UDC 004.891.032.26:629.7.01.066

IMAGE CLASSIFIER RESILIENT TO ADVERSARIAL ATTACKS, FAULT INJECTIONS AND CONCEPT DRIFT – MODEL ARCHITECTURE AND TRAINING ALGORITHM

Moskalenko V. V. – PhD, Associate Professor, Associate professor of Computer Science department, Sumy State University, Sumy, Ukraine.

Moskalenko A. S. – PhD, Senior lecturer of Computer Science department, Sumy State University, Sumy, Ukraine.

Korobov A. G. – PhD, Senior lecturer of Computer Science department, Sumy State University, Sumy, Ukraine.

Zaretsky M. O. – Postgraduate student of Computer Science department, Sumy State University, Sumy, Ukraine.

ABSTRACT

Context. The problem of image classification algorithms vulnerability to destructive perturbations has not yet been definitively resolved and is quite relevant for safety-critical applications. Therefore, object of research is the process of training and inference for image classifier that functioning under influences of destructive perturbations. The subjects of the research are model architecture and training algorithm of image classifier that provide resilience to adversarial attacks, fault injection attacks and concept drift.

Objective. Stated research goal is to develop effective model architecture and training algorithm that provide resilience to adversarial attacks, fault injections and concept drift.

Method. New training algorithm which combines self-knowledge distillation, information measure maximization, class distribution compactness and interclass gap maximization, data compression based on discretization of feature representation and semi-supervised learning based on consistency regularization is proposed.

Results. The model architecture and training algorithm of image classifier were developed. The obtained classifier was tested on the Cifar10 dataset to evaluate its resilience over an interval of 200 mini-batches with a training and test size of mini-batch equals to 128 examples for such perturbations: adversarial black-box $L\infty$-attacks with perturbation levels equal to 1, 3, 5 and 10; inversion of one randomly selected bit in a tensor for 10%, 30%, 50% and 60% randomly selected tensors; addition of one new class; real concept drift between a pair of classes. The effect of the feature space dimensionality on the value of the information criterion of the model performance without perturbations and the value of the integral metric of resilience during the exposure to perturbations is considered.

Conclusions. The proposed model architecture and learning algorithm provide absorption of part of the disturbing influence, graceful degradation due to hierarchical classes and adaptive computation, and fast adaptation on a limited amount of labeled data. It is shown that adaptive computation saves up to 40% of resources due to early decision-making in the lower sections of the model, but perturbing influence leads to slowing down, which can be considered as graceful degradation. A multi-section structure trained using knowledge self-distillation principles has been shown to provide more than 5% improvement in the value of the integral mectric of resilience compared to an architecture where the decision is made on the last layer of the model. It is observed that the dimensionality of the feature space noticeably affects the resilience to adversarial attacks and can be chosen as a tradeoff between resilience to perturbations and efficiency without perturbations.

KEYWORDS: image classification, robustness, resilience, graceful degradation, adversarial attacks, faults injection, concept drift.

ABBREVIATIONS

CE is a Cross-Entropy Function;

CIFAR is a Canadian Institute for Advanced Research dataset;

CMA-ES is the covariance matrix adaptation evolution strategy optimization algorithm;

CNN is a Convolutional Neural Network;

GAN is a Generative Adversarial Network;

FIFO is a First In, First Out queue organization;

MED is a Median value of array;

IRQ is a Interquartile Range value;

KL is a Kullback-Leibler divergence function.

NOMENCLATURE

D_U is the unlabeled images for training and testing;

D_S is the labeled images for training and testing;

n is a number of unlabeled examples;

K is a size of set of classes;

n_k is a number of labeled examples of k-th class;

e_{ξ_1} is a ξ_1-th parameter which impacts on feature representation;

f_{ξ_2} is a ξ_2-th parameter which impacts on efficiency of decision rules;

α_k is a false positive rate for k-th class;

β_k is a false negative rate for k-th class;

$D_{1,k}$ is a true positive rate or sensitivity for k-th class;

$D_{2,k}$ is a true negative rate or specificity for k-th class;

J is a function of information criteria;

$\overline{J}$ is a class-wise averaged value of information-based classifier efficiency criterion;

$\overline{J}_0$ is a performance at normal functioning that introduced for mapping integral metric of resilience to a value between 0 and 1;

DOI 10.15588/1607-3274-2022-3-9

T_c control time period which can be set a priori and estimated as the mean time between adverse events or maximum allowable recovery time;

G a search domain for optimal parameter values;

T a confidence threshold;

η a coefficient to regulate tradeoff between performance without perturbation and resilience under perturbations;

μ_k membership function that represent confidence in the forecast of input sample belonging to the k -th class;

z_i is a binary feature representation of i -th example at the feature extractor output;

$dist(\cdot)$ is a Euclidean Squared distance;

$\bar{z}_k$ is a trainable k -th class prototype;

N is a dimension of high-level feature space;

r_k is a trainable scale factor for radius of hyperspherical decision boundary (container) of k -th class, $r_k \in (0;1)$;

L_{INF} is a loss function based on information criterion;

H_o is a priori entropy for two alternative decision systems;

H_γ is a posteriori entropy, which characterizes the residual uncertainty after decision-making;

TP_k is a numbers of true positives for decision rule of k -th class;

TN_k is a numbers of true negatives for decision rule of k -th class;

FP_k is a numbers of false positives for decision rule of k -th class;

FN_k is a numbers of false negatives for decision rule of k -th class;

ε is a constant added for numerical stability, $\varepsilon = 10^{-6}$;

y_i is class labels for *i*-th example after one-hot encoding;

n_{MB} is a size of mini-batch;

$\hat{y}_i$ is the value of the smoothed membership function for the *i*-th sample to each class;

relu is an activation function RELU;

$\odot$ is the component-wise multiplication sign (Hadamard product);

$\bar{d}$ is the averaged value of the normalized distance between the prototypes of the classes;

$\bar{r}$ is the averaged value of the scaling factor of the radius of the class container;

z' is a feature presentation of the first augmented version of the input sample x_i ;

z'' is a feature presentation of the second augmented version of the input sample x_i ;

$q_k^{\mu}(\cdot)$ is an assessment of the probability of belonging the feature representation of input image to the k -th class container;

τ is a temperature parameter that controls the dynamic range of the similarity function;

$q_k^{dist}(\cdot)$ is an assessment of the probability of belonging the feature representation of input image to the to k -th class;

S number of sections of multi-sectional classifier model;

e is a column matrix of ones, $e = [1, 1, ..., 1]^T$;

Hadamard is a square matrix whose entries are either +1 or −1 and whose rows are mutually orthogonal;

λ_{INF} is a coefficient for regulating the influence of the information criterion based component to the resulting loss;

λ_{CCL} is a coefficient for regulating the impact of contrastive-center loss to the resulting loss;

λ_C is a coefficient for regulating the impact of average distance between class prototypes and average radius of separate hypersurface class boundaries (container) to the resulting loss;

λ_{FSD} is a coefficient for regulating the impact of feature-level self-knowledge distillation to the resulting loss;

λ_{CSD} is a coefficient for regulating the impact of classifier-level self-knowledge distillation to the resulting loss;

λ_D is a coefficient for regulating the impact of discretization error of feature representation to the resulting loss;

λ_{UCE}^{out} is a coefficient for regulating the impact of consistency regularization based on unlabeled examples which hits out of class containers to the resulting loss;

λ_{UCE}^{in} is a coefficient for regulating the impact of consistency regularization based on unlabeled examples which hits into class containers to the resulting loss;

λ_{UL2} is a regularization coefficient for regulating the impact of Euclidean distance between feature representations from last layer and intermediate sections to the resulting loss.

INTRODUCTION

Image classification is one of the most widespread tasks in the field of artificial intelligence. Classification analysis of visual objects is often a component of safety-critical applications, such as autopilots of public transport and combat drones and medical diagnostics. It is used in production processes, monitoring traffic flows, inspection of infrastructure and industrial facilities and other similar tasks. Therefore, there is a need to ensure the resilience of artificial intelligence algorithms to destructive perturbations such. In the case of artificial intelligence for

DOI 10.15588/1607-3274-2022-3-9

image classification, specific perturbations such as adversarial attacks or noise, faults or fault injection attacks, as well as concept drift and out-of-distribution increase aleatoric and epistemic uncertainty and its involve a decrease in the productivity of the intellectual algorithm [1–3].

The resilience of the image classifier to perturbations is primarily ensured by achieving robustness for absorption of a certain level of destructive influences and implementing the graceful degradation mechanism to achieve the most effective behavior in conditions of incomplete certainty [1]. Data analysis models need to be continuously improved to take into account the non-stationary environment and new challenges. That is why the ability of the model to quickly recover performance by adapting to destructive effects and improve to increase the efficiency of subsequent adaptations are equally important components of resilience [2]. Recovery and improvement mechanisms are developed within the framework of the continual learning and meta-learning frameworks [4, 5].

Achieving a certain level of resilience is predicated upon the introduction of a certain resource and functional redundancy into the system, but in practice there are always resource constraints [6]. When designing and operating resilient systems taking into account resource constraints, the principles of rational resilience (affordable resilience) are often used. This involves achieving an effective balance between the system's lifecycle costs and the technical characteristics of the its resilience [7]. Researchers are trying to improve the resource efficiency of the inference by using biologically inspired cognitive mechanisms or adaptive computation based on cascade and multi-branch models [8, 9].

Separate components of resilience to certain types of destructive influences have been researched in many scientific papers, but the complex influence of multiple destructive factors at once had still not been considered [1–3]. In addition, machine learning algorithms for classification analysis of images that simultaneously implement such components of resilience as robustness, graceful degradation, recovery and improvement have not yet been proposed. Not all implementations of these components are compatible, especially under resource constraint conditions.

The object of research is the process of training and inference for image classifier that functioning under influences of destructive perturbations.

The subjects of the research are model architecture and training algorithm of image classifier that provide resilience to adversarial attacks, fault injection attacks and concept drift.

The research goal is development an effective model architecture and training algorithm of image classifier that provide resilience to adversarial attacks, fault injections and concept drift.

1 PROBLEM STATEMENT

Let $D_U = \{x_j^U \mid j = \overline{1,n}\}$ is set of unlabeled images and $D_S = \{x_{k,\,j}^S \mid k = \overline{1,K};\; j = \overline{1,n_k}\}$ is set of labeled images for classifier training and testing, where n is number of unlabeled examples and $\mathrm{n_k}$ is number of examples of k-th class. Is this case class index m can be composite form for implement hierarchical labeling for class hierarchy. Moreover, the structure of the vector of model parameters is known

$$g =< e_1,..,e_{\xi_1},..,e_{\Xi_1},\; f_1,..,f_{\xi_2},..,f_{\Xi_2} >, \quad (1)$$
$$\Xi_1 + \Xi_2 = \Xi\;,$$

In this case, the constraints $R_{\xi_1}(e_1,...,e_{\xi_1},...,e_{\Xi_1}) \le 0$, $R_{\xi_2}(f_1,...,f_{\xi_2},...,f_{\Xi_2}) \le 0$ are impose on parameters. These inequalities may include resource constraints, necessitating the development of resource-efficient algorithms.

It is necessary to find by machine learning an optimal values of parameters g (1) that provide tradeoff between maximum of class-wise averaged value of information-based efficiency criterion $\overline{J}$ and value of integrated metric R for resilience quantification on control time period T_c:

$$\overline{J} = \frac{1}{K}\sum_{k=1}^{K} J(\alpha_k, \beta_k, D_{1,k}, D_{2,k}) \quad (2)$$

$$R = \frac{\frac{1}{|P|}\sum_P \int_{t=0}^{T_c} \overline{J}(t)dt}{\int_{t=0}^{T_c} \overline{J}_0(t)dt}\;, \quad (3)$$

$$g^* = \arg\max_G \left\{\eta\overline{J}(g) + (1-\eta)R(g)\right\}. \quad (4)$$

2 REVIEW OF THE LITERATURE

The problem of image representation and image classification analysis remains an active research topic due to its relevance in safety-critical applications which require resilience to challenging operating conditions [2, 10]. Basic principles of system resilience to destructive perturbations are formulated in [6, 7]. These presuppose the existence of mechanisms of perturbation absorption, perturbation detection, graceful degradation, restoration of productivity and improvement. Research [1, 2, 3] studied vulnerability of artificial intelligence algorithms, identifying the following destructive effects: noise and adversarial attacks, faults and fault injection in the environment of intelligent algorithm deployment, concept drift and emergence of novelty, i.e., test examples that out of distribution of training data.

DOI 10.15588/1607-3274-2022-3-9

The ability to absorb destructive perturbations is called robustness. There are many methods and approaches to increase robustness to adversarial attacks. Some researchers separate methods for ensuring robustness to competitive attacks into the following categories : gradient masking methods, robustness optimization methods and methods of detecting adversarial examples [11]. Gradient masking includes some input data preprocessing methods (jpeg compression, random padding and resizing), thermometer encoding, adversarial logit pairing), defensive distillation, randomly choosing a model from a set of models or using dropout, and the use of generative models (ie, PixelDefend [12] and Defense-GAN [13]). However [14] demonstrated inefficiency of gradient masking methods. Robust optimization approach includes adversarial training, regularization methods which minimize the effects of small perturbations of the input (such as Jacobian regularization or L2-distance between feature representations for natural and perturbed samples), and provable defenses (ie, Reluplex algorithm [15]). Finally, yet another approach lies in developing an adversarial examples detector to reject such examples at the input of the main model. However, Carlini and Wagner [16], rigorously demonstrate that the properties of adversarial examples are difficult and resource-intensive to detect. In [11] it was proposed to divide the methods of protection against adversarial attacks into two groups, implementing two separate principles : methods of increasing intra-class compactness and inter-class separation of feature vectors and methods of marginalization or removal of non-robust image features. This work [17] emphasize the possibility for further development of these basic principles and their combination, taking into account other requirements and constraints.

There are three main approaches to ensure robustness to the injection of faults in the computing environment where neural networks are deployed : introduction of explicit redundancy [18], learning algorithm modification [19] and architecture optimization [19]. Faults are understood as accidental or intentional bit flips in memory which stores the weights or the original value of the neuron. The introduction of explicit redundancy is achieved, as a rule, by duplication of critical neurons and synapses, uniform distribution of synaptic weights and removal of unimportant weights and neurons. It is also possible to increase the robustness of the neural network to the injection of faults at the stage of machine learning by adding noise, perturbations or injecting direct faults during training. The same can also be achieved by including a regularization (penalty) term in the performance measure to indirectly incorporate faults in conventional algorithms [20]. Optimizing the architecture to increase robustness means minimizing the maximum error at the output of the neural network for a given number of inverted bits in memory where weights or results of intermediate calculations are stored. Authors of research [20] solved this problem with evolutionary search algorithms or Neural Architecture Search tools. However, architecture optimization is traditionally a very resource-intensive process.

