

X_k is a set of preferred solutions according to the binary choice relation R_S at the iterate step k ;

k is an iterate step;

X_{k-1} is a set of preferred solutions according to the binary choice relation R_S at the iterate step $k-1$;

X_{jk} is a set of preferred solutions according to the binary choice relation R_S at the iterate step k for the branch j of evolutionary search;

W is a power of burner.

INTRODUCTION

The basis of the research is a fuzzy experiment in which the permissible range of parameters determining the state of the system is comprehensively investigated. In each experiment, in addition to the input parameters of the system, the output functions (criteria) of the system under

study are measured or calculated. If we confine ourselves only to the experimental sampling of fuzzy parameters, then it will not be possible to make decisions about the preference of the system parameters over the entire allowable area. It is advisable to build a mathematical model of the function of choice, which will allow to extend the rule of preferences of fuzzy parameters to the entire admissible region. Having an expression for the function of choice, we can formulate and solve the problem of finding the most preferable solutions. The search of the most preferable solutions can be implemented as a result of solving a generalized mathematical programming problem.

The object of study is the process of decision-making while developing or managing systems with some fuzzy parameters. The mathematical model of such system is used to make decisions for the development or management of systems. The mathematical model of the system can be built on the basis of deductive laws of functioning or on the basis of an experimental study of the system.

The subject of study is the process of decision-making for project or control in complex systems with multiple criteria when information about the system is presented in the form of a limited set of fuzzy experiment results.

The purpose of the work is to increase the speed the decision-making process for a system with several criteria when setting information about the properties of the system is a set of fuzzy experimental results.

1 PROBLEM STATEMENT

A system is characterized by a set of parameters $x = \{x^1, x^2, \dots, x^n\}$, $x \in \Omega$ and a set of output parameters (functions, criteria) $z = \{z^1, z^2, \dots, z^f\}$. There are training set of experimental results: $B_{ob} = \langle x_q y_q \rangle$, $q = 1, 2, \dots, N_{ob}$ and the result of the expert evaluation in the form of the of the evaluation table $B = \{b_{ij}\}$, $i = 1, 2, \dots, N_{ob}$, $j = 1, 2, \dots, N_{ob}$, which is obtained using expert choice relation \tilde{R} , where b_{ij} – is the table of fuzzy relation.

It is required to find the choice function C for all set Ω with binary relation \tilde{R}_S such that binary relation \tilde{R}_S corresponds with expert choice relation \tilde{R} , with the table of fuzzy relation $B = \{b_{ij}\}$, $i = 1, 2, \dots, N_{ob}$, $j = 1, 2, \dots, N_{ob}$ $B = \{b_{ij}\}$, $i = 1, 2, \dots, N_{ob}$, $j = 1, 2, \dots, N_{ob}$.

2 REVIEW OF THE LITERATURE

There is sufficient experience in using binary relations of choice in constructing a mechanism for choosing decisions, in particular, scientific results [1–4] and other.

If there is a system that does not have a reliable mathematical model based on deductive laws of functioning, then the inductive principles of mathematical modeling of such systems are known [5, 6] that have received significant development. In inductive modelling, according Ivakhnenko A. G. and others, various mathematical models were constructed from experimental data. In this case, it is possible to build functional dependencies for each of several output functions of the system. Having

mathematical dependencies for several output functions, you can solve the decision problem as a multi-objective optimization problem. There is a fairly large number of scientific results in the field of multi-optimized optimization [7–10].

Most of these results relate to the situation where there are mathematical models for each of the output functions – Pareto optimization. In this case, the adoption of the final decision from the set of Pareto-optimal is an additional procedure. This schema is fulfil possible.

The formulation of an optimization problem as an optimization task with respect to choice relation is an alternative approach. Previously, generalized mathematical programming problems were formulated for which solution methods were proposed [11, 12] and other. Later works are also devoted to solving the problem of generalized mathematical programming, for example [13].

Effective methods for solving optimization problems are developed on the basis of evolutionary search algorithms, for example, [14–16]. Including evolutionary algorithms useful for solving problems of generalized mathematical programming [16, 17] without the convexity condition of choice relation.

The utilization of fuzzy sets and fuzzy relations [18, 19] is a new state of investigations of complex objects.

Multicriteria Fuzzy Decision-Making is presented in [20] and results today, for example [21].

Previously, it was offered a general scheme for constructing a selection mechanism for decision-making in systems with several criteria where there is a sample of experimental results only [22]. But constructing a selection mechanism for decision-making in systems with several criteria where there is a sample of fuzzy experimental results was not presented before. The scheme of decision-making includes the solving the problem of generalized mathematical programming as the final step of the selection mechanism. For the solving the problem of generalized mathematical programming may be applied the evolution search algorithm with choice function in form of preference or in form of lock [23, 24].

3 MATERIALS AND METHODS

The system is characterized by the set of parameters $x = \{x^1, x^2, \dots, x^n\}$ and there are the set of output parameters (functions, criteria) $z = \{z^1, z^2, \dots, z^f\}$.

After experimental investigation it was obtained the set of experimental results: $B_{ob} = \langle x_q y_q \rangle$, $q = 1, 2, \dots, N_{ob}$. In This results will be called presentation according to terminology of the theory of decision making [2–4]. We assume that according to expert evaluation it was formed the result of the expert evaluation in the form of the of the evaluation table $B = \{b_{ij}\}$, $i = 1, 2, \dots, N_{ob}$, $j = 1, 2, \dots, N_{ob}$, which is obtained using expert choice relation \tilde{R} , where b_{ij} – is the table of fuzzy relation.