Papers [21, 22] propose methods of domain randomization and adversarial domain augmentation which increase the robustness of the model under bounded data distribution shifts. Domain randomization is the generation of synthetic data with amount of variations large enough so that that real world data is viewed as simply another domain variation [21]. This can include randomization of view angles, textures, shapes, shaders, camera effects, scaling and many other parameters. Adversarial domain augmentation creates multiple augmented domains from the source domain by leveraging adversarial training with relaxed domain discrepancy constraint based on Wasserstein Auto-Encoder [22]. Transfer learning and multi-task learning also reinforce resistance to out-of-distribution perturbations. However, if there is a real concept drift in the data stream, there is a need to detect such a situation and implement reactive mechanisms to adapt [23]. There are studies on adaptation to real concept drift, but the lack of labels for test data or a significant delay in obtaining them remains a challenge.

Adversarial attacks, error injections, concept drift and out-of-distribution examples cannot always be absorbed, so the development of reactive resilience mechanisms, namely graceful degradation, recovery and improvement, remains relevant [2, 6]. The implementation of these mechanisms is often associated with the need to detect the perturbation. The most successful methods of detecting an adversarial and out-of-distribution samples and concept drift are based on the analysis of high-level feature space using a distance-based confidence score or prototype-based classifier [24, 25]. In [25], the mechanism for detecting faults affecting inference is based on the calculation of the reference value of the contrastive loss function on test diagnostic samples of data in the absence of faults. To detect faults, the current value of the contrast loss function for diagnostic data is compared with the reference value. In research [27] is proposed mechanisms of Nested Learning and Hierarchical Classification, particularly useful for the implementation of the mechanism of graceful degradation.

In [28], consider algorithms for adapting models to destructive perturbations, where the principles of active learning or contrastive learning are used to increase the speed of adaptation by reducing the requirement for labeled data in quantities. Semi-supervised learning methods are proposed in [29] for the simultaneous use of both labeled and unlabeled data in order to accelerate adaptation to concept drift. The methods of lifelong learning, which allow to continuously accumulate knowledge from different tasks and improve, as well as different reminder mechanisms helping avoid catastrophic forgetting problem are considered in [5]. Various approaches to the implementation of meta-learning to improve the effectiveness of adaptation are covered in [4]. The paper considers the principle of self-distillation for training neural networks which can implement adaptive

DOI 10.15588/1607-3274-2022-3-9

calculations and speed up the inference mode as the learning efficiency of the lower layers of the neural network grows.

Thus, there are numerous studies of separate principles of resilience of data classification models, but there are virtually no works which consider their coterminous combination. However, in systems analysis, there are studies related to the provision of affordable resilience [7] which are particularly relevant for data analysis systems operating under resource constraints.

3 MATERIALS AND METHODS

When building the model, we aim to implement the main characteristics of resilience: robustness, graceful degradation, recovery and improvement. The model is based on the following principles:

– hierarchical labeling and hierarchical classification to implement the principles of graceful degradation by coarsening the prediction with a more abstract class with reasonable confidence when classes at the bottom of the hierarchy are recognized with low confidence level;

– combining the mechanisms of self-knowledge distillation and nested learning to increase the robustness of the model by increasing the informativeness of the feedback for the lower layers at the training stage and accelerate inference by skipping high-level layers for simple samples at inference stage;

– prototype and compact spherical container formation for each class to simplify detection of out-of-distribution samples and concept drift;

– using memory FIFO-buffer with limited size to store labeled and unlabeled data with corresponding values of loss function obtained by inference for implementation diagnostic and recovery mechanism.

These principles should ensure resource-efficiency because the model will have small branches for intermediate decisions, which introduces minimal redundancy, since the main part of the feature extractor body is shared between intermediate classifiers. In addition, the size of the data buffers can be set to an acceptable capacity from the point of view of resource constraints.

Fig. 1 depicts the architecture of the resilient classifier with sectional design. Sections consist of ResBlocks of the well-known ResNet50 architecture. ResNet50 architecture also provided the inspiration for the Bottleneck module, serving to mitigate the impacts between each classifier of the lower sections, and to add distillation knowledge from the high-level feature map to the lower-level feature maps. The output of each section is used to construct a separate classifier. Each classifier receives feedback from the data labels and the last layer. Feedback from the last layer, denoted by a dotted line, ensures the implementation of the principle of self-knowledge distillation.

A set of prototype vectors is constructed for the classification analysis of the feature representation of each section output. Prototype vectors are not fixed, they are determined in the training process together with weights of feature extractor. To implement the graceful degradation principle, prototypes can belong to different levels in the hierarchy according to the hierarchy of labeling. In the example provided, a 2-level hierarchy is used. To increase immunity to noise and implementation of the information bottleneck, we approximate the feature representation to a discrete form, which is why the output of the feature extractor of each section uses the sigmoid layer and the corresponding regularization in the training algorithm.

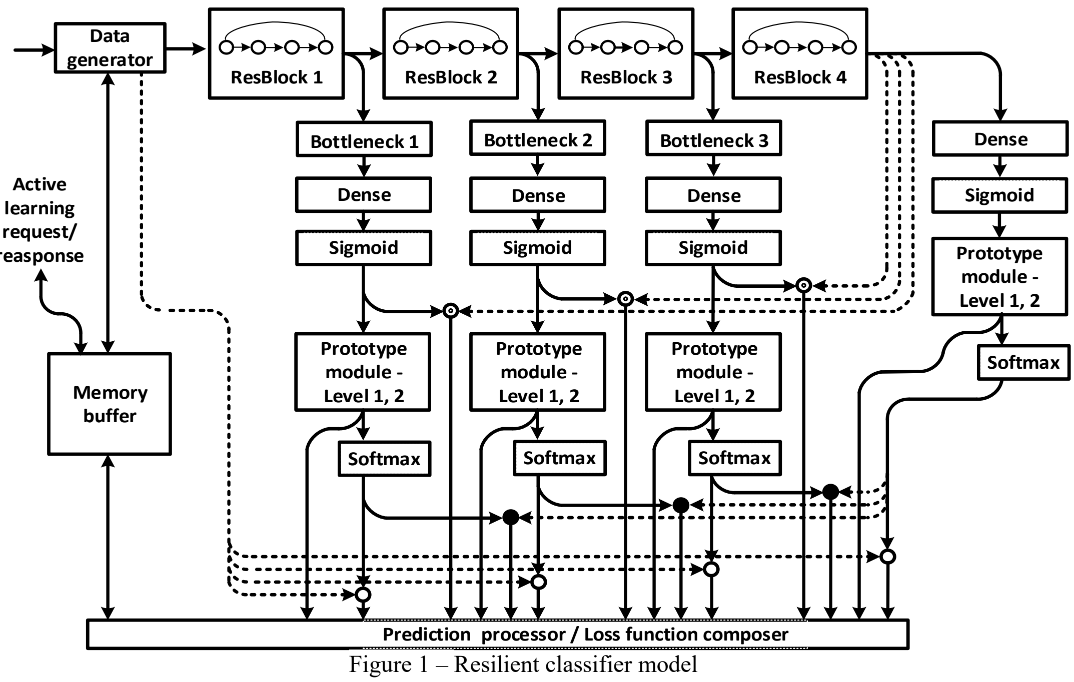


Figure 1 – Resilient classifier model

DOI 10.15588/1607-3274-2022-3-9

The radius of hyperspherical containers of classes is optimized for each prototypical classifier. Container radii are stored in memory to detect high levels of uncertainty when making decisions. Test samples outside the class containers become candidates for incremental learning using unlabeled samples and trigger a request for manual labeling (active learning) to be performed at a later stage. Controlling for the samples outside the class container can also be used for real concept drift and out-of-distribution detection.

After updating the weights and parameters of the model, the diagnostic dataset and the corresponding value of the loss function must be stored (or updated) in memory. After that, a subset of diagnostic data should be passed along for processing together with the test samples in each batch. This will allow comparison of the past and present values of the loss function to detect errors or injection faults in the memory of the neural network weights. Where the difference between past and present values of the loss function exceeds a certain threshold $\alpha = 0.01$ a neural network fine-tuning algorithm utilizing the diagnostic data needs to be initiated to bring this difference under a threshold $\beta = 0.001$.

Multi-section structure of the model with intermediate classifiers allows implementing adaptive calculations, allowing accelerating the recognition of simple images. At the same time, as the model is continually trained, it becomes faster due to increased recognition confidence of the lower section classifiers. This, in turn, will allow the rest of the high-level sections of the model to be skipped. The following rules for classification analysis in the adaptive calculations framework are proposed:

– Neural network calculations are performed sequentially, section by section;

– high-level sections can be skipped if in the output of the current section the maximal value of the membership function to a particular class of the lower hierarchical level exceeds the confidence threshold T;

– if the maximal value of membership function of any of the hierarchical levels of the classifier at the output of the current section has not increased compared to the previous section, then the subsequent calculations can be omitted;

– where any of the conditions of omission of the subsequent sections are fulfilled or the classifier in question is the last classifier in the model and the maximal value of the membership function of the lower hierarchical level does not exceed the confidence threshold, the higher level in the hierarchy is checked;

– where a class with a sufficient confidence level has not been identified, a decision is refused, a request for a manual labeling is generated, and the corresponding sample is designated as suitable for unsupervised tuning.

The confidence in the forecast of i-th sample belonging to the k-th recognition class, is determined by the following membership function

$$\mu_k(z_i) = 1 - \frac{dist(z_i, \overline{z}_k)}{N \cdot r_k}. \quad (5)$$

If maximum value of the function (1) for an input unlabeled sample z_i is less than zero, such a forecast should not be trusted and such sample should be added to the buffer of unlabeled data outside the training distribution. Where the input unlabeled sample falls into one of the containers of the recognition classes (at any of the levels), it should be added to the in-class unlabeled data buffer within the training distribution. Unlabeled sample buffers can be used for training with pseudo-labeling, soft-labeling or for consistency regularization.

Where the model was trained, but in the buffer of the new labeled data an occurrence of n samples of the c-th class misallocated during forward propagation to k-th class container is detected, the real concept drift is recognized.

To avoid catastrophic forgetting in the context of concept drift or emergence of a new recognition class a reminder function is implicitly implemented. Such function is based on unlabeled data buffers and prototypical vectors in feature space, which are changing slowly. Upper layers knowledge distillation mechanism also serves the same purpose.

Data from unlabeled data buffer can be moved to the labeled data queue after the feedback on their actual affiliation with the classes is received. The priority of specific samples being recommended for manual labeling depends on the value of the membership function (1).

During the development of the training algorithm, we aim to ensure the robustness, graceful degradation, recovery and improvement. To this end, the training algorithm will be based on the following principles :

– accounting for the hierarchy of data labeling and hierarchy class prototypes by calculating the loss function separately for each level of the hierarchy to provide graceful degradation at inference;

– implementation of self-knowledge distillation, i.e., distillation of knowledge from the high-level layer (section) of the model down to lower layers (sections) as additional regularization components to increase robustness and provide adaptive calculations in inference mode;

– increasing the compactness of the distribution of classes and the buffer zone between classes to increase resistance to noise, outliers, and adversarial attacks in turn as additional distance-based regularization component;

– penalization of discretization error (compression to binary form) of the feature representation as a way for implementing an information bottleneck to improve the robustness and informativeness of the feature representation;

– implementation of reactive mechanisms for rapid performance recovery under perturbations based on the fine-tuning weights on diagnostic data to eliminate the effects of detected faults, reset (re-initialization)

DOI 10.15588/1607-3274-2022-3-9

prototypes of drifting or new classes, use of new unlabeled data for consistency regularizing;

– ability to effectively use both labelled and unlabeled data samples to speed up adaptation with a limited quantity of labelled data, which usually comes with a significant lag;

– avoidance of catastrophic forgetting when adapting to perturbations without full retraining by implementing a reminding mechanism utilizing the data buffers, class prototypes and distillation feedback of the upper layers.

The proposed training method consists of two main stages :

– preparatory training the model on labeled and unlabeled data using a semi-supervised regime;

– adaptation to perturbation with semi-supervised supervision and active learning feedback.