It is necessary to find the fuzzy binary relation \tilde{R}_S with membership function $\mu_{\tilde{R}_S}(x, x)$ that taking account the set of presentations of the training sample B_{ob} by ex-

pert evaluation of output functions for any pair of presentations, so that the binary relation $\tilde{R}_S: x_i \tilde{R}_S x_j$ for $i, j = 1, 2, \dots, N_{ob}$. The result of the expert evaluation for the comparison of the presentations with each other will be represented in the form of the matching matrix $B = \{b_{ij}\}$, $i = 1, 2, \dots, N_{ob}, j = 1, 2, \dots, N_{ob}$, where $b_{ij} \in \{\alpha_1, \alpha_2, \dots, \alpha_\zeta\}$, α_i – is a possible evaluation of pair preference.

The fuzzy binary relation \tilde{R}_S with the choice function obtained in this way can be used to search for the most preferable solutions on the entire Ω set, taking into account possible limitations as well.

The methods for solving the problems are based on the approach to the evolutionary search for R_S – optimal solutions. For subset X , $X \subset \Omega$ we denote the function of choice in the form

$$S(X) = \{x \in X | \forall y \in [X \setminus S(X)], x \tilde{R}_S y\}. \quad (1)$$

We shall assume that set $S(X)$ contains the concrete number of elements N_{op} .

We shall that for the set Ω it was determined relation \tilde{R}_G with membership function $\mu_{\tilde{R}_G}(x, y): \Omega \times \Omega \rightarrow [0, 1]$. Relation \tilde{R}_G will be termed generation relation.

For subset X , $X \subset \Omega$ we denote the function of generation in the form

$$G(X) = X \cup G_H(X). \quad (2)$$

We shall assume that set $G(X)$ contains the concrete number of elements N_E .

The algorithm to search R_S – optimal solution can be represented as

$$X_k = S(G(X_{k-1})), k = 1, 2, \dots \quad (3)$$

The iterate algorithm (3) – is the general form of evolutionary search.

According to [15–17] we will consider the decomposition

$$X_k = \bigcup_{j=1}^{N_b} X_{jk}, X_{ik} \cap X_{jk} = \emptyset, i \neq j. \quad (4)$$

The algorithm (3) takes the form

$$X_{jk} = S(G(X_{jk-1})), j = \overline{1, N_b}, k = 1, 2, \dots \quad (5)$$

These iterate algorithms (3), (5) are the general form of evolutionary search.

The evolutionary search algorithm converges to the most preferred solution of choice relation. This position has been theoretically and experimentally proven for clear choice relationships. For a fuzzy choice, this position is based on experimental results.

4 EXPERIMENTS

There are considered tubular gas heater [18]. Tubular heaters design parameters (inlet system parameters) are below:

- Burner area, S_b ;
- Useful area for primary air passage, S_{fir} ;
- Primary air flow, L_I ;
- Total air flow, L_{tot} ;
- Burner power, W .

There are criteria (outlet system functions) of the heater:

- Ash transfer by the time, A ;
- Concentration CO at exhaust gases, C_{CO} ;
- Concentration NO_x at exhaust gases, C_{NO_x} .

There are following requirements for parameters that characterize tubular heaters work: for CO it is less than 130 mg/m^3 and for NO_x – less than 250 mg/m^3 . Therefore such tags as CO and NO_x are shown at tubular heater schematically block diagram. Also such parameter as ash is typical because of strengthened primary air supply creates unintended carrying out ash from the burner. It leads to tube clogging, which degrades heat transfer and reduces tube efficiency time. Tubular heater pellets burner principle diagram is shown in Fig. 1.

For expert evaluation the rating scale was used $b_{ij} \in \{0; 0.3; 0.4; 0.5; 0.6; 0.7\}$; which make sense: {much worse; worse; slightly worse; comparable; slightly better; better; much better}.

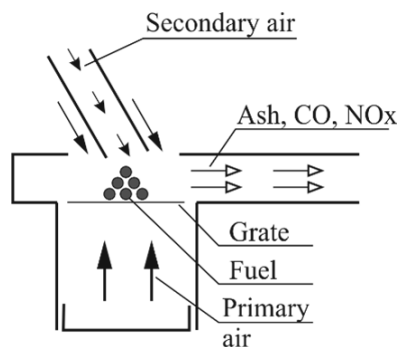


Figure 1 – Tubular heater pellets burner principle diagram

Table 1 – Matching matrix for experimental data array 1

№	S	S _p	L	L _p	W	Y _A	Y _{CO}	Y _{NOx}
	m ²	m ²	m ³ /h	m ³ /h	kW	g/min	mg/m ³	mg/m ³
1	0.005	0.00286	572.4	25.2	33.5	2.1	510	293
2	0.005	0.00286	543.6	23.4	31.3	2.88	6734	207
3	0.005	0.00286	543.6	21.6	54.7	2.77	43	259
4	0.01	0.00643	633.6	46.8	18	0.21	4500	257
5	0.01	0.00643	651.6	54	32	0.47	694	205
6	0.01	0.00643	684	50.4	35.5	5.5	110	230
7	0.0025	0.00021	201	2.7	6.4	3.57	2765	89
8	0.0025	0.00021	168	4.1	9	7	2902	134
9	0.0025	0.00021	165	4.3	18	10	7214	109
10	0.0025	0.00021	151	5.1	18	7	7844	125
11	0.0025	0.00021	215	2.2	4.7	1.6	1429	146
12	0.0025	0.00021	201	2.8	11.3	4.9	1311	193
13	0.0025	0.00021	196	3	10	5	1019	210
14	0.0025	0.00021	182	3.9	12.8	3.6	779	212
15	0.0025	0.00021	178	2.5	5.3	1.8	812	201
16	0.0025	0.00021	167	2.8	4.5	0.7	2148	160
17	0.0025	0.00021	155	3	6	1.7	722	265
18	0.0025	0.00021	150	3.5	11.2	2.8	617	259
19	0.0025	0.00021	140	4	18	5.4	1144	240
20	0.0025	0.00021	136	4.5	22.5	10.5	853	257
21	0.0025	0.00021	128	7	22.5	11.3	783	261
22	0.0025	0.00021	127	2.5	8.2	1.9	1099	134
23	0.0025	0.00021	123	3	9	1	450	188
24	0.0025	0.00021	111	3.4	11.3	1.9	246	151
25	0.0025	0.00021	105	3.8	15	3	438	190
26	0.0025	0.00021	97	4.1	15	4.8	1225	238
27	0.0025	0.00021	85	5	22.5	10.3	830	203
28	0.0025	0.00021	80	6.5	18	10.8	945	217
29	0.0025	0.00021	210	2.75	3.9	1.3	2926	161
30	0.0025	0.00021	175	4.1	9	3.4	6663	56
31	0.0025	0.00021	172	4.3	7.5	5.6	2845	148
32	0.0025	0.00021	168	5.1	18	35	1986	131
33	0.0025	0.00021	152	2.2	5	5	1826	116

Table 2 – Matching matrix for experimental data array 1. Results of experiments for 33 modes

DATA 0.5,0.3,0.5,0.6,0.7,0.7,0.5,0.7,0.5,0.7,0.7,0.7,0.4,0.4,0.4,0.5,0.6,0.7,0.7,0.7
 DATA 0.7,0.5,0.7,0.7,0.7,0.7,0.7,0.7,0.7,0.7,0.7,0.7,0.7,0.6,0.6,0.7,0.7,1.0,1.0,1.0
 DATA 0.5,0.3,0.5,0.6,0.7,0.6,0.6,0.5,0.7,0.6,0.6,0.6,0.5,0.4,0.3,0.4,0.5,0.6,1.0,1.0,1.0
 DATA 0.4,0.3,0.3,0.5,0.7,0.5,0.4,0.4,0.5,0.4,0.4,0.4,0.3,0.3,0.3,0.3,0.3,0.6,0.6,0.6
 DATA 0.3,0.3,0.3,0.3,0.5,0.3,0.3,0.3,0.4,0.3,0.4,0.3,0.3,0.3,0.0,0.3,0.4,0.4,0.5,0.5,0.5
 DATA 0.3,0.3,0.4,0.5,0.7,0.5,0.5,0.5,0.6,0.4,0.5,0.4,0.4,0.3,0.0,0.3,0.3,0.5,0.7,0.7,0.5
 DATA 0.3,0.3,0.4,0.6,0.7,0.5,0.5,0.5,0.6,0.5,0.5,0.5,0.5,0.3,0.0,0.3,0.5,0.5,0.7,0.7,0.6
 DATA 0.5,0.3,0.5,0.6,0.7,0.5,0.5,0.5,0.6,0.5,0.5,0.5,0.5,0.4,0.0,0.3,0.5,0.5,0.7,0.7,0.7
 DATA 0.3,0.3,0.3,0.5,0.6,0.4,0.4,0.4,0.5,0.4,0.4,0.4,0.4,0.3,0.0,0.3,0.4,0.4,0.5,0.5,0.5
 DATA 0.5,0.3,0.4,0.6,0.7,0.6,0.5,0.5,0.6,0.5,0.6,0.5,0.5,0.3,0.0,0.3,0.5,0.5,0.7,0.7,0.7
 DATA 0.3,0.3,0.4,0.6,0.7,0.6,0.5,0.5,0.6,0.4,0.5,0.5,0.4,0.3,0.0,0.3,0.5,0.5,0.6,0.6,0.6
 DATA 0.3,0.3,0.4,0.6,0.7,0.6,0.5,0.5,0.6,0.5,0.5,0.5,0.5,0.3,0.0,0.3,0.5,0.5,0.6,0.6,0.6
 DATA 0.3,0.3,0.5,0.6,0.7,0.6,0.5,0.5,0.6,0.5,0.6,0.5,0.5,0.3,0.0,0.3,0.5,0.5,0.6,0.6,0.6
 DATA 0.6,0.3,0.6,0.7,0.7,0.7,0.6,0.7,0.7,0.7,0.7,0.5,0.3,0.5,0.6,0.7,1.0,1.0,1.0
 DATA 0.6,0.4,0.7,0.7,1.0,1.0,1.0,1.0,1.0,1.0,1.0,1.0,0.7,0.5,0.6,0.7,0.7,1.0,1.0,1.0

DATA 0.6,0.4,0.6,0.7,0.7,0.7,0.7,0.7,0.7,0.7,0.7,0.7,0.7,0.7,0.7,0.5,0.4,0.5,0.6,0.6,1.0,1.0,1.0
 DATA 0.5,0.3,0.5,0.7,0.6,0.7,0.5,0.5,0.6,0.5,0.5,0.5,0.5,0.4,0.3,0.4,0.5,0.5,0.7,0.7,0.7
 DATA 0.4,0.3,0.4,0.7,0.6,0.5,0.5,0.5,0.6,0.5,0.5,0.5,0.5,0.3,0.3,0.4,0.5,0.6,0.6,0.6
 DATA 0.3,0.0,0.0,0.4,0.5,0.3,0.3,0.3,0.5,0.3,0.4,0.4,0.4,0.0,0.0,0.0,0.3,0.4,0.5,0.5,0.5
 DATA 0.3,0.0,0.0,0.4,0.5,0.5,0.4,0.3,0.5,0.3,0.4,0.4,0.4,0.0,0.0,0.0,0.3,0.4,0.5,0.5,0.5
 DATA 0.3,0.0,0.0,0.4,0.5,0.5,0.4,0.3,0.5,0.3,0.4,0.4,0.4,0.0,0.0,0.0,0.3,0.4,0.5,0.5,0.5