The main criterion for learning in both cases is the information measure. The loss function based on the use of the information measure has the form:

$$L_{INF} = 1 - \overline{J} \ . \quad (6)$$

The normalized modification of C. Shannon's entropy-based information measure is used as the criterion of the recognition efficiency of the k -th class and calculated by the formula [30]

$$J_k = \frac{H_o - H_\gamma}{H_o} = $$
$$= 1 + \frac{1}{2}\left(\frac{\alpha_k}{\alpha_k + D_{2,k}} \log_2 \frac{\alpha_k}{\alpha_k + D_{2,k}} + \frac{D_{1,k}}{D_{1,k} + \beta_k} \log_2 \frac{D_{1,k}}{D_{1,k} + \beta_k} + \frac{\beta_k}{D_{1,k} + \beta_k} \log_2 \frac{\beta_k}{D_{1,k} + \beta_k} + \frac{D_{2,k}}{\alpha_k + D_{2,k}} \log_2 \frac{D_{2,k}}{\alpha_k + D_{2,k}} \right). \quad (7)$$

A separate hyperspherical surface is built for each class in the radial basis feature space. The accuracy characteristics of the hyperspherical decision boundary for each class can be calculated on the basis of statistical tests as follows :

$$D_{1,k} = \frac{TP_k}{TP_k + FN_k + \varepsilon} + \varepsilon , \quad (8)$$

$$D_{2,k} = \frac{TN_k}{TN_k + FP_k + \varepsilon} + \varepsilon , \quad (9)$$

$$\alpha_k = \frac{FN_k}{FN_k + TP_k + \varepsilon} + \varepsilon , \quad (10)$$

$$\beta_k = \frac{FP_k}{FP_k + TN_k + \varepsilon} + \varepsilon . \quad (11)$$

Procedures for calculating statistical tests are not differentiable, so in the training mode their smoothed versions can be used instead [31]

$$TP \approx \sum_{i=1}^{n_{MB}} \hat{y}_i \odot y_i , \quad (12)$$

$$FP \approx \sum_{i=1}^{n_{MB}} \hat{y}_i \odot (1 - y_i) , \quad (13)$$

$$FN \approx \sum_{i=1}^{n_{MB}} (1 - \hat{y}_i) \odot y_i , \quad (14)$$

$$TN \approx \sum_{i=1}^{n_{MB}} (1 - \hat{y}_i) \odot (1 - y_i) , \quad (15)$$

$$\hat{y}_i = \left\{ relu(\mu_{i,k}) \mid k = \overline{1,K} \right\} . \quad (16)$$

Admissible domain of criterion function (7) is bounded by inequalities $D_{1,k} \geq 0.5$ and $D_{2,k} \geq 0.5$, or $\beta_k < 0.5$ and $\alpha_k < 0.5$. In order to take into account the admissible domain of function (7) in the optimization procedure based on error backpropagation method it is proposed to perform the following operations when calculating the loss function [30]:

$$D_{1,k} = \max(D_{1,k}, 0.5) , \quad (17)$$

$$D_{2,k} = \max(D_{2,k}, 0.5) , \quad (18)$$

$$\alpha_k = \min(\alpha_k, 0.5) , \quad (19)$$

$$\beta_k = \min(\beta_k, 0.5) . \quad (20)$$

To increase the compactness of class distribution and inter-class gap in feature space it is proposed to use the contrastive-center loss function that calculated for labeled training samples [32]

$$L_{CCL} = \frac{dist(z_i, \overline{z}_{y_i})}{\sum_{k=1, k \neq y_i}^{K} dist(z_i, \overline{z}_k) + 1} . \quad (21)$$

To optimize boundaries of classes it is proposed to use additional regularization component L_C that connects the average distance between class prototypes and the average radius of separate hypersurface class boundaries (container)

$$L_C = \frac{\overline{r}}{\overline{d} + 1} , \quad (22)$$

DOI 10.15588/1607-3274-2022-3-9

$$\overline{d} = \frac{1}{N(K-1)^2} \sum_{k=1}^{K} \sum_{c=1}^{K} dist(\overline{z}_c, \overline{z}_k), \quad (23)$$

$$\overline{r} = \frac{1}{K} \sum_{k=1}^{K} r_k . \quad (24)$$

To speed up adaptation to changes, unlabeled data examples can be used in consistency regularization [29]. In this case, unlabeled data is divided into two groups : unlabeled examples that fall into the class containers; unlabeled examples that out of all class containers.

It is proposed to use unlabeled data that fall into the class containers in regularization component L_{UCE}^{in} which can be calculated by following formulas:

$$L_{UCE}^{in} = CE\left(q^{\mu}(z_i', \tau = 1), q^{\mu}(z_i'', \tau = 1)\right), \quad (25)$$

$$q_k^{\mu}(z_i, \tau) = \frac{\exp\left(\mu_k(z_i)/\tau\right)}{\sum_{c=1}^{K} \exp\left(\mu_c(z_i)/\tau\right)}. \quad (26)$$

Certain portions γ (<10%) of unlabeled data, which fall into class containers and have maximum values of $q\ (z_i)$, can be pseudo-labeled with the corresponding classes. Such pseudo-labeled data can be included in every mini-batches during training.

Unlabeled examples that out of all class containers may be examples of unknown classes or result of concept drift. In this case, soft-labeling $q_k^{dist}(z_i)$ based on distances to prototype of known classes should be used in consistency regularization component L_{UCE}^{out} :

$$L_{UCE}^{out} = CE\left(q^{\mu}(z_i), q^{dist}(z_i)\right), \quad (27)$$

$$q_k^{dist}(z_i) = \frac{\exp\left(-dist\left(z_i, \overline{z}_k\right)\right)}{\sum_{c=1}^{K} \exp\left(-dist\left(z_i, \overline{z}_c\right)\right)}. \quad (28)$$

Consistent regularization can be performed not only at the level of the classification module, but also at the level of features. The corresponding regularization component L_{UL2} of the loss function is calculated by the formula

$$L_{UL2} = dist\left(z_i', z_i''\right). \quad (29)$$

Kullback-Leibler divergence loss L_{CSD} and L_2 loss from hints L_{FSD} and calculated based on the S-th (last) output of the model and the s-th output (intermediate) of the model are used in additionally for self-knowledge distillation

$$L_{FSD} = dist(z_i^s, z_i^S), \quad (30)$$

$$L_{CSD} = KL\left(q^{\mu}(z_i^s, \tau), q^{\mu}(z_i^S, \tau)\right). \quad (31)$$

A regularization component which penalizes the discretization error of feature representation is introduced in addition to implement the information bottleneck [30]

$$L_D = z_i^T (e - z_i). \quad (32)$$

The initial values of the parameters of the lower level class prototypes are initialized on the basis of the Hadamard matrix using the principle of label smoothing. For this first the dimensionality of the Hadamard matrix is determined $N_{Hadamard} = 2^{ceil(\log_2(N))}$, where $ceil()$ is the function rounding a number to a larger integer value. All values less than 0 are replaced by 0, ie $Z = \max\left(0, Hadamard(N_{Hadamard})\right)$ subsequently. As the next step, to facilitate the process of adapting the class prototype to the data structure, the proposed approach uses label smoothing. This is performed according to the formula $Z' = Z * 0.7 + 0.15$, as a result of which the 1's will turn into 0.87, and the 0s into 0.15. K of the first vectors truncated by N first features , ie $\overline{z} = Z'[1:K, 1:N]$ are then selected from the resulting matrix. The trainable scale factor r_k for radius of hyperspherical decision boundary (container) of k-th class is initialized with a value of half of Plotkin's Bound, divided by the dimensionality of the feature space

$$r_k \leftarrow \left(\frac{1}{2} \frac{N}{2} \frac{K}{K-1}\right) \frac{1}{N} = \frac{K}{4 \cdot (K-1)}. \quad (33)$$

Appearance of a sample with a label indicating a new ($K+1$)-th lower-level class necessitates a formation of a new prototype for the class $\overline{z}_{K+1}$ with the corresponding initial values of the radius scale factor r_{K+1}. This is achieved by selecting the nearest vector from the remaining unused rows of a modified Hadamard matrix Z', where the proximity is determined on the basis of Euclidean Squared distance. Initial value of Radius scale factor for the new class is also determined by formula (14), but taking into account the new number of classes.

Each coordinate of the prototype of the upper hierarchical level is initialized by copying the corresponding coordinate of one of the prototypes of the lower level, selected at random. Initial class radius of the upper hierarchical level is determined by formula (14) taking into account the number of classes at this level.

Where a real concept drift is recognized, prototypes of drifting classes are populated with random numbers from the range [0; 1].

The resulting loss function is formed by the sum of the above components, averaged by sections of the model and

DOI 10.15588/1607-3274-2022-3-9

levels of class hierarchy, with coefficients that regulate the impact of individual components depending on the training regime.

The following combined loss function averaged over hierarchical levels and model sections is suggested for supervised learning

$$L_S = \lambda_{INF}\overline{L}_{INF} + \lambda_{CCL}\overline{L}_{CCL} + \lambda_C\overline{L}_C + \lambda_{FSD}\overline{L}_{FSD} + \\ +\lambda_{CSD}\overline{L}_{CSD} + \lambda_D L_D . \quad (34)$$

When new labeled data appear, they are combined with unlabeled data from FIFO-buffer to implement continuous adaptation using the loss function

$$L_{TOTAL} = L_S + \lambda_{UCE}^{out}\overline{L}_{UCE}^{out} + \lambda_{UCE}^{in}\overline{L}_{UCE}^{in} + \lambda_{UL2}\overline{L}_{UL2} . \quad (35)$$

Default values of coefficients are proposed as follows: $\lambda_{INF} = 1.0$, $\lambda_{CCL} = 1.0$, $\lambda_C = 0.001$, $\lambda_{FSD} = 0.01$, $\lambda_{CSD} = 0.1$, $\lambda_D = 0.001$, $\lambda_{UCE}^{out} = 0.1$, $\lambda_{UCE}^{in} = 0.1$, $\lambda_{UL2} = 0.01$.

4 EXPERIMENTS

The Cifar10 dataset was chosen for experimental research because it is publicly available and its images are small in size, which speeds up experimental research. The classes of this dataset can be arranged in a hierarchical structure. For example, the first upper level class will be the animal class, which includes the subclasses bird, cat, deer, dog, frog and horse. The second upper level class will be the vehicle class, which includes airplane, automobile, ship and truck subclasses. Therefore, 12 prototype vectors will be used at the output of the classifier of each section, of which 2 for upper level prototypes and 10 lower level prototypes. For all experiments, the chosen confidence threshold, considered sufficient to make a decision, is $T = 0.8$. The Cifar10 dataset consists of 50,000 training images and 10,000 test 32x32 color images distributed evenly between 10 classes. For convenience of the analysis for training of base model we will use 70% of training data to form dataset_base, and use the remaining 30% for additional dataset_additional training dataset.

As a result of perturbations, there is a notable decrease in model performance. To test the ability to recover, we define recovery as the state of reaching 95% of the performance level observed prior to perturbation. The control interval is set at $T_C = 200$ to ensure testing on the full volume of test data. During recovery, each test mini-batch is preceded by a training mini-batch. The size of the mini-batch is equal to 128 examples.

To test the model for resistance to faults and the ability to recover, it is suggested to use the TensorFI2 library, which is capable of simulating software and hardware faults. In the experiment, it is proposed to consider the influence of the most difficult to absorb type of faults by generation of random bit inversion (bit-flip injection) in each layer of the model. A fixed share of tensors is randomly selected (fault rate) and 1 bit is randomly selected from them to be inverted. For diagnostics and recovery, along with test data, diagnostic data is added to the input of the model in each mini-batch. Diagnostic data are generated from the dataset_additional set and data quantity is equal to the size of 128 examples.

Different model weights have different importance and impact on model performance. In addition, a fault in the higher bits of tensor value leads to a greater distortion of the results than a fault in the lower bits. Therefore, statistical characteristics should be used to evaluate and compare the model's resilience to different proportions of damaged tensors. The statistical characteristics are derived from a large number of experiments, where bits and tensors for inversion are chosen randomly from a uniform distribution. For simplicity, we can consider the median value (MED) and interquartile value (IRQ) of the integral metric of classifier's resilience for the classes of the upper and lower hierarchical level, calculated after 1000 experiments. We can also consider the influence of the dimensionality of the feature space.

To test the model for resistance to noise and adversarial attacks, it is suggested not to rely on gradients or other features of the model architecture and learning algorithm. Instead testing will be carried out on the basis of black box attacks. To assess the level of disturbances, the resistance to which is tested, it is necessary to choose a metric. In practice, such metrics as L0-norm, L1-norm, L2-norm and L∞-norm have become widespread. However, only L0-norm and L∞-norm impose restrictions on the spatial distribution of noise, which prevents the formation of distorted samples that are incorrectly classified even by humans. In addition, the selection of the perturbation level by the metric L0-norm or L∞-norm does not depend on the size of the image, which is convenient for comparison. Covariance matrix adaptation evolution strategy (CMA-ES) using the $L\infty$ metric [33] is chosen as an evolutionary attack strategy for our experiments. Classifier efficiency measurements are performed on perturbed test samples, with each mini-batch of perturbed test data created on the basis on the actual model. At the same time, mini-batches of perturbed data from the dataset_additional set are created, and 50% of them are provided with data labels for active learning emulation. Perturbed data from the dataset_additional set is not involved in measuring the model's efficiency, but is used to adapt it to disturbances of this type.

Resilience testing to the appearance of new classes and to the concept drift is performed on the classes of lower hierarchical level. Each of the classes will be considered as a new class in turn. Likewise, real concept drift will be examined between any pair of classes.

5 RESULTS

Fig. 2 shows an example of model performance recovery curves for classes of the lower hierarchical level with the feature space dimension N=64 after fault injection. The vertical axis corresponds to the value of the

DOI 10.15588/1607-3274-2022-3-9

information criterion averaged over the set of the classes, and the horizontal axis corresponds to the number of test iterations of the trained model on the dataset_base set. The first 50 iterations take place without fault injection, and on the 51st iteration, 4 versions of the model are generated with a different proportion of tensors with an inverted bit in a random position, i.e. $fault\_rate \in \{0,1; 0,3; 0,5; 0,6\}$. Therefore only 4 recovery curves of the model's performance are presented below.

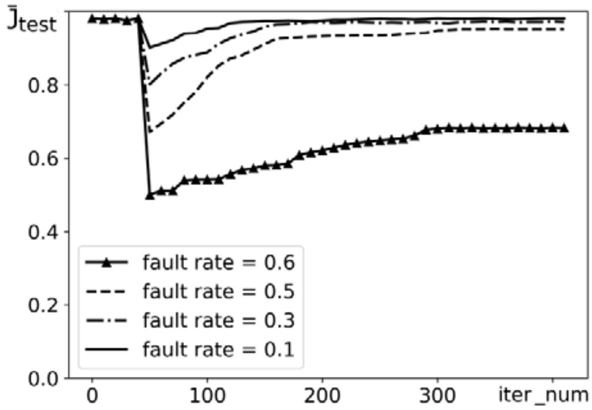


Figure 2 – Example of performance recovery curves after fault injection computed for low-level classes with information measure as performance metric

Table 1 below shows the experimental data after testing the resilience of the model to the faults injection, where $\bar{J}_0$ is the average value of the information criterion before the impact of the fault injection, averaged over the set of the classes, N is the selected dimension of the features. In this case, the table shows the data collected for different hierarchical levels of the model. The hierarchical level number is denoted by the symbol H .

Table 1 – Experimental data of model resilience to the faults injection testing

H	N	fault rate	MED(R)	IRQ(R)	$\bar{J}_0$
1	64	0.1	0.981	0.021	0.992
1	64	0.3	0.952	0.019	0.992
1	64	0.5	0.883	0.020	0.992
1	64	0.6	–	–	0.992
2	64	0.1	0.978	0.022	0.978
2	64	0.3	0.945	0.021	0.978
2	64	0.5	0.873	0.038	0.978
2	64	0.6	–	–	0.978
1	128	0.1	0.981	0.019	0.985
1	128	0.3	0.955	0.018	0.985
1	128	0.5	0.919	0.020	0.985
1	128	0.6	–	–	0.985
2	128	0.1	0.979	0.021	0.971
2	128	0.3	0.951	0.022	0.971
2	128	0.5	0.880	0.019	0.971
2	128	0.6	–	–	0.971

Analysis of the table 1 shows that if the share of damaged tensors reaches 60%, it becomes impossible to ensure recovery during processing T_C mini-batches. Fig. 2 shows the performance recovery curves, where the curve corresponding to the damage of 60% of the tensors after 200 iterations does not improve and does not show a recovery of 95% of the performance prior to perturbance. In addition, the analysis of the table 1 shows that increasing the dimensionality of the feature space leads to both a slight decrease in the performance of the model without disturbances, and a slight improvement in the median value of the integral metric of resilience. The corresponding interquartile value of the integral metric of resilience is in the interval [0.01; 0.04].

Fig. 3 shows an example of recovery curves of model performance for classes of the lower hierarchical level with the feature space dimension N=64 after the application of adversarial attacks. The vertical axis corresponds to the value of the information criterion averaged over the set of the classes, and the horizontal axis corresponds to the number of iterations of testing the trained model on the dataset_base set. The first 50 iterations are tested without adversarial attacks, and on the 51st iteration, data sets with 4 different threshold values of the disturbance level are generated, i.e. $threshold \in \{1; 3; 5; 10\}$. Therefore, 4 performance recovery curves are displayed.

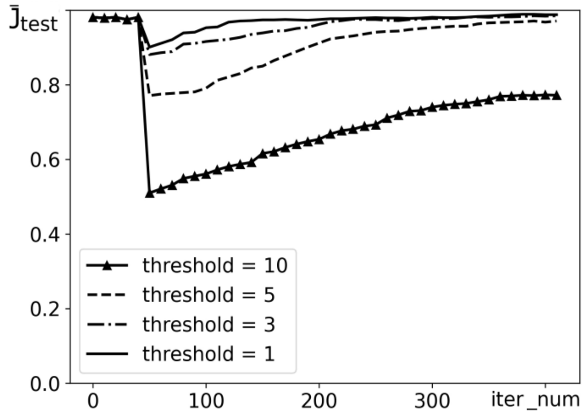


Figure 3 – Example of performance recovery curves after adversarial attack computed for low-level classes with information measure as performance metric

Table 2 shows the result of the experimental testing the model's resilience to adversarial L∞ -attacks.

Analysis of the table 2 shows that if the adversarial perturbation level is less than 10, it becomes impossible to obtain recovery by processing T_C mini-batches. Fig. 3 shows performance recovery curves, where the curve corresponding to a perturbation level of 10 after 200 iterations does not provide 95% performance recovery. In addition, the analysis of the table 2 shows that an increase in the dimensionality of the feature space leads to a slight decrease in the efficiency of the model on unperturbed data, but also to a noticeable improvement in the median value of the integral index of resilience, with corresponding interquartile value of resilience being in the interval [0.01; 0.03]. Therefore, according to formula

DOI 10.15588/1607-3274-2022-3-9

(4), the dimension of space $N = 128$ is a more optimal compromise option than lower dimension $N = 64$.

Table 2 – Experimental data of the model resilience to adversarial attacks testing

H	N	threshold	MED(R)	IRQ(R)	$\overline{J}_0$
1	64	1	0.980	0.017	0.992
1	64	3	0.955	0.019	0.992
1	64	5	0.885	0.017	0.992
1	64	10	0.667	0.027	0.992
2	64	1	0.978	0.028	0.978
2	64	3	0.954	0.021	0.978
2	64	5	0.879	0.017	0.978
2	64	10	–	–	0.978
1	128	1	0.988	0.018	0.985
1	128	3	0.967	0.018	0.985
1	128	5	0.925	0.022	0.985
1	128	10	0.701	–	0.985
2	128	1	0.983	0.021	0.971
2	128	3	0.962	0.021	0.971
2	128	5	0.905	0.026	0.971
2	128	10	–	–	0.971

A comparison of the averaged information efficiency criterion and the integral metric of resilience for different hierarchical levels shows that the upper-level classifier is characterized by a lower level of uncertainty and exhibits a higher level of resilience to disturbances, which allows it to be used in graceful degradation mechanisms in case of adversarial attacks.

Fig. 4 shows the performance recovery curve for the worst-case variant of the new class and the worst-case pair of drifting classes in terms of the model's resilience to these perturbations.

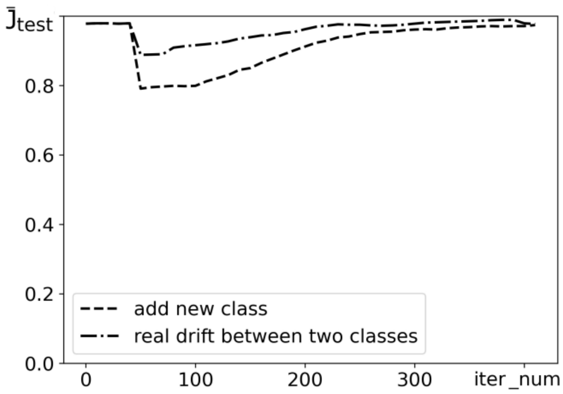


Figure 4 – Worst cases of performance recovery curves after add new class and real concept drift between pair of classes

Analysis of Fig. 4 shows that in both cases the T_C quantity of mini-batches (iterations) was sufficient for recovery. In comparison, learning from scratch required more than 100 times more mini-batches (taking into account 10 learning epochs and a mini-batch size of 128 samples). The worst performing new class from the point of view of the integral metric of resilience was the "bird" class (R=0.88). The worst pair of drifting classes from the point of view of the integral metric of resilience were "truck" and "automobile" classes with corresponding R=0.95.

Thus, the ability of the proposed algorithm to restore performance after exposure to perturbations has been experimentally proven. Described method of adaptation to adversarial attacks ensures absorption of disturbances of this type and amplitude and ensures performance recovery. Superior efficiency and resilience of the algorithm during the analysis of classes of a higher hierarchical level was also confirmed; this forms the basis for implementation of graceful degradation mechanisms.

6 DISCUSSION

The proposed model of the classifier has a multi-section structure designed to implement adaptive calculations and increase the generalization capabilities of the model due to self-knowledge distillation. Integral metric of model resilience using the outputs of each section and the model using the output of only the last layer of the model are compared to identify the influence of the multi-section structure on the resilience of the model. The model with the feature space dimension N=64 is considered.

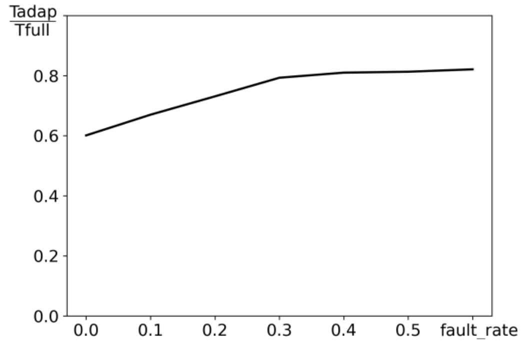
Table 3 – Comparison of the integral metric of resilience for the model using the outputs of individual sections and the model with a single output in the last layer

Only single output	Perturbation	MED(R)	IRQ(R)
True	Fault injection (*fault_rate*=0.3)	0.891	0.034
True	Adversarial attack (*threshold*=3)	0.912	0.053
False	Fault injection (*fault_rate*=0.3)	0.955	0.018
False	Adversarial attack (*threshold*=3)	0.965	0.021

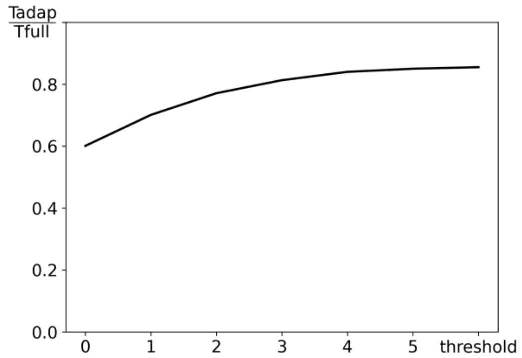
Analysis of the table 3 shows that the median value of the integral metric of resilience for the model using the outputs in all sections is 5–6% higher compared to the model with a single output on the last layer.

It is assumed that as the multi-sectional model architecture is trained, its computational efficiency of inference is improved by saving resources on simple examples without perturbations. Fig. 5 shows the dependence of the ratio of the average time spent in the adaptive mode T_{adap} to the time of inference across the entire network T_{full} on the *fault_rate* (Fig. 5a) and maximum amplitude of the adversarial L_{∞} -attack (Fig. 5b).

DOI 10.15588/1607-3274-2022-3-9



a



b

Figure 5 – Dependence of the average time ratio in the adaptive mode to the time of inference across the entire network on the factor of influence : a – *fault_rate*; b – maximum amplitude of the adversarial L_∞ -attack

Analysis of Fig. 5 confirms the hypotheses that the average inference time increases when the amplitude of the adversarial attack and the frequency of faults increase and vice versa. This can also be considered a mechanism of graceful degradation.

CONCLUSIONS

The **scientific novelty** of obtained result are the new model architecture and the learning algorithm of a multilayer classifier with the property of resilience to the injection of faults, adversarial attacks, and concept drift.

The model with the proposed architecture has a multi-section structure. At the output of each section, a hierarchy of optimized prototypes and radii of hyperspherical separation boundaries (containers) of classes is built, which ensures the absorption of some part of disturbances and the graceful degradation.

A new learning algorithm that combines ideas and principles of self-knowledge distillation, maximization of compactness of class distribution and interclass buffer zone, discretization of feature representation and consistency regularization is proposed. Self-knowledge distillation is aimed at improving the efficiency of an inference by adaptive computing and the mechanism of graceful degradation. Consistency regularization is carried out both at the level of classification output and at the level of features and is used to increase the robustness and speed of adaptation to destructive perturbations due to the effective use of unlabeled data. At the same time, the main component of the loss function is the information criterion of the classifier's effectiveness, expressed as a functional of smoothed probability estimates for errors of the first and second kind, true positives and true negatives tests.

During testing of the proposed algorithm on the Cifar10 dataset, it was found that if the proportion of damaged tensors reaches 60%, it is not possible to ensure recovery during the processing of mini-batches both for the upper and lower levels of class hierarchy. Similarly, if the adversarial L_∞ -attack perturbation level is 10, it fails to recover during mini-batches processing at the lower class hierarchy level, but for the upper class hierarchy level it is able to achieve 95% recovery of the performance obtained on unperturbed samples. In addition, it was observed that increasing the dimensionality of the feature space leads to a noticeable improvement in the median value of the integral mectric of resilience. At the same time, the interquartile value of the integral metric of resilience is in the interval [0.01; 0.03].

A comparison of the averaged information efficiency criterion and the integral metric of resilience for different class hierarchy levels shows that the upper level of class hierarchy is characterized by a lower level of uncertainty and exhibits a higher level of resilience to disturbances, which allows it to be used in graceful degradation mechanisms under the influence of adversarial attacks.

The median value of the integral metric of resilience of model that uses the outputs of all sections is 5–6% higher compared to the model that has a single output on the last layer. The multi-section structure of the model saves 40% of time on the test dataset, but in the case of perturbation influences, the processing slows down a bit.

The proposed learning algorithms provide adaptation to the appearance of a new class and a real concept drift between a pair of classes in T_C =200 iterations with a mini-batch size of 128 examples. The worst class in the Cifar10 dataset, from the point of view of the integral metric of resilience, if we consider it as a new class, is the "bird" class, for which the value R=0.88 was reached. The worst pair of drifting classes from the point of view of the integral metric of resilience are the "truck" and "automobile" classes, for which the value of R=0.95 was reached.

The practical significance of the achieved outcomes is formation of a new methodological basis for the development of classification analysis algorithms with resiliece to adversarial attacks, fault injection and concept drift.

The **prospects for further research** are the development of criteria, models, and methods for measuring and certifying the resilience of image classification analysis models.

ACKNOWLEDGMENTS

The research was concluded in the Intellectual Systems Laboratory of Computer Science Department at

DOI 10.15588/1607-3274-2022-3-9

Sumy State University with the financial support of the Ministry of Education and Science of Ukraine in the framework of state budget scientific and research work of DR No. 0122U000782 "Information technology for providing resilience of artificial intelligence systems to protect cyber-physical systems".

Contribution of authors : development of conceptual provisions and methodology of research, development of mathematical model and training algorithm, analysis of research results – V. V. Moskalenko; software development and conducting experiments for testing resilience to faults injection – A. S. Moskalenko; software development and conducting experiments for testing resilience to adversarial attacks – A. G. Korobov; software development and conducting experiments for testing resilience to real concept drift.