Table 3 – Matching matrix for experimental data array 2

DATA 0.5,0.5,0.0,0.3,0.5,0.3,0.3,0.3,0.3,0.5,0.3
 DATA 0.5,0.5,0.0,0.5,0.6,0.3,0.3,0.3,0.3,0.6,0.4
 DATA 1.0,1.0,0.5,1.0,1.0,1.0,1.0,1.0,1.0,1.0,1.0,1.0
 DATA 0.7,0.5,0.0,0.5,0.6,0.3,0.3,0.3,0.4,0.4,0.6,0.6
 DATA 0.5,0.4,0.0,0.4,0.5,0.3,0.3,0.3,0.3,0.5,0.4
 DATA 0.7,0.7,0.0,0.7,0.7,0.5,0.5,0.4,0.5,0.5,0.6,0.5
 DATA 0.7,0.7,0.0,0.7,0.7,0.5,0.5,0.5,0.5,0.6,0.7,0.5
 DATA 0.7,0.7,0.0,0.7,0.7,0.5,0.5,0.5,0.5,0.4,0.7,0.6
 DATA 0.7,0.7,0.0,0.6,0.5,0.5,0.5,0.5,0.5,0.6,0.6
 DATA 0.7,0.7,0.0,0.6,0.7,0.5,0.4,0.6,0.5,0.5,0.6,0.5
 DATA 0.5,0.4,0.0,0.4,0.5,0.4,0.3,0.3,0.4,0.4,0.5,0.3
 DATA 0.7,0.6,0.0,0.4,0.6,0.5,0.5,0.4,0.4,0.5,0.7,0.5

5 RESULTS

There are presented results with choice function in the form (10).

$$\Gamma(x) = \prod_{i=1}^5 \cdot (1 + a_{1i} \cdot (a_{2i} - r_i)^2) \tag{10}$$

$$\begin{aligned} r_1 &= x_1^1 - x_2^1; r_2 = x_1^2 - x_2^2; r_3 = x_1^3 - x_2^3; \\ r_4 &= x_1^4 - x_2^4; r_5 = x_1^5 - x_2^5; \end{aligned} \tag{11}$$

$$\Gamma(x_1) \geq \Gamma(x_2) \equiv x_1 \cdot \tilde{R} \cdot x_2. \tag{12}$$

Parameters a_{1i} , a_{2i} were obtained after evolutionary search the choice function for array 1 of experimental data and for array 2 of experimental data.

The choice function in the form (10) with specific values of parameters A_1, A_2, \dots, Aa_6 was used to solve the problem of generalized mathematical programming: to find maximum $\text{MAX} \mu_{\tilde{R}_s}(x,y)$ of choice function with restrictions: $0.002 \leq x_1 \leq 0.1$; $0.0002 \leq x_2 \leq 0.007$; $80 \leq x_3 \leq 650$; $2 \leq x_4 \leq 50$; $4 \leq x_5 \leq 55$

The aim of this paper is to summarize the present position of fuzzy MCDM research area. This summary includes the classification of a mathematical decision-making model, the distributions of publications with respect to their subject areas, publication years, citation frequencies, authors, and publishing journals. We also

classify the studies into three groups: The first group develops new fuzzy methodologies or modifies the existing approaches; second group uses the existing approaches in a specific problem area. Third group integrates different MCDM techniques. It also presents expected future trends on fuzzy MCDM.

Step 1: Determine key criteria and sub-criteria for a comprehensive assessment of the potential supplier. At this stage, the identification of key criteria and sub-criteria is based on a review of the literature, SCOR metrics and scientific reports related to the content of the research to determine the necessary criteria for the topic.

Step 2: Hybrid fuzzy set theory into the ANP model is the most effective tool for addressing complex problems of decision-making, which has a connection with various qualitative criteria. There are ten suppliers that are highly effective for providing plastic raw materials. In this step, an FANP is proposed to identify the weight of all criteria.

Step 3: For ranking potential suppliers list, VIKOR model is used in the final stage. VIKOR ranks alternatives and determines the solution named compromise that is the closest to the ideal.

Evolutionary search for solving the problem of generalized mathematical programming is illustrated at Table 5 and results of evolutionary search for three branches of evolution are presented in Table 6.

Table 4 – Evolutionary search the choice function

Iteration step of evolutionary search	Error at the training array 1	Error at the test array 2
1	0.3822	
2	0.3679	
4	0.2774	
8	0.1612	
15	0.1612	
20	0.1552	
25	0.1433	
35	0.1432	
75	0.1430	
500	0.1401	0.1646

Table 5 – Evolutionary search for solving the problem of generalized mathematical programming

Iteration step of evolution	Maximum function Branch 1 of evolution	Maximum function Branch 2 of evolution	Maximum function Branch 3 of evolution
1	0.7462	0.7247	0.5972
2	0.8294	0.9682	0.7936
3	0.9727	0.9721	0.9353
...			
10	0.9727	0.9721	0.9691
300	0.9839	0.9837	0.9787

Table 6 – Result of evolutionary search for solving the problem of generalized mathematical programming

Branches of evolution	Parameter 1 x^1	Parameter 2 x^2	Parameter 3 x^3	Parameter 4 x^4	Parameter 5 x^5
Branch 1	0.00517	0.000508	251.6	5.297	16.73
Branch 2	0.00499	0.000537	273.0	6.809	15.98
Branch 3	0.00524	0.000525	186.5	2.000	8.192

6 DISCUSSION

Matrix of conformity (Table 3 and Table 4) is the result of expert assessment and is subjective. For expert assessment, it is obviously possible to use the whole variety of available pair-wise comparison methods and use a different scale for such an assessment. The choice function given in the article in the form of an algebraic function is certainly not the only possible one, here you can use the whole variety of pattern recognition methods.

At Literature Review [20] are presented different methods of mathematical decision-making models among them: Multicriteria decision-making (MCDM), fuzzy multiattribute decision-making (MADM), fuzzy multiobjective decision-making (MODM).

In [21] are presented the detail realisation of Fuzzy Multicriteria Decision-Making Model (MCDM). There are three stages of the mathematical modelling:

Step 1: At this stage it was determine key criteria and sub-criteria for a comprehensive assessment of the potential supplier. The identification of key criteria and sub-criteria is based on a review of the literature, SCOR metrics and scientific reports related to the content of the research to determine the necessary criteria.