REFERENCES

1. Eigner O., Eresheim S., Kieseberg P., Klausner L., Pirker M., Priebe T., Tjoa S., Marulli F., Mercaldo F. Towards Resilient Artificial Intelligence: Survey and Research Issues, *2021 IEEE International Conference on Cyber Security and Resilience (CSR), Virtual conference, 26–28 July,* 2021, pp. 536–542. DOI: 10.1109/CSR51186.2021.9527986.
2. Olowononi F. O., Rawat D. B., Liu C. Resilient Machine Learning for Networked Cyber Physical Systems: A Survey for Machine Learning Security to Securing Machine Learning for CPS, *IEEE Communications Surveys Tutorials,* 2021, Vol. 23, No. 1, pp. 524–552. DOI: 10.1109/COMST.2020.3036778.
3. Dymond J. Graceful Degradation and Related Fields, *A review for Applied Research Centre at the Alan Turing Institute*, 2021, pp. 1–32. DOI: 10.48550/arXiv.2106.11119.
4. Hospedales T., Antoniou A., Micaelli P., Storkey A. Meta-Learning in Neural Networks: A Survey, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 2021, 20 p. DOI: 10.1109/TPAMI.2021.3079209.
5. Parisi G., Kemker R., Part J., Kanan C. , Wermter S. Continual lifelong learning with neural networks: A review, *Neural Networks*, 2019, No. 113, P. 54–71. DOI: 10.1016/j.neunet.2019.01.012
6. Fraccascia L., Giannoccaro I., Albino V. Resilience of Complex Systems: State of the Art and Directions for Future Research, *Complexity*, 2018, pp. 1–44. DOI: 10.1155/2018/3421529.
7. Madni A. Affordable Resilience, *Transdisciplinary Systems Engineering*, 2017, pp. 133–159. DOI: 10.1007/978-3-319-62184-5_9.
8. Zhang L., Bao C., Ma K. Self-Distillation: Towards Efficient and Compact Neural Networks, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021, Vol. 44 (8), pp. 4388–4403. DOI: 10.1109/TPAMI.2021.3067100.
9. Marquez E., Hare J., Niranjan M. Deep Cascade Learning, *IEEE Transactions on Neural Networks and Learning Systems,* 2018, Vol. 29(11), pp. 5475–5485. DOI : 10.1109/TNNLS.2018.2805098.
10. Makarichev V., Lukin V., Illiashenko O., Kharchenko V. Digital Image Representation by Atomic Functions: The Compression and Protection of Data for Edge Computing in IoT Systems, *Sensors*, 2022, Vol. 22(10), P. 3751. DOI : 10.3390/s22103751.
11. Smith L. N. A useful taxonomy for adversarial robustness of Neural Networks, *Trends in Computer Science and Information Technology,* 2020, pp. 037–041. DOI: 10.48550/arXiv.1910.10679.
12. Song Y.. Kim T., Nowozin S., Ermon S., Kushman N. PixelDefend: Leveraging Generative Models to Understand and Defend against Advers arial Examples, *Sixth International Conference on Learning Representations, Vancouver CANADA, 30 Apr. –3 May*, 2018, 20 p. DOI: 10.48550/arXiv.1710.10766.
13. Samangouei P., Kabkab M., Chellappa R. Defense-GAN: Protecting Classifiers Against Adversarial Attacks Using Generative Models, *Sixth International Conference on Learning Representations (ICLR 2018), Vancouver CANADA, 30 Apr – 3 May,* 2018, 17 p. DOI: 10.48550/arXiv.1805.06605.
14. Athalye A., Carlini N., Wagner D., Obfuscated Gradients Give a False Sense of Security: Circumventing Defenses to Adversarial Examples. [online] arXiv.org, 2022. Access mode: https://arxiv.org/abs/1802.00420 [Accessed 1 June 2022]. DOI: 10.48550/arXiv.1802.00420.
15. Xu J., Li Z., Du B., Zhang M., Liu J. Reluplex made more practical: Leaky ReLU [Text], *IEEE Symposium on Computers and Communications (ISCC), Rennes, France, July 7–July 10 2020,* IEEE, 2022, 7 p. DOI: 10.1109/ISCC50000.2020. 9219587.
16. Carlini N., Wagner D. Adversarial Examples Are Not Easily Detected: Bypassing Ten Detection Methods, *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security, Dallas Texas USA, 3 Nov. 2017*. NY, United States, 2017, pp. 3–14. DOI: 10.1145/3128572.3140444.
17. Silva S., Najafirad P. Opportunities and Challenges in Deep Learning Adversarial Robustness: A Survey, *IEEE Transactions on Knowledge and Data Engineering*, 2020, 20 p. DOI: 10.48550/arXiv.2007.00753.
18. Huang K., Siegel P. H., Jiang A. Functional Error Correction for Robust Neural Networks, *IEEE Journal on Selected Areas in Information Theory,* 2020, 24 p. DOI: 10.48550/arXiv.2001.03814.
19. Hacene G. B., Leduc-Primeau F., Soussia A. B., Gripon V., Gagnon F. Training modern deep neural networks for memory-fault robustness, *IEEE International Symposium on Circuits and Systems (ISCAS 2019), Sapporo, Hokkaido*. Japan, 26–29 May 2019, 5 p. DOI: 10.1109/ISCAS.2019.8702382.
20. Li W., Ning X., Ge G., Chen X., Wang Y. , Yang H. FTT-NAS: Discovering Fault-Tolerant Neural Architecture, *Proceeding of 25th Asia and South Pacific Design Automation Conference (ASP-DAC)*. Beijing, China, 13–16 Jan. 2020, IEEE Press, pp. 211–216. DOI: 10.1109/ASP-DAC47756.2020.9045324.
21. Valtchev S., Wu J. Domain randomization for neural network classification, *Journal of Big Data*, 2021, Vol. 8, Article No. 94, 12 p. DOI: 10.1186/s40537-021-00455-5.
22. Qiao F., Zhao L., Peng X. Learning to Learn Single Domain Generalization, *Computer Vision and Pattern Recognition,* 2020, pp. 1–13. DOI : 10.48550/arXiv.2003.13216.
23. Priya S., Uthra R. Deep learning framework for handling concept drift and class imbalanced complex decision-making on streaming data, *Complex & Intelligent Systems*, 2021, 17 p. DOI: 10.1007/s40747-021-00456-0.
24. Jiang H., Kim B., Guan M. Y., Gupta M. R. To Trust Or Not To Trust A Classifier, *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, 2018, pp. 5546–5557. DOI: 10.48550/arXiv.1805.11783.

DOI 10.15588/1607-3274-2022-3-9

25. Shu Y., Shi Y. , Wang Y. , Huang T., Tian Y. P-ODN: Prototype-based Open Deep Network for Open Set Recognition, *Scientific Reports*, 2020, No. 10, Article No. 7146. DOI: 10.1038/s41598-020-63649-6.
26. Wang C., Zhao P., Wang S., Lin X. Detection and recovery against deep neural network fault injection attacks based on contrastive learning, *3rd Workshop on Adversarial Learning Methods for Machine Learning and Data Mining at KDD,* Virtual Event, USA ,14 Aug 2021, 5 p.
27. Achddou R., Di Martino J., Sapiro G. Nested Learning for Multi-Level Classification, *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* Toronto, Canada, 6–11 June 2021, pp. 2815–2819. DOI: 10.1109/ICASSP39728.2021.9415076.
28. Margatina K., Vernikos G., Barrault L., Aletras N. Active Learning by Acquiring Contrastive Examples, *Conference on Empirical Methods in Natural Language Processing, Online and Punta Cana, Dominican Republic, Nov. 2021*, pp. 650–663. DOI: 10.48550/arXiv.2109.03764.
29. Park J., Yun S., Jeong J., Shin J. OpenCoS: Contrastive Semi-supervised Learning for Handling Open-set Unlabeled Data, *International Conference on Learning Representations ICLR Virtual, 3–7 May 2022*, 14 p. DOI: 10.48550/arXiv.2107.08943.
30. Konkle T., Alvarez G. A self-supervised domain-general learning framework for human ventral stream representation, *Nature Communications*, 2022, Vol. 13, Article No. 491, 12 p. DOI: 10.1038/s41467-022-28091-4.
31. Moskalenko V., Zaretskyi M., Moskalenko A., Korobov A., Kovalsky Y. Multi-stage deep learning method with self-supervised pretraining for sewer pipe defects classification, *Radioelectronic and computer systems*, 2021, No. 4, pp. 71–81. DOI: 10.32620/reks.2021.4.06.
32. Li G., Pattabiraman K., DeBardeleben N. TensorFI: A Configurable Fault Injector for TensorFlow Applications, *2018 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW).* Memphis, TN, USA, 15–18 Oct. 2018, pp. 313–320. DOI: 10.1109/ISSREW.2018.00024.
33. Kotyan S., Vargas D. Adversarial robustness assessment: Why in evaluation both L0 and L∞ attacks are necessary [Text], *PLOS ONE*, 2022, No. 17(4), Article No. e0265723, 22 p. DOI: 10.1371/journal.pone.0265723.

Received 15.07.2022.
Accepted 21.08.2022.

УДК 004.891.032.26:629.7.01.066

КЛАСИФІКАТОР ЗОБРАЖЕНЬ ІЗ РЕЗІЛЬЄНТНІСТЮ ДО ПРОТИБОРЧИХ АТАК, ІНЖЕКЦІЇ НЕСПРАВНОСТЕЙ ТА ДРЕЙФУ КОНЦЕПЦІЙ – АРХІТЕКТУРА МОДЕЛІ ТА АЛГОРИТМ НАВЧАННЯ

Москаленко В. В. – канд. техн. наук, доцент, доцент кафедри комп’ютерних наук, Сумський державний університет, Суми, Україна.

Москаленко А. С. – канд. техн. наук, старший викладач кафедри комп’ютерних наук, Сумський державний університет, Суми, Україна.

Коробов А. Г. – канд. техн. наук, старший викладач кафедри комп’ютерних наук, Сумський державний університет, Суми, Україна.

Зарецький М. О. – молодший науковий співробітник Лабораторії інтелектуальних систем, Сумський державний університет, Суми, Україна.

АНОТАЦІЯ

Актуальність. Проблема вразливості алгоритмів класифікаційного аналізу зображень до деструктивних збурень досі не була повністю вирішена і є досить актуальною для критичних до безпеки застосувань. Тому об’єктом дослідження є процес навчання та формування рішень для класифікатора зображень, що функціонує під впливом деструктивних збурень. Предметом дослідження є архітектура моделі та алгоритм навчання класифікатора зображень, що забезпечують стійкість до протиборчих атак, інжекції несправностей і дрейфу концепцій.

Мета дослідження – є розроблення ефективних архітектури моделі та алгоритму навчання, які забезпечують стійкість до протиборчих атак, інжекції несправностей та дрейфу концепцій.

Методи дослідження. Архітектура моделі та алгоритм навчання реалізовані шляхом поєднання ідей і принципів самодистиляції знань, максимізації інформаційної міри та компактності розподілу класів, максимізації міжкласового зазору, стиснення даних на основі дискретизації ознакового подання, а також навчання з частковим залученням учителя на основі регуляризації узгодженості.

Результати. Розроблено архітектуру моделі і алгоритм навчання класифікатора зображень. Отриманий класифікатор було випробувано на наборі даних Cifar10 для оцінювання його резільєнтності на інтервалі в 200 міні-пакетів із розміром навчального і тестового міні-пакету в 128 зразків для таких збурень : протиборчі L∞-атаки чорної шухляди з рівнями 1, 3, 5 та 10; інверсія одного випадково обраного біту в тензорі для 10%, 30%, 50% та 60% випадково обраних тензорів; додавання одного нового класу; реальний дрейф концепцій між парою класів. Розглянуто вплив розмірності простору ознак на значення інформаційного критерію ефективності моделі без збурень та на значення інтегрального показника резільєнтності під час впливу збурень.

Висновки. Запропоновані архітектура моделі і алгоритм навчання забезпечують поглинання частини збурюючого впливу, витончену деградацію за рахунок ієрархічності класів та адаптивних обчислень, а також швидку адаптацію на обмеженій кількості розмічених даних. Показано, що адаптивні обчислення дозволяють економити до 40% ресурсів за рахунок раннього прийняття рішень на нижніх секціях моделі, однак збурюючий вплив призводить до уповільнення, що можна розглядати як витончену деградацію. Доведено, що багатосекційна структура, що навчається з використанням принципів дистиляції само-знань, забезпечує більш ніж на 5% покращення значення інтегрального показника резільєнтності порівняно з архітектурою, де рішення приймається на останньому шарі моделі. Помічено, що розмірність простору ознак помітно впливає на стійкість до протиборчих атак і може обиратися як компроміс між резільєнтністю до збурень та ефективність без впливу збурень.

DOI 10.15588/1607-3274-2022-3-9

КЛЮЧОВІ СЛОВА: класифікація зображень, робастність, резільєнтність, витончена деградація, протиборчі атаки, інжекція несправностей, дрейф концепцій.

УДК 004.891.032.26:629.7.01.066

КЛАССИФИКАТОР ИЗОБРАЖЕНИЙ С РЕЗИЛЬЕНТНОСТЬЮ К СОСТЯЗАТЕЛЬНЫМ АТАКАМ, ИНЖЕКЦИИ НЕИСПРАВНОСТЕЙ И ДРЕЙФУ КОНЦЕПЦИЙ – АРХИТЕКТУРА МОДЕЛИ И АЛГОРИТМ ОБУЧЕНИЯ

Москаленко В. В. – канд. техн. наук, доцент, доцент кафедры компьютерных наук, Сумской государственный университет, Сумы, Украина.

Москаленко А. С. – канд. техн. наук, старший преподаватель кафедры компьютерных наук, Сумской государственный университет, Сумы, Украина.

Коробов А. Г. – канд. техн. наук, старший преподаватель кафедры компьютерных наук, Сумской государственный университет, Сумы, Украина.

Зарецький М. О. – младший научный сотрудник Лаборатории интеллектуальных систем, Сумской государственный университет, Сумы, Украина.

АННОТАЦИЯ

Актуальность. Проблема уязвимости алгоритмов классификационного анализа изображений к деструктивным возмущениям до сих пор не была полностью решена и достаточно актуальна для критических к безопасности применений. Поэтому объектом исследования является процесс обучения и формирования решений классификатора изображений, функционирующем под влиянием деструктивных возмущений. Предметом исследования является архитектура модели и алгоритм обучения классификатора изображений, обеспечивающие устойчивость к состязательным атакам, инжекции неисправностей и дрейфу концепций.

Цель исследования – разработка эффективных архитектуры модели и алгоритма обучения, которые обеспечивают устойчивость к противоборствующим атакам, инжекции неисправностей и дрейфа концепций.

Методы исследования. Архитектура модели и алгоритм обучения реализуются путем сочетания идей и принципов самодистилляции знаний, максимизации информационной меры и компактности распределения классов, максимизации межклассового зазора, сжатия данных на основе дискретизации признакового представления, а также обучения с частичным привлечением учителя на основе регуляризации согласованности.

Результаты. Разработана архитектура модели и алгоритм обучения классификатора изображений. Полученный классификатор был испытан на наборе данных Cifar10 для оценивания его резильентности на интервале в 200 мини-пакетов с размером обучающего и тестового мини-пакета в 128 образцов для таких возмущений: состязательные L_∞-атаки чёрного ящика с уровнями 1, 3, 5 и 10; инверсия одного случайно выбранного бита в тензоре для 10%, 30%, 50% и 60% случайно выбранных тензоров; добавление одного нового класса; реальный дрейф концепции между парой классов. Рассмотрено влияние размерности пространства признаков на значение информационного критерия эффективности модели без возмущений и значение интегрального показателя резильентности во время воздействия возмущений.

Выводы. Предлагаемые архитектура модели и алгоритм обучения обеспечивают поглощение части возмущающего воздействия, изощренную деградацию за счет иерархичности классов и адаптивных вычислений, а также быструю адаптацию на ограниченном количестве размеченных данных. Показано, что адаптивные вычисления позволяют экономить до 40% ресурсов за счет раннего принятия решений на нижних секциях модели, однако возмущающее влияние приводит к замедлению, что можно рассматривать как изощренную деградацию. Доказано, что многосекционная структура, обучающаяся с использованием принципов самодистилляции знаний, обеспечивает более чем на 5% улучшение значения интегрального показателя резильентности по сравнению с архитектурой, где решение принимается на последнем слое модели. Замечено, что размерность пространства признаков заметно влияет на устойчивость к противоборствующим атакам и может выбираться как компромисс между резильентностью к возмущениям и эффективностью без возмущений.