Step 2: At this stage are applied hybrid fuzzy set theory into the ANP model for the most effective tool, which has a connection with various qualitative criteria. In this step, an FANP is proposed to identify the weight of all criteria.

Step 3: At the final stage the VIKOR model is used. VIKOR ranks alternatives and determines the solution named compromise that is the closest to the ideal.

As you can see, in the set of multicriteria decision-making methods (MCDM), the results of the experimental study of the object and their expert assessment are not used directly. The resulting compromise solution gives the final solution, but if the requirements for the compromise solution change, then the sequence of steps 1–3 must be repeated, that is, the resulting model is difficult to use in the research process. The proposed decision-making approach based on evolutionary search algorithms allows not only obtaining a model, but also makes it possible to solve new problems on its basis, using evolutionary search as a solution to a stochastic programming problem.

Comparison of the results of the development of a mathematical decision-making model based on a finite set

of experimental results using the approach of fuzzy experiments and without this approach showed that using the approach of fuzzy experiments it is possible to increase the adequacy of the mathematical decision-making model and reduce the error of erroneous recognition of results for the verification (control) sequence.

At this stage it was determine key criteria and sub-criteria for a comprehensive assessment of the potential supplier. The identification of key criteria and sub-criteria is based on a review of the literature, SCOR metrics and scientific reports related to the content of the research to determine the necessary criteria.

At this stage are applied hybrid fuzzy set theory into the ANP model for the most effective tool, which has a connection with various qualitative criteria. In this step, an FANP is proposed to identify the weight of all criteria.

At the final stage the VIKOR model is used. VIKOR ranks alternatives and determines the solution named compromise that is the closest to the ideal.

As you can see, in the set of Multicriteria decision-making (MCDM) methods, the results of the experimental study of the object and their expert assessment are not used directly. The resulting compromise solution gives the final solution, but if the requirements for the compromise solution change, then the sequence of steps 1–3 must be repeated, that is, the resulting model is difficult to use in the research process. The proposed decision-making approach based on evolutionary search algorithms allows not only obtaining a model, but also makes it possible to solve new problems on its basis, using evolutionary search as a solution to a stochastic programming problem.

CONCLUSIONS

Thus, the selection mechanism extends to the entire allowable range of input parameters. The constructed selection function is determined already on the whole admissible space of input parameters, and not only on the set of experimental points.

The decision-making includes the following procedures: an experimental study of a technical system with several fuzzy criteria (functions) depending on its parameters; the use of expert evaluation to build a matrix of preferences for individual implementations; building a function of choosing preferred solutions based on a preference matrix by constructing a mathematical model of preference recognition, formulation and solving the prob-

lem of generalized mathematical programming as the final step in building the selection mechanism.

The scientific novelty is result presented as a holistic decision-making mechanism for a fuzzy system based on inductive modelling of complex systems, in which the following steps can be distinguished: an experimental study of a process with several fuzzy criteria (functions) depending on its parameters; the use of expert evaluation to build a matrix of preferences for individual implementations; building a function of choosing preferred solutions based on a preference matrix by constructing a mathematical model of preference recognition, formulation and solving the problem of generalized mathematical programming as the final step in building the selection mechanism.

The practical significance of obtained results is that the stated decision-making mechanism can be used for a wide range of complex fuzzy systems with several criteria.

Prospects for further research are to the improvement of methods and means for constructing a function of choice for a limited number of fuzzy experimental results.

ACKNOWLEDGMENT

The main part of the work was carried out within the framework of the budget research work of the Pridneprovskaya State Academy of Civil Engineering and Architecture (state registration number 0112U005350).