КЛЮЧЕВЫЕ СЛОВА: классификация изображений, робастность, резильентность, утонченная деградация, состязательные атаки, инжекция неисправностей, дрейф концепций.

ЛІТЕРАТУРА/ЛИТЕРАТУРА

1. Towards Resilient Artificial Intelligence: Survey and Research Issues / [O. Eigner, S. Eresheim, P. Kieseberg et al.] // 2021 IEEE International Conference on Cyber Security and Resilience (CSR), Virtual conference, 26–28 July 2021. – P. 536–542. DOI: 10.1109/CSR51186.2021.9527986.
2. Olowononi F. O. Resilient Machine Learning for Networked Cyber Physical Systems: A Survey for Machine Learning Security to Securing Machine Learning for CPS / F. O. Olowononi, D. B. Rawat, C. Liu // IEEE Communications Surveys Tutorials. – 2021. – Vol. 23, No. 1. – P. 524–552. DOI: 10.1109/COMST.2020.3036778.
3. Dymond J. Graceful Degradation and Related Fields / J. Dymond // A review for Applied Research Centre at the Alan Turing Institute. – 2021. – P. 1–32. – DOI: 10.48550/arXiv.2106.11119.
4. Meta-Learning in Neural Networks: A Survey / [T. Hospedales, A. Antoniou, P. Micaelli, A. Storkey] // IEEE Transactions on Pattern Analysis and Machine Intelligence. – 2021. – 20 p. DOI: 10.1109/TPAMI.2021.3079209.
5. Continual lifelong learning with neural networks: A review / [G. Parisi, R. Kemker, J. Part et al] // Neural Networks. – 2019. – No. 113. – P. 54–71. DOI: 10.1016/j.neunet.2019.01.012
6. Fraccascia L. Resilience of Complex Systems: State of the Art and Directions for Future Research / L. Fraccascia, I. Giannoccaro, V. Albino // Complexity. – 2018. – P. 1–44. DOI: 10.1155/2018/3421529.
7. Madni A. Affordable Resilience / A. Madni // Transdisciplinary Systems Engineering. – 2017. – P. 133–159. DOI: 10.1007/978-3-319-62184-5_9.

DOI 10.15588/1607-3274-2022-3-9

8. Zhang L. Self-Distillation: Towards Efficient and Compact Neural Networks / L. Zhang, C. Bao, K. Ma // IEEE Transactions on Pattern Analysis and Machine Intelligence. – 2021. – Vol. 44 (8). – P. 4388–4403. DOI: 10.1109/TPAMI.2021.3067100.
9. Marquez E. Deep Cascade Learning / E. Marquez, J. Hare, M. Niranjan // IEEE Transactions on Neural Networks and Learning Systems. – 2018. – Vol. 29(11). – P. 5475–5485. DOI : 10.1109/TNNLS.2018.2805098.
10. Digital Image Representation by Atomic Functions: The Compression and Protection of Data for Edge Computing in IoT Systems / [V. Makarichev, V. Lukin, O.Illiashenko, V. Kharchenko] // Sensors. – 2022. – Vol. 22(10). – P. 3751. DOI : 10.3390/s22103751.
11. Smith L. N. A useful taxonomy for adversarial robustness of Neural Networks / L. N. Smith // Trends in Computer Science and Information Technology. – 2020. – P. 037–041. DOI: 10.48550/arXiv.1910.10679.
12. PixelDefend: Leveraging Generative Models to Understand and Defend against Advers arial Examples / [Y. Song, T. Kim, S. Nowozin et al.] // Sixth International Conference on Learning Representations, Vancouver CANADA, 30 Apr. –3 May, 2018. – 20 p. – DOI: 10.48550/arXiv.1710.10766.
13. Samangouei P. Defense-GAN: Protecting Classifiers Against Adversarial Attacks Using Generative Models / P. Samangouei, M. Kabkab, R. Chellappa // Sixth International Conference on Learning Representations (ICLR 2018), Vancouver CANADA, 30 Apr – 3 May, 2018. – 17 p. DOI: 10.48550/arXiv.1805.06605.
14. Athalye A. Obfuscated Gradients Give a False Sense of Security: Circumventing Defenses to Adversarial Examples. [online] arXiv.org / A. Athalye, N. Carlini, D.Wagner, – 2022. – Access mode: https://arxiv.org/abs/1802.00420 [Accessed 1 June 2022]. DOI: 10.48550/arXiv.1802.00420.
15. Reluplex made more practical: Leaky ReLU [Text] / [J. Xu, Z. Li, B. Du et al.] // IEEE Symposium on Computers and Communications (ISCC), Rennes, France, July 7–July 10 2020. – IEEE, 2022. – 7 p. DOI: 10.1109/ISCC50000.2020. 9219587.
16. Carlini N. Adversarial Examples Are Not Easily Detected: Bypassing Ten Detection Methods / N. Carlini, D. Wagner // Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security, Dallas Texas USA, 3 Nov. 2017. – NY, United States, 2017. – P. 3–14. DOI: 10.1145/3128572.3140444.
17. Silva S. Opportunities and Challenges in Deep Learning Adversarial Robustness: A Survey / S. Silva, P. Najafirad // IEEE Transactions on Knowledge and Data Engineering. – 2020. – 20 p. DOI: 10.48550/arXiv.2007.00753.
18. Huang K. Functional Error Correction for Robust Neural Networks / K. Huang, P. H. Siegel, A. Jiang // IEEE Journal on Selected Areas in Information Theory. – 2020. – 24 p. DOI: 10.48550/arXiv.2001.03814.
19. Training modern deep neural networks for memory-fault robustness / [G. B. Hacene, F. Leduc-Primeau, A. B. Soussia et al.] // IEEE International Symposium on Circuits and Systems (ISCAS 2019), Sapporo, Hokkaido, Japan, 26–29 May 2019. – 5 p. DOI: 10.1109/ISCAS.2019.8702382.
20. FTT-NAS: Discovering Fault-Tolerant Neural Architecture / [W. Li, X. Ning, G. Ge et al.] // Proceeding of 25th Asia and South Pacific Design Automation Conference (ASP-DAC), Beijing, China, 13–16 Jan. 2020. – IEEE Press – P. 211–216. DOI: 10.1109/ASP-DAC47756.2020.9045324.
21. Valtchev, S. Domain randomization for neural network classification / S. Valtchev, J. Wu // Journal of Big Data. – 2021. – Vol. 8, Article No. 94. – 12 p. DOI: 10.1186/s40537-021-00455-5.
22. Qiao F. Learning to Learn Single Domain Generalization / F. Qiao, L. Zhao, X. Peng // Computer Vision and Pattern Recognition. – 2020. – P. 1–13. DOI : 10.48550/arXiv.2003.13216.
23. Priya S. Deep learning framework for handling concept drift and class imbalanced complex decision-making on streaming data / S. Priya, R. Uthra // Complex & Intelligent Systems – 2021. – 17 p. DOI: 10.1007/s40747-021-00456-0.
24. To Trust Or Not To Trust A Classifier / [H. Jiang, B. Kim, M. Y. Guan, M. R. Gupta] // Proceedings of the 32nd International Conference on Neural Information Processing Systems. – 2018. – P. 5546–5557. DOI: 10.48550/arXiv.1805.11783.
25. P-ODN: Prototype-based Open Deep Network for Open Set Recognition / [Y. Shu, Y. Shi, Y. Wang et al.] // Scientific Reports. – 2020. – No. 10. – Article No. 7146. DOI: 10.1038/s41598-020-63649-6.
26. Detection and recovery against deep neural network fault injection attacks based on contrastive learning / [C. Wang, P. Zhao, S. Wang, X. Lin] // 3rd Workshop on Adversarial Learning Methods for Machine Learning and Data Mining at KDD, Virtual Event, USA ,14 Aug 2021. – 5 p.
27. Achddou R. Nested Learning for Multi-Level Classification / R. Achddou, J. Di Martino, G. Sapiro // IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toronto, Canada, 6–11 June 2021. – P. 2815–2819. DOI: 10.1109/ICASSP39728.2021.9415076.
28. Active Learning by Acquiring Contrastive Examples / [K. Margatina, G. Vernikos, L. Barrault, N. Aletras] // Conference on Empirical Methods in Natural Language Processing, Online and Punta Cana, Dominican Republic, Nov. 2021. – P. 650–663. DOI: 10.48550/arXiv.2109.03764.
29. OpenCoS: Contrastive Semi-supervised Learning for Handling Open-set Unlabeled Data / [J. Park, S. Yun, J. Jeong, J. Shin] // International Conference on Learning Representations ICLR Virtual, 3–7 May 2022. – 14 p. DOI: 10.48550/arXiv.2107.08943.
30. Konkle T. A self-supervised domain-general learning framework for human ventral stream representation / T. Konkle, G. Alvarez // Nature Communications. – 2022. – Vol. 13, Article No. 491. – 12 p. DOI: 10.1038/s41467-022-28091-4.
31. Multi-stage deep learning method with self-supervised pretraining for sewer pipe defects classification / [V. Moskalenko, M. Zaretskyi, A. Moskalenko et al.] // Radioelectronic and computer systems. – 2021. – No. 4. – P. 71–81. DOI: 10.32620/reks.2021.4.06.
32. Li G. TensorFI: A Configurable Fault Injector for TensorFlow Applications / G. Li, K. Pattabiraman, N. DeBardeleben // 2018 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW), Memphis, TN, USA, 15–18 Oct. 2018. – P. 313–320. – DOI: 10.1109/ISSREW.2018.00024.\
33. Kotyan, S. Adversarial robustness assessment: Why in evaluation both L0 and L∞ attacks are necessary [Text] / S. Kotyan, D. Vargas // PLOS ONE. – 2022. – No. 17(4), Article No. e0265723. – 22 p. DOI: 10.1371/journal.pone.0265723.

DOI 10.15588/1607-3274-2022-3-9

UDC 004.85

MULTILINGUAL TEXT CLASSIFIER USING PRE-TRAINED UNIVERSAL SENTENCE ENCODER MODEL

Orlovskiy O. V. – Post-graduate student, Computer Systems Software Department, Yuriy Fedkovych Chernivtsi National University, Chernivtsi, Ukraine.

Khalili Sohrab – CEO, CreateITTogether LLC Company, Fullerton, CA, USA.

Ostapov S. E. – Professor, Head of Computer Systems Software Department, Yuriy Fedkovych Chernivtsi National University, Chernivtsi, Ukraine.

Hazdyuk K. P. – Assistant, Computer Systems Software Department, Yuriy Fedkovych Chernivtsi National University, Chernivtsi, Ukraine.

Shumylyak L. M. – Assistant, Computer Systems Software Department, Yuriy Fedkovych Chernivtsi National University, Chernivtsi, Ukraine.

ABSTRACT

Context. Online platforms and environments continue to generate ever-increasing content. The task of automating the moderation of user-generated content continues to be relevant. Of particular note are cases in which, for one reason or another, there is a very small amount of data to teach the classifier. To achieve results under such conditions, it is important to involve the classifier pre-trained models, which were trained on a large amount of data from a wide range. This paper deals with the use of the pre-trained multilingual Universal Sentence Encoder (USE) model as a component of the developed classifier and the affect of hyperparameters on the classification accuracy when learning on a small data amount (~ 0.05% of the dataset).

Objective. The goal of this paper is the investigation of the pre-trained multilingual model and optimal hyperparameters influence for learning the text data classifier on the classification result.

Method. To solve this problem, a relatively new approach to few-shot learning has recently been used – learning with a relatively small number of examples. Since text data is still the dominant way of transmitting information, the study of the possibilities of constructing a classifier of text data when learning from a small number of examples (~ 0.002–0.05% of the data set) is an actual problem.

Results. It is shown that even with a small number of examples for learning (36 per class) due to the use of USE and optimal configuration in learning can achieve high accuracy of classification on English and Russian data, which is extremely important when it is impossible to collect your own large data set. The influence of the approach using USE and a set of different configurations of hyperparameters on the result of the text data classifier on the example of English and Russian data sets is evaluated.

Conclusions. During the experiments, a significant degree of relevance of the correct selection of hyperparameters is shown. In particular, this paper considered the batch size, optimizer, number of learning epochs and the percentage of data from the set taken to train the classifier. In the process of experimentation, the optimal configuration of hyperparameters was selected, according to which 86.46% accuracy of classification on the Russian-language data set and 91.13% on the English-language data, respectively, can be achieved in ten seconds of training (training time can be significantly affected by technical means used).

KEYWORDS: few shot learning, low-data learning, pre-trained models, USE, neural networks, data mining, data set, text data classifier.

ABBREVIATIONS

USE is a Universal Sentence Encoder;

SGD is a Stochastic gradient descent;

RMSProp is a Root Mean Squared Propagation;

Adam is a Adaptive Moment Optimization.

NOMENCLATURE

O_T is a set of optimizer's type;

o_T is an element of an set of optimizer's type;

P is a parameters set;

p_j is an element of a parameters set;

N_{par} is a parameters number;

P' is a specific parameters set for each training subset;

M is a toxic messages dataset;

m_i is a toxic message;

M^k is a training subset of the toxic messages;

L is a language of dataset;

S is a size of dataset (in MB);

N_S is a number of records in dataset;

N_{cat} is a classification categories number in the dataset;

N_{sam} is a proportion of the original samples in training subsample (in %);

N_{ep} is a number of executed epochs of neural network training;

Ac is a classification accuracy;

$F()$ is a function depends on M, M^k, P' which describes Ac;

max is a function $F()$ maximum.

INTRODUCTION

Deep learning systems using large amounts of data have repeatedly shown their effectiveness in a wide range of classification problems [1]. However there are often situations in which it seems impossible to prepare a sufficient number of marked examples for classifier training or requires the involvement of resources that do not justify the expected end result. To solve this problem, a relatively new approach to few-shot learning has recently been used – learning with a relatively small number of examples. Since text data is still the dominant way of transmitting information [2], the study of the possibilities of constructing a classifier of text data when learning

DOI 10.15588/1607-3274-2022-3-10

from a small number of examples (~ 0.002 – 0.05% of the data set) is an urgent task.

Another important bonus for improving the efficiency of development time will be the ability to classify text simultaneously in several of the most popular languages using a single model. In particular, this paper investigates the results of the model's work on texts created in Russian and English.

The object of study is the process of toxic message classification.

The subject of study is the investigation of the pre-trained USE-model on the classification accuracy.

The purpose of the work is the development and investigation of the multilanguage classificator on the base of pre-trained USE-model.

1 PROBLEM STATEMENT

The challenge facing the authors of the paper is as follows. For each specific set of toxic messages $M=\{m_i\}$, where $i=1,\ldots, N_S$, it is necessary to select the training subset $M^k \in M$ and choose the best optimizer type $o_T \in O_T$ (with the parameter set $P=\{p_j\}$, where $j=1, \ldots, N_{\text{par}}$) so that specific parameters values $P' \in P \mid \forall M^k \in M$ made it possible to achieve the maximum classification accuracy, i.e. $Ac=\text{F}(M, M^k \in M, P') \rightarrow \max$ for each classifier type $o_T \in O_T$. An additional condition imposed on the data subset is that its amount does not exceed 0.05% of the complete dataset, such as $N_{\text{sam}} \leq 0.00005\, N_S$.

2 REVIEW OF THE LITERATURE

In our previous paper, an overview of typical approaches used in the development of text data classifiers, in particular on the example of the classification of destructive messages [3] was made. Special attention was paid to the problem of the data preprocessing methods affect for learning process.

This paper deals with the study of the influence of the pre-trained USE model on the accuracy of the classification of text messages with learning process, which uses only several examples per class.

The paper [4] discusses the problem of data augmentation in a small data subset. Initially, the classifier uses several original examples per class, and then several artificially created examples, which aim, if possible, to comprehensively reflect the features of a particular class. Thus, it is expected that several universal artificial examples will help to replace the lack of a large number of instances, each of which reflects a certain aspect of the class in its own way.

Research [5] helps us to better understand how few shot models work in general, how different approaches to their construction differ, what are the advantages and disadvantages of this class of models developed over the last few years. Also in the work special attention is paid to the use of transformers, which is relevant for our model.

The article [6] deals with the affect of pre-trained models when they are used as components of the model. The results obtained by the authors for the problem of text generation by involving a previously trained model in the developed generator encourage us to investigate the effect that such a solution may have for the classification problem.

The problem of classes optimization in the classification process requires special attention, especially with regard to their quantity, potential merger or replacement. This can also greatly affect both the speed of classifier development and the data preparation. Details of the classes composition and their potential modification are demonstrated in [7].

The article [8] demonstrates the examples of the pre-trained model from Google – Universal Sentence Encoder (USE) using [9]. In particular, a wide range of tasks for which the model can be used is shown, where the task of classifying text data is only one of the possibilities.

Investigation in the paper [10] demonstrate the inclusion of the optimal hyperparameters choice in classifier training, including studies of the effectiveness of various optimizers of the data, such as Adam and its modifications.

As mentioned earlier the main purpose of this paper is to study the influence of the pre-trained multilingual model and the optimal parameters for learning the text data classifier on the classification result. To solve the problem, a classifier based on an artificial neural network was used. One of the network layers will be the pre-trained USE model [9]. Different configurations of hyperparameters were tested during the training. The classification results were verified on the two data sets described below.

3 MATERIALS AND METHODS

The experiments were performed using two datasets. The first – "Fake or real news dataset" [11] has the following characteristics, presented in Table 1.

Table 1 – Characteristics of the data set "Fake or real news dataset"

Characteristic	Value
Language, *L*	English
Dataset size, *S*	~ 29 MB
Number of samples, *Ns*	~ 6335
Number of classification cathegories, *Ncat*	1 (fake)

The second dataset, "Russian Language Toxic Comments. Small dataset with labeled comments from 2ch.hk and pikabu.ru" [12], has the following characteristics presented in Table 2.

Note that although both datasets are intended for the classification problem, these tasks are somewhat different. In the first case, we find "fake" or real news, and in the second – toxic or not a certain message.

Data for training process on both datasets were distributed as shown in Table 3.

DOI 10.15588/1607-3274-2022-3-10

Table 2 – Characteristics of the dataset "Russian Language Toxic Comments. Small dataset with labeled comments from 2ch.hk and pikabu.ru"

Characteristic	Value
Language, *L*	Russian
Dataset size, *S*	4.45 MB
Number of samples, *Ns*	~ 11.500
Number of classification cathegories, *Ncat*	1 (toxic)

Table 3 – Data distribution for training process

Training stage	Data distribution
Training process	0.002 – 0.05%
Validation/testing process	99.998 – 99.95%

The investigations were performed using a neural network, the architecture of which is schematically shown in Fig. 1.

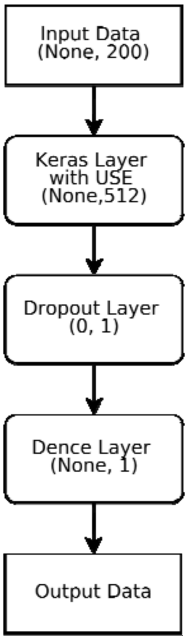


Figure 1 – Scheme of neural network architecture based on USE model

As can be seen from the figure, our classifier contains three layers. The main one is KerasLayer, which includes a massive pre-trained USE model [9]. The Dropout-Dense layer combination block helps to avoid re-learning the classifier and helps to reduce the dimension of the network and, as a result, speed up learning.

Consider the architecture in more detail. A list of English and Russian sentences (depending on the data set) of different lengths is transmitted to the input in the presented classifier based on the neural network. It is known that to learn the network, we can not use as input elements of the word in their usual form, we perform the following manipulations on the data set:

1. Tokenization. For example: we transform every object that looks like "Hello, gentlemen!" to an array of unique words ["hello", "gentlemen"] without punctuation.

2. Indexing. We create a dictionary from the resulting array of all unique words in the dataset, which looks like this: {1: "hello", 2: "gentlemen", ...}.

3. Indexes representation. For each of the objects in the dataset, we form an array in which the words involved in the object are represented as indexes from our dictionary. For example: [1, 2, 1].

Also, one of the typical problems we have to face when preparing data for training is the problem of different sizes of objects among our data. We need to bring the learning elements to the same dimension. This is solved using the padding technique: choose the maximum value of words, such as 200, and fill the remaining spaces in each object with zeros. If the object contains more words than the selected maximum value – each subsequent word after the maximum is cut off. Although in the data sets we have chosen, the size of most sentences does not exceed the value of 100 words, nevertheless, the dimension with a value of 200 is chosen to capture atypical cases, if any.

The basis of the presented classifier is KerasLayer, which is connected to the pre-trained USE model [13] based on the "transformer" architecture presented by researchers from Google in 2019. This model exists in several variations, but in these experiments a multilingual version was chosen (16 languages, including English and Russian). Also noteworthy is the fact that the purpose of the model is not only to classify but also to cluster texts, find their semantic similarities, as well as some multilingual operations. In the experiments conducted, the developed nature of the model related to multilingual classification was useful.

Having received from USE the resulting tensor, we transfer it to the Dropout layer. Its purpose, in this case, is to weed out a certain percentage of nodes, which we will establish, replacing them with a value of zero. This is necessary so that nodes at the next level are forced to process missing data representations. In this way we achieve an effect in which the result of the whole network has the best level of generalization – we avoid the effect of retraining, in which the network can show good results on a familiar data set and far from the desired results on an unfamiliar set. In the presented experiments the value of random exclusion of nodes in 10% was used. Of course, this percentage can be selected empirically.

After passing the Dropout layer we transfer the data to the Dense layer and the RELU activation function is applied to them. Then the result passes through the sigmoidal activation function, where the classification for each of the labels takes place, and we get a value between 0 and 1.

Of course, the presented architecture can be optimized, but this is more of a challenge for the future. Now our task is to determine the influence of the pre-trained USE-model on the results of the classifier.

DOI 10.15588/1607-3274-2022-3-10

4 EXPERIMENTS

Note first of all that in the experiments, a combination of different sets of optimizers, loss functions and other hyperparameters was tested. Unless otherwise noted, the default optimizer was Adam, a loss function: binary crossentropy.

Initially, the classifier was trained in the "basic mode" on 0.002% of examples from the dataset (few-shot learning). The batch parameter is equal 4. It is optimal taking into account the hardware used for training. Number of epochs – 2. With such settings, we obtained the accuracy of the classification in the range of 73.85–74.17%. In this case, and further we mean the range obtained by repeated experiments with the same parameters in the samples described in Table 1 and Table 2.

Next, we conducted an iterative experiment consisting of the following steps:

1. We change one of the key parameters that may affect the resulting classification accuracy (batch; number of epochs in training; dataset percentage included in the sample for training (train); used optimizer). Note that we consider this list not exhaustive of possible options. Nevertheless, the influence of these parameters is investigated in the experiments presented in this paper.
2. We teach the classifier on the selected dataset without changing other parameters.
3. Measure the classification accuracy.

The experiment's results are presented in the next chapter (see Table 4).

5 RESULTS

Note that the configuration with the Adamax optimizer proved to be the best in the considered experiments (№7 in Table 4). We obtained the maximum classification accuracy of 0.9113 on the English-language dataset [11] by repeating the experiment with the same parameters.

Table 4 – The results of the classifier when using different hyperparameters on datasets [11, 12]

No	O_T	N_{ep}	N_{sam}	Ac
1	Adam	2	0.002%	0.7385–0.7417
2	Adam	7	0.002%	0.7826–0.6771
3	Adam	2	0.005%	0.7074–0.7460
4	Adam	10	0.005%	0.6651–0.7657
5	SGD	2	0.005%	0.5552 – 0.6649
6	NAdam	2	0.005%	0.8027 – 0.8104
7	Adamax	2	0.005%	0.8475 – 0.8646
8	RMSProp	2	0.005%	0.7773 – 0.7839

6 DISCUSSIONS

Analyzing the results described in Table 4, we immediately note the key advantage of the few-shot learning approach. Using only 0.002% of samples N_{sam} and two learning epochs N_{ep}, we obtained a quite acceptable result of classification accuracy Ac in the range 0.7385 – 0.7417%. This amount of data used and the number of epochs can significantly reduce network learning time. Depending on the used hardware, the speed of the learning process can vary, however, we can safely say about ten seconds to complete the experiment. This can be extremely relevant when prototyping a certain idea on selected data, when you need to get a quick result and already starting from it to build a further, more detailed experiment. Also, such a scenario may be quite applicable in an area where it is impossible or impractical to collect a relatively large data amount for classifier training, and the value of a quick result on a relatively small amount of "live" data is significant.

Continuing to experiment, we noted that if the number of epochs increases to 7 while maintaining the previous values in the above configuration, we can observe an expansion of the range of classification accuracy (#2 in Table 4). In particular, the lower accuracy limit fell by 6.14%, and the upper accuracy limit increased by only 4.09%. At the same time, after graduating from the 7th epoch, the classifier steadily came to a state of reduced accuracy. Examining this question, we came to the conclusion that it is necessary to continue the selection of the optimal configuration of hyperparameters. In experiments #3 and #4, we tried to increase the number of examples for training to 0.005% of the data samples. As we can see in Table 4, the result is slightly different, but the general trend repeats the result of experiments #1 and #2.

The next hyperparameter for selection was the optimizer type N_{cat}. We first used the SGD optimizer (experiment #5 in Table 4) for two epochs and 0.005% of the sample data. However, the best result in the range (0.6649) was 4.25% behind the worst result obtained with the Adam optimizer for the same other experimental parameters. In our opinion, this is most likely due to the fact that this optimizer performs better when working with other types of data, in contrast to text data in our experiments.

In Experiment #8 we usied the RMSProp optimizer and improved the result obtained with Adam while maintaining other parameters at the same level. Based on the lower bar of the accuracy range, we can reached an improvement of 6.99%. However, the best results in our experiments were achieved using modifications of the Adam optimizer. In particular, using NAdam, we recorded improvements in the lower bar of the accuracy range by 9.53%, and with Adamax by as much as 14.01%! The results are shown in Table 4 in experiments #6 and #7, respectively. Based on the obtained results, we consider this configuration with the described hypeparameters to be optimal when training the classifier on text data.

DOI 10.15588/1607-3274-2022-3-10

CONCLUSIONS

The few-shot learning approach is extremely relevant in a large number of domains, where collecting and preparing a large set of data for learning seems impractical.

The universal knowledge base taken out of the cognition of our datasets is the pre-trained multilingual USE model, which allows simultaneous work with data in 16 languages, of which 2 are used in this work.

In our experiments, the optimal configuration of hyperparameters was selected, according to which 86.46% accuracy of classification on the Russian-language data set and 91.13% on the English-language data, respectively, can be achieved in ten seconds of training (training time can be significantly affected by technical means used).

The scientific novelty. It is shown that even with a small number of examples for learning (36 per class) due to the use of USE and optimal configuration in learning can achieve high accuracy of classification on English and Russian data, which is extremely important when it is impossible to collect your own large dataset.

The practical significance. The obtained results allow to build classifiers of text data with a sufficiently high rate of accuracy in the presence of a small amount of data for learning.

Prospects for further research. In the following studies, you can take into account more hyperparameters to analyze their impact on the final result of the classifier. It is also quite relevant to compare the influence of different pre-trained analog models according to USE, which we relied on in conducting all the experiments described in this paper.

The urgent problem of mathematical support development is solved to automate the sampling at diagnostic and recognizing model building by precedents.

The scientific novelty of obtained results is that the method of training sample selection is firstly proposed. It determines the weights characterizing the term and feature usefulness for a given initial sample of precedents and given feature space partition. It characterizes the individual absolute and relative informativity of instances relative to the centers and the boundaries of feature intervals based on the weight values. This allows to automate the sample analysis and its division into subsamples, and, as a consequence, to reduce the training data dimensionality. This in turn reduces the time and provides an acceptable accuracy of neural model training.

The practical significance of obtained results is that the software realizing the proposed indicators is developed, as well as experiments to study their properties are conducted. The experimental results allow to recommend the proposed indicators for use in practice, as well as to determine effective conditions for the application of the proposed indicators.

Prospects for further research are to study the proposed set of indicators for a broad class of practical problems.

ACKNOWLEDGEMENTS

This investigation is supported by the state budget scientific research project of Yuriy Fedkovych Chernivtsi National Univesity "Investigation, simulation and software development for complex dynamic systems" (state registration number state 0121U109232).