REFERENCES

1. Fishburn P. Representable choice function, *Econometrica*, 1976, Vol. 44, No. 5, pp. 1033–1043. <https://www.jstor.org/stable/1911543>
2. Aizerman M. A. Some new problems in the general theory of choice (one line of research), *Autom. Remote Control*, 1984, pp. 1103–1135. <https://zbmath.org/?q=an:0569.90003>
3. Aizerman M. A., Litvakov B. M. On some extensions of the option choice theory (fundamentals of the pseudocriterion theory), *Autom. Remote Control*, 1988, Vol. 49, Issue 3, pp. 337–347. <https://zbmath.org/?q=an:0665.90003>
4. Sholomov L. A., Yudin D. B. Design of multistep choice schemes, *Autom. Remote Control*, 1986, Vol. 47, Issue 10, pp. 1414–1424. <https://zbmath.org/?q=an:0615.90088>
5. Ivakhnenko A.G. Heuristic Self-Organization in Problems of Engineering Cybernetics, *Automatica*, 1970, Vol. 6, pp. 207–219.
6. Ivakhnenko A.G. Polynomial Theory of Complex Systems, *IEEE Transactions on Systems Man and Cybernetics*, 1971, Vol. 4, pp. 364–378.
7. Lemarchand L., Masse D., Rebreyend P., et al. Multiobjective optimization for multimode transportation problems, *Advances in Operations Research*, 2018, Vol. 2018, 13 p. <https://doi.org/10.1155/2018/8720643>
8. Sagawa M., Kusuno N., Aguirre H. et al. Evolutionary multiobjective optimization including practically desirable solutions, *Advances in Operations Research*, 2017, Vol. 2017, 16 p. <https://doi.org/10.1155/2017/9094514>
9. Giagkiozis I., Fleming P. J. Pareto front estimation for decision making, *Evolutionary computation*, 2014, Vol. 22, No. 4, pp. 651–678. <https://www.researchgate/publication/261369702>
10. Wang Y., Sun X. A many-objective optimization algorithm based on weight vector adjustment, *Computational Intelligence and Neuroscience*, 2018, Vol. 2018, 21 p. DOI: 10.1155/2018/4527968.
11. Yudin D. B. Methods of generalized convex programming and estimation of their complexity, *Dokl. Akad. Nauk SSSR*, 272:1, 1983, pp. 40–43. <https://www.ams.org/mathscinet-getitem?mr=725061>
12. Yudin D. B., Sholomov L. A. Multistep schemes of generalized mathematical programming and the choice function", *Dokl. Akad. Nauk SSSR*, 1985, Vol. 282:5, pp. 1066–1069. <https://www.ams.org/mathscinet-getitem?mr=796945>
13. Kolbin V. V. Generalized mathematical programming as a decision model, *Applied Mathematical Sciences*, 2014, Vol. 8, No. 70, pp. 3469–3476. DOI 10.12988/ams.2014.44231.
14. Irodov V. F., Maksimenkov V. P. Application of an evolutionary program for solving the travelling-salesman problem, *Sov. Autom. Control; translation from Avtomatika*, 1981, Vol. 4, pp. 7–10. <https://zbmath.org/0508.90066>
15. Irodov V. F. The construction and convergence of evolutionary algorithms of random search for self-organization, *Sov. J. Autom. Inf. Sci. 20 ; translation from Avtomatika*, 1987, Vol. 4, pp. 34–43. <https://zbmath.org/0656.90087>
16. Irodov V. F. Self-organization methods for analysis of nonlinear systems with binary choice relations, *System Analysis Modeling Simulation*, 1995, Vol. 18–19, pp. 203–206. <https://dl.acm.org/citation.cfm?id=208028#>
17. Irodov V. F., Khatskevych Yu. V. Convergence of evolutionary algorithms for optimal solution with binary choice relations, *Stroitel'stvo. Materialovedeniye. Mashinostroyeniye. Seriya : Energetika, ekologiya, komp'yuternyye tekhnologii v stroitel'stve*, 2017, Vol. 98, pp. 91–96. http://nbuv.gov.ua/UJRN/smmeect_2017_98_16
18. Zadeh L. A. Fuzzy Sets, *Inform. Control*, 1965, vol. 8, pp. 338–353.
19. Bellman R. E., Zadeh L. A. Decision Making in Fuzzy Environment. *Management Science, Environment. Management Science*, 1970, Vol. 17, No 4, P.141–164.
20. Kahraman C., Cevik Onar S., Oztaysi B. Fuzzy Multicriteria Decision-Making, *A Literature Review. International Journal of Computational Intelligence Systems*, 2015, Vol. 8, No. 4, pp. 637–666. <https://doi.org/10.1080/18756891.2015.1046325>
21. Wang C. N., Ngyen V. T., Chuoy J. T. Fuzzy Multicriteria Decision-Making Model (MCDM) for Raw Materials Supplier Selection in Plastics Industry, *Mathematics*, 2019, Vol. 7, pp. 981–1017. <https://www.mdpi.com/2227-7390/7/10/981>
22. Irodov V. F., Barsuk R. V. Decision-making Limited Number of Experiments With Multiple Criteria, *Radio Electronics, Computer Science, Control*, 2020, No. 1, pp. 200–208. <https://doi.org/10.15588/1607-3274-2020-1-20>
23. Irodov V. F., Chornomorets G. Y., Barsuk R. V. Multiobjective Optimization at Evolutionary Search with Binary Choice Relation, *Cybernetics and Systems Analysis*, 2020, No. 56(3), pp. 449–454. <https://doi.org/10.1007/s10559-020-00260-7>
24. Irodov V. F., Barsuk R. V., Chornomorets G. Ya. et al. Experimental Simulation and Multiobjective Optimization of the Work of a Pellet Burner for a Tubular Gas Heater, *Journal of Engineering Physics and Thermophysics*, 2021, Vol. 94, pp. 227–233. <https://doi.org/10.1007/s10891-021-02290-0>

Received 19.08.2021.
Accepted 30.12.2021.

УДК 519.816

ПРИЙНЯТТЯ РІШЕНЬ ПРИ ЕВОЛЮЦІЙНОМУ ПОШУКУ ПРИ ОБМЕЖЕНІЙ КІЛЬКОСТІ НЕЧІТКИХ ЕКСПЕРИМЕНТІВ З КІЛЬКОМА КРИТЕРІЯМИ

Іродов В. Ф. – д-р техн. наук, професор, професор кафедри інформаційних технологій Дніпровського технологічного університету «Шаг», Дніпро, Україна.

Шаптала М. В. – канд. техн. наук., доцент, ректор Дніпровського технологічного університету «Шаг», Дніпро, Україна.

Дудкін К. В. – канд. техн. наук, директор ТОВ «КВ-автоматизація», Київ, Україна.

Шаптала Д. Е. – канд. техн. наук, доцент кафедри інформаційних технологій Дніпровського технологічного університету «Шаг», Дніпро, Україна.

Чирін Д. А. – аспірант кафедри системного аналізу та моделювання в теплопостачанні, Придніпровська державна академія будівництва та архітектури, Дніпро, Україна.

АНОТАЦІЯ

Актуальність. Розглянуто механізм прийняття рішень під час обмеженої кількості нечітких експериментів з кількома критеріями. Об'єктом дослідження є процес прийняття рішень щодо проекту або управління в складних системах з декількома критеріями.

Мета. Необхідно визначити оптимальні (найбільш переважні) параметри систем з кількома критеріями, користуючись не математичною моделлю системи, а лише обмеженою кількістю нечітких експериментів.

Метод. Експериментальне дослідження процесу з кількома критеріями (функціями) залежно від його параметрів; використання експертної нечіткої оцінки для побудови матриці уподобань для окремих реалізацій; побудова функції вибору кращих рішень на основі матриці переваг шляхом побудови математичної моделі розпізнавання переваг, формулювання та вирішення проблеми узагальненого математичного програмування як завершального етапу побудови механізму відбору. Механізм прийняття рішень залежить від процедури експертної оцінки при порівнянні обмеженого набору результатів між собою, а також від постановки умов при вирішенні задачі узагальненого математичного програмування. Порівняння кінцевої кількості нечітких експериментів зручно для експертної оцінки. Презентація остаточного вибору в результаті вирішення задачі узагальненого математичного програмування зручна для використання такого механізму в системах автоматичного управління вже без участі людини. Запропонована схема прийняття рішень під час обмеженої кількості нечітких експериментів була застосована до прийняття рішень з управління проектами паливних гранул.

Результати. Нечіткі результати експериментального прийняття рішень представлені за наявності кількох критеріїв для пелетного паливника трубчастого нагрівача, які підтверджують прийнятність розробленого механізму прийняття рішень. Запропоновано нову схему побудови механізму відбору для прийняття рішень в системах з декількома критеріями, де є вибірка нечітких експериментальних результатів.

Висновки. Схема прийняття рішень включає вирішення узагальненого математичного програмування як завершальний етап побудови механізму відбору. Для проблеми узагальненого математичного програмування може бути застосований еволюційний пошук з функцією вибору у формі переваги або у формі блокування.

КЛЮЧОВІ СЛОВА: прийняття рішень, множинні критерії, нечіткі експерименти, функція вибору, еволюційний пошук.

УДК 519.816

ПРИНЯТИЕ РЕШЕНИЙ ПРИ ЭВОЛЮЦИОННОМ ПОИСКЕ ПРИ ОГРАНИЧЕННОМ КОЛИЧЕСТВЕ НЕЧЕТКИХ ЭКСПЕРИМЕНТОВ С НЕСКОЛЬКИМ КРИТЕРИЯМИ

Іродов В. Ф. – д-р техн. наук, професор, професор кафедри інформаційних технологій Дніпровського технологічного університету «Шаг», Дніпро, Україна.

Шаптала М. В. – канд. техн. наук, доцент, ректор Дніпровського технологічного університету «Шаг», Дніпро, Україна.

Дудкін К. В. – канд. техн. наук, директор ООО «КВ-автоматизація», Київ, Україна.

Шаптала Д. Е. – канд. техн. наук, доцент кафедри інформаційних технологій Дніпровського технологічного університету «Шаг», Дніпро, Україна.

Чирін Д. А. – аспірант кафедри системного аналізу та моделювання в теплоснабженні, Придніпровська державна академія будівництва та архітектури, Дніпро, Україна.

АННОТАЦИЯ

Актуальность. Рассмотрен механизм принятия решений во время ограниченного количества нечетких экспериментов с несколькими критериями. Объектом исследования является процесс принятия решений по проекту или управления в составных системах с несколькими критериями.

Цель. Необходимо определить оптимальные (наиболее предпочтительные) параметры систем с несколькими критериями, используя не математическую модель системы, а лишь ограниченное количество нечетких экспериментов.

Метод. Экспериментальное исследование процесса с несколькими критериями (функциями) в зависимости от его параметров; использование экспертной нечеткой оценки для построения матрицы предпочтений для отдельных реализаций; построение функции выбора лучших решений на основе матрицы преимуществ путем построения математической модели распознавания преимуществ, формулирования и решения проблемы обобщенного математического программирования как заключительного этапа построения механизма отбора. Механизм принятия решений зависит от процедуры экспертной оценки при сравнении ограниченного набора результатов между собой, а также от постановки условий при решении задачи обобщенного математического программирования. Сравнение конечного числа нечетких экспериментов удобно для экспертной оценки. Презентация окончательного выбора в результате решения задачи обобщенного математического программирования

удобна для использования такого механизма в системах автоматического управления уже без участия человека. Предложенная схема принятия решений во время ограниченного количества нечетких экспериментов была применена к принятию решений по управлению проектами топливных гранул.

Результаты. Нечеткие результаты экспериментального принятия решений представлены при наличии нескольких критериев для пеллетного горелки трубчатого нагревателя, подтверждающие приемлемость разработанного механизма принятия решений. Предложена новая схема построения механизма отбора для принятия решений в системах с несколькими критериями, где есть выборка нечетких экспериментальных результатов. Выводы. Схема принятия решений включает решение обобщенного математического программирования как завершающий этап построения механизма отбора. Для проблемы обобщенного математического программирования может быть применен эволюционный поиск с функцией выбора в форме преимущества или в форме блокировки.

КЛЮЧЕВЫЕ СЛОВА: принятие решений, множественные критерии, нечеткие эксперименты, функция выбора, эволюционный поиск.

ЛІТЕРАТУРА / LITERATURA

1. Fishburn P. Representable choice function / P. Fishburn. – *Econometrica*. – 1976. – Vol. 44, № 5. – P. 1033–1043. <https://www.jstor.org/stable/1911543>
2. Aizerman M. A. Some new problems in the general theory of choice (one line of research) / M. A. Aizerman // *Autom. Remote Control*. – 1984. – P. 1103–1135. <https://zbmath.org/?q=an:0569.90003>
3. Aizerman M. A. On some extensions of the option choice theory (fundamentals of the pseudocriterion theory) / M. A. Aizerman, B. M. Litvakov // *Autom. Remote Control*. – 1988. – Vol. 49, Issue 3. – P. 337–347. <https://zbmath.org/?q=an:0665.90003>
4. Sholomov L. A. Design of multistep choice schemes / L. A. Sholomov, D. B. Yudin // *Autom. Remote Control*. – 1986. – Vol. 47, Issue 10. – P. 1414–1424. <https://zbmath.org/?q=an:0615.90088>
5. Ivakhnenko A.G. Heuristic Self-Organization in Problems of Engineering Cybernetics / A.G. Ivakhnenko // *Automatica*. – 1970. – Vol. 6. – P. 207–219.
6. Ivakhnenko A.G. Polynomial Theory of Complex Systems / A.G. Ivakhnenko // *IEEE Transactions on Systems Man and Cybernetics*. – 1971. – Vol. 4. – P.364–378.
7. Multiobjective optimization for multimode transportation problems / [L. Lemarchand, D. Masse, P. Rebreyend, et al.] // *Advances in Operations Research*. – 2018. – Vol. 2018. – 13 p. <https://doi.org/10.1155/2018/8720643>
8. Evolutionary multiobjective optimization including practically desirable solutions / [M. Sagawa, N. Kusuno, H. Aguirre, et al.] // *Advances in Operations Research*. – 2017. – Vol. 2017. – 16 p. <https://doi.org/10.1155/2017/9094514>.
9. Giagkiozis I. Pareto front estimation for decision making / I. Giagkiozis, P. J. Fleming // *Evolutionary computation*. – 2014. – Vol. 22, № 4 – P. 651–678. <https://www.researchgate/publication/261369702>.
10. Wang Y. A many-objective optimization algorithm based on weight vector adjustment / Y. Wang, X. Sun // *Computational Intelligence and Neuroscience*. – 2018. – Vol. 2018. – 21 p. DOI: 10.1155/2018/4527968.
11. Yudin D. B. Methods of generalized convex programming and estimation of their complexity / D. B. Yudin // *Dokl. Akad. Nauk SSSR*. – 1983. – 272:1. – P. 40–43. <https://www.ams.org/mathscinet-getitem?mr=725061>
12. Yudin D. B. Multistep schemes of generalized mathematical programming and the choice function // D. B. Yudin, L. A. Sholomov // *Dokl. Akad. Nauk SSSR*. – 1985. – Vol. 282:5 – P. 1066–1069. <https://www.ams.org/mathscinet-getitem?mr=796945>
13. Kolbin V. V. Generalized mathematical programming as a decision model / V. V. Kolbin // *Applied Mathematical Sciences*. – 2014. – Vol. 8, № 70 – P. 3469–3476. DOI 10.12988/ams.2014.44231.
14. Irodov V. F. Application of an evolutionary program for solving the travelling-salesman problem / V. F. Irodov, V. P. Maksimenkov // *Sov. Autom. Control; translation from Avtomatika*. – 1981. – Vol. 4. – P. 7–10. <https://zbmath.org/0508.90066>
15. Irodov V. F. The construction and convergence of evolutionary algorithms of random search for self-organization / V. F. Irodov // *Sov. J. Autom. Inf. Sci.* 20 ; translation from Avtomatika. – 1987. – Vol. 4 – P. 34–43. <https://zbmath.org/0656.90087>
16. Irodov V. F. Self-organization methods for analysis of nonlinear systems with binary choice relations // *System Analysis Modeling Simulation* – 1995. – Vol. 18–19 – P. 203–206. <https://dl.acm.org/citation.cfm?id=208028#>.
17. Irodov V. F. Convergence of evolutionary algorithms for optimal solution with binary choice relations / V. F. Irodov, Yu. V. Khatskevych // *Stroitel'stvo. Materialovedeniye. Mashinostroyeniye. Seriya : Energetika, ekologiya, komp'yuternyye tekhnologii v stroitel'stve*. – 2017. – Vol. 98. – P. 91–96. http://nbuv.gov.ua/UJRN/smmeect_2017_98_16.
18. Zadeh L. A. Fuzzy Sets / L. A. Zadeh // *Inform. Control*. – 1965. – Vol. 8. – P. 338–353.
19. Bellman R. E. Decision Making in Fuzzy Environment. *Management Science* // R. E Bellman, L. A. Zadeh // *Environment. Management Science*. – 1970. – Vol. 17, No. 4. – P.141–164.
20. Kahraman C. Fuzzy Multicriteria Decision-Making / C. Kahraman, S. Cevik Onar, B. Oztaysi // *A Literature Review. International Journal of Computational Intelligence Systems*. – 2015. – Vol. 8, No. 4. – P. 637–666. <https://doi.org/10.1080/18756891.2015.1046325>
21. Wang C. N. Fuzzy Multicriteria Decision-Making Model (MCDM) for Raw Materials Supplier Selection in Plastics Industry / C. N. Wang, V. T. Ngyen, J. T Chuoy // *Mathematics*. – 2019. – Vol. 7 – P. 981–1017. <https://www.mdpi.com/2227-7390/7/10/981>
22. Irodov V. F. Decision-making Limited Number of Experiments With Multiple Criteria / V. F. Irodov, R. V. Barsuk // *Radio Electronics, Computer Science, Control*. – 2020. – No. 1. – P. 200–208. <https://doi.org/10.15588/1607-3274-2020-1-20>.
23. Irodov V. F. Multiobjective Optimization at Evolutionary Search with Binary Choice Relation / V. F. Irodov, G. Y. Chornomorets, R. V. Barsuk // *Cybernetics and Systems Analysis*. – 2020. – No. 56(3). – P. 449–454. <https://doi.org/10.1007/s10559-020-00260-7>.
24. Experimental Simulation and Multiobjective Optimization of the Work of a Pellet Burner for a Tubular Gas Heater / [V. F. Irodov, R. V Barsuk, G. Ya. Chornomorets et al.] // *Journal of Engineering Physics and Thermophysics*. – 2021. – Vol. 94. – P. 227–233. <https://doi.org/10.1007/s10891-021-02290-0>.

Наукове видання

**Радіоелектроніка,
інформатика,
управління**

№ 1/2022

Науковий журнал

Головний редактор – д-р техн. наук С. О. Субботін

Заст. головного редактора – д-р техн. наук Д. М. Піза

Комп'ютерне моделювання та верстання
Редактор англійських текстів

С. В. Зуб
С. О. Субботін

Оригінал-макет підготовлено у редакційно-видавничому відділі НУ «Запорізька політехніка»

Свідоцтво про державну реєстрацію
КВ № 24220-14060 ПР від 19.11.2019.

*Підписано до друку 28.03.2022. Формат 60×84/8.
Папір офс. Різогр. друк. Ум. друк. арк. 20,46.
Тираж 300 прим. Зам. № 255.*

69063, м. Запоріжжя, НУ «Запорізька політехніка», друкарня, вул. Жуковського, 64

Свідоцтво суб'єкта видавничої справи
ДК № 6952 від 22.10.2019.