REFERENCES

1. Yann L., Yoshua B., Geoffrey H. Deep learning, *Nature,* 2015, Vol. 521(7553), pp. 436–444.
2. Ma L., Goharian N., Chowdhury A. et al. Extracting unstructured data from template generated web documents, *Information and knowledge management, Twelfth international conference, 2003, proceedings*, 2003, pp. 512–515.
3. Orlovskyi O., Ostapov S. Analysis of the text preprocessing methods influence on the destructive messages classifier, O.Orlovskyi, *Advanced Information Systems,* 2020, Vol. 4(3), pp.104–108.
4. Few-Shot Text Classification with Triplet Networks, Data Augmentation, and Curriculum Learning [Electronic resource], Access mode: https://arxiv.org/abs/2103.07552
5. A Neural Few-Shot Text Classification Reality Check [Electronic resource]. Access mode: https://arxiv.org/abs/2101.12073
6. Few-Shot Text Generation with Pattern-Exploiting Training [Electronic resource]. Access mode: https://arxiv.org/abs/2012.11926
7. Halder K., Akbik A., Krapac J. et al. Task-Aware Representation of Sentences for Generic Text Classification, *Computational Linguistics, 28th International Conference, December 2020, proceedings*, 2020, P. 3202–3213.
8. Reddy T., Williams R., Breazeal C. Text classification for AI education [Electronic resource]. Access mode: https://robots.media.mit.edu/wp-content/uploads/sites/7/2021/01/Text_classifier.pdf
9. Universal-sentence-encoder-multilingual-large. 16 languages (Arabic, Chinese-simplified, Chinese-traditional, English, French, German, Italian, Japanese, Korean, Dutch, Polish, Portuguese, Spanish, Thai, Turkish, Russian) text encoder [Electronic resource]. Access mode: https://tfhub.dev/google/universal-sentence-encoder-multilingual-large/3
10. Sriporn K., Tsai C. F., Tsai C. E. et al. Analyzing Malaria Disease Using Effective Deep Learning Approach, *Diagnostics*, 2020, No. 10, pp. 744–749.
11. Fake or real news dataset [Electronic resource]. Access mode:https://github.com/lutzhamel/fake-news/blob/master/data/fake_or_real_news.csv.
12. Russian Language Toxic Comments. Small dataset with labeled comments from 2ch.hk and pikabu.ru [Electronic resource]. Access mode: https://www.kaggle.com/blackmoon/russian-language-toxic-comments.
13. Yang Y.. Cer D., Ahmad A. et al. Multilingual Universal Sentence Encoder for Semantic Retrieval, [Electronic resource]. Access mode: https://aclanthology.org/2020.acl-demos.12.pdf

Received 14.01.2022.
Accepted 22.06.2022.

DOI 10.15588/1607-3274-2022-3-10

УДК 004.85

МУЛЬТИМОВНИЙ КЛАСИФІКАТОР ТЕКСТУ З ВИКОРИСТАННЯМ ПРЕДТРЕНОВАНОЇ МОДЕЛІ UNIVERSAL SENTENCE ENCODER

Орловський О. В. – аспірант, кафедра програмного забезпечення комп'ютерних систем Чернівецького національного університету імені Юрія Федьковича, Чернівці, Україна.

Khalili, Sohrab – CEO, CreateITTogether LLC Company, Fullerton, Каліфорнія, США.

Остапов С .Е. – професор, завідувач кафедри програмного забезпечення комп'ютерних систем Чернівецького національного університету імені Юрія Федьковича, Чернівці, Україна.

Газдюк К. П. – ассистент, кафедра програмного забезпечення комп'ютерних систем Чернівецького національного університету імені Юрія Федьковича, Чернівці, Україна.

Шумиляк Л. М. – ассистент, кафедра програмного забезпечення комп'ютерних систем Чернівецького національного університету імені Юрія Федьковича, Чернівці, Україна.

АНОТАЦІЯ

Актуальність. Онлайн-платформи продовжують сьогодні генерувати усе більші обсяги інформації. Автоматизація модерування контенту у таких платформах, у зв'язку з цим, залишається актуальною задачею. Особливої уваги потребують випадки, коли з різних причин, доступно лише невеликі обсяги даних для навчання класифікаторів. У таких випадках необхідно залучати попередньо навчені моделі, які використовували для навчання великі об'єми даних широкого діапазону. У цій роботі досліджено питання застосування попередньо навченої мультимовної моделі Universal Sentence Encoder (USE) як компоненту розробленого нами класифікатора, а також впливу різних параметрів на точність класифікації при навчанні на малому об'ємі даних (~ 0,05% обсягу повного набору).

Метод. Для вирішення поставленого завдання використовується відносно новий підхід до навчання, – за допомогою невеликого набору повідомлень. Оскільки текстові повідомлення усе ще домінують як спосіб передавання інформації, застосовується розроблений класифікатор, навчений на невеликому (~ 0,002 – 0,05% повного набору) обсязі даних.

Результати. Показано, що навіть при невеликій кількості прикладів для навчання (36 на клас) за рахунок використання ЕСЕ та оптимальної конфігурації в навчанні можна досягти високої точності класифікації за англійськими та російськими даними, що надзвичайно важливо, коли неможливо зібрати свій власний великий набір даних. Оцінено вплив підходу з використанням USE та набору різних конфігурацій гіперпараметрів на результат класифікатора текстових даних на прикладі наборів даних англійською та російською мовами.

Висновки. У ході експериментів показана значна ступінь актуальності правильного підбору гіперпараметрів. Зокрема, у цій роботі розглядалися розмір пакету, оптимізатор, кількість епох навчання та відсоток даних із набору, взятих для навчання класифікатора. У процесі експерименту була обрана оптимальна конфігурація гіперпараметрів, згідно з якою 86,46% точності класифікації за російськомовним набором даних і 91,13% за англомовним відповідно можна досягти за десять секунд навчання (на час навчання можуть істотно вплинути використовувані технічні засоби).

КЛЮЧОВІ СЛОВА: few shot learning, навчання при малій кількості даних, предтреновані моделі, USE, нейронні мережі, інтелектуальний аналіз даних, набір даних, класифікатор текстових даних.

УДК 004.85

МУЛЬТИЯЗЫЧНЫЙ КЛАССИФИКАТОР ТЕКСТА С ИСПОЛЬЗОВАНИЕМ ПРЕДВАРИТЕЛЬНО ТРЕНИРОВАННОЙ МОДЕЛИ UNIVERSAL SENTENCE ENCODER

Орловский А. В. – аспирант, кафедра программного обеспечения компьютерных систем Черновицкого национального унивеситета имени Юрия Федьковича, Украина.

Khalili Sohrab – CEO, CreateITTogether LLC Company, Fullerton, Калифорния, США.

Остапов С. Э. – профессор, заведующий кафедрой программного обеспечения компьютерных систем Черновицкого национального унивеситета имени Юрия Федьковича, Украина.

Газдюк Е. П. – ассистент, кафедра программного обеспечения компьютерных систем Черновицкого национального унивеситета имени Юрия Федьковича, Украина.

Шумиляк Л. М. – ассистент, кафедра программного обеспечения компьютерных систем Черновицкого национального унивеситета имени Юрия Федьковича, Украина.

АННОТАЦИЯ

Актуальность. Онлайн-платформы продолжают сегодня генерировать все более возрастающие объемы информации. Задачи автоматизации модерирования контента ползователей в связи с этим остается актуальной задачей. Особого внимания заслуживают случаи, когда, по разным причинам, доступны очень небольшие объемы данных для обучения классификатора. Для достижения приемлемых результатов необходимо применять предварительно обученные модели, которые использовали большие объемы данных широкого диапазона для предварительного обучения. В данной работе исследуется вопрос применения предварительно обученной мультиязыковой модели Universal Sentence Encoder (USE) в качестве компонента разработанного нами классификатора, а также влияния различных параметров на точность классификации при обучении на малом объема данных (~ 0,05% набора данных).

Метод. Для решения поставленной задачи используется относительно новый подход к обучению – по небольшой выборке сообщений. Поскольку текстовые сообщения все еще доминируют как способ передачи информации, использование классификатора текстовых данных при обучении на небольшой выборке (~ 0,002–0,05% набора данных) сообщений.

DOI 10.15588/1607-3274-2022-3-10

Результаты. Показано, что обучение даже на небольшой выборке (36 на класс) с использованием USE и оптимальной конфигурации при обучении можно достичь высокой верности классификации англо- и русскоязычных текстовых сообщений. Выполнена оценка влияния разных наборов гиперпараметров на результаты классификации.

Выводы. В ходе экспериментов показана актуальность правильного подбора гиперпараметров: размер пакета, тип оптимизатора, количество эпох, размер обучающей выборки. При оптимальных значениях гиперпараметров достигнута вероятность распознавания англоязычных деструктивных сообщений в 91,13%, при этом обучение проводилось всего на протяжении 10 секунд (что, безусловно, зависит от параметров использованных технических средств).

КЛЮЧЕВЫЕ СЛОВА: few shot learning, обучение на малой выборке данных, предварительно обученные модели, USE, нейронные сети, интеллектуальный анализ данных, набор данных, классификатор текстовых данных.

ЛІТЕРАТУРА / ЛИТЕРАТУРА

1. Yann L. Deep learning / L. Yann, B. Yoshua, H. Geoffrey // Nature. – 2015. – Vol. 521(7553). – P. 436–444.
2. Extracting unstructured data from template generated web documents / [L. Ma, N. Goharian, A. Chowdhury et al.] // Information and knowledge management: Twelfth international conference, 2003: proceedings, 2003. – P. 512–515.
3. Orlovskyi O. Analysis of the text preprocessing methods influence on the destructive messages classifier / O. Orlovskyi, S. Ostapov // Advanced Information Systems. – 2020. – Vol. 4(3). – P. 104–108.
4. Few-Shot Text Classification with Triplet Networks, Data Augmentation, and Curriculum Learning [Electronic resource] – Access mode: https://arxiv.org/abs/2103.07552
5. A Neural Few-Shot Text Classification Reality Check [Electronic resource]. – Access mode: https://arxiv.org/abs/2101.12073
6. Few-Shot Text Generation with Pattern-Exploiting Training [Electronic resource]. – Access mode: https://arxiv.org/abs/2012.11926
7. Task-Aware Representation of Sentences for Generic Text Classification / [K. Halder, A. Akbik, J. Krapac et al.] // Computational Linguistics: 28th International Conference, December 2020: proceedings. – 2020. – P. 3202–3213.
8. Reddy T. Text classification for AI education [Electronic resource] / T. Reddy, R. Williams, C. Breazeal. – Access mode: https://robots.media.mit.edu/wp-content/uploads/sites/7/2021/01/Text_classifier.pdf
9. Universal-sentence-encoder-multilingual-large. 16 languages (Arabic, Chinese-simplified, Chinese-traditional, English, French, German, Italian, Japanese, Korean, Dutch, Polish, Portuguese, Spanish, Thai, Turkish, Russian) text encoder [Electronic resource]. – Access mode: https://tfhub.dev/google/universal-sentence-encoder-multilingual-large/3
10. Sriporn K. Analyzing Malaria Disease Using Effective Deep Learning Approach / K. Sriporn, C. F. Tsai, C. E. Tsai et al. // Diagnostics. – 2020. – No. 10. – P. 744–749.
11. Fake or real news dataset [Electronic resource]. – Access mode:https://github.com/lutzhamel/fake-news/blob/master/data/fake_or_real_news.csv.
12. Russian Language Toxic Comments. Small dataset with labeled comments from 2ch.hk and pikabu.ru [Electronic resource]. – Access mode: https://www.kaggle.com/blackmoon/russian-language-toxic-comments.
13. Yang Y. Multilingual Universal Sentence Encoder for Semantic Retrieval / Y. Yang, D. Cer, A. Ahmad et al. [Electronic resource] – Access mode: https://aclanthology.org/2020.acl-demos.12.pdf

DOI 10.15588/1607-3274-2022-3-10

ПРОГРЕСИВНІ ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

PROGRESSIVE INFORMATION TECHNOLOGIES

ПРОГРЕССИВНЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

UDC 004.2 : 004.94

SYNTHESIS OF THE FINITE STATE MACHINE WITH DATAPATH OF TRANSITIONS ACCORDING TO THE OPERATIONAL TABLE OF TRANSITIONS

Barkalov A. A. – Dr. Sc., Professor, Professor of Institute of Computer Science and Electronics, University of Zielona Gora, Zielona Gora, Poland.

Titarenko L. A. – Dr. Sc., Professor, Professor of Institute of Computer Science and Electronics, University of Zielona Gora, Zielona Gora, Poland.

Babakov R. M. – Dr. Sc., Associate Professor, Associate Professor of department of information technologies, Vasyl Stus Donetsk National University, Vinnytsia, Ukraine.

ABSTRACT

Context. The problem of formalizing the description of a microprogram finite state machine based on the principle of operational transformation of state codes with the help of a modified transition table is considered. The object of research was a model of a microprogram finite state machine with datapath of transitions.

Objective. The goal of the work is development and research of a method for formally specifying a microprogram finite state machine with datapath of transitions in the form of a modified table of transitions containing sufficient information for synthesizing the logic circuit of the finite state machine in the basis of programmable logic devices.

Method. A new way of representing the formal solution of the problem of algebraic synthesis of a microprogram finite state machine with datapath of transitions in the form of an operational table of transitions is proposed. This table is a modification of the direct structural table traditionally used in the synthesis of microprogram finite state machines. The use of the previously known representation of the formal solution of the problem of algebraic synthesis in the form of a system of isomorphisms of automaton algebras is too formalized and makes it difficult to synthesize the logical circuit of the finite state machine due to the separate description of the transition and output functions. It is shown that the structure of a microprogram finite state machine with datapath of transitions requires information about the set of interpretations of state codes and the transition operations used to be entered into the traditional table of transitions. It is noted that the proposed operational table of transitions contains sufficient information for the synthesis of the logical circuit of the finite state machine. An example of constructing an operational table of transitions for a finite state machine given by a graph-scheme of the implemented control algorithm is shown. The example demonstrates various ways to interpret state codes. The procedure for synthesizing the circuit for generating codes of transition operations and the circuit for generating microoperations according to the operational table of transitions is proposed.

Results. An example of the implementation of the main stages of the synthesis of a finite state machine with datapath of transitions according to the operational table of transitions is considered. Examples of synthesized finite state machine models in the VHDL language are given, which take into account the peculiarities of the representation of finite state machine models in Xilinx Vivado CAD. The results of the synthesis of the finite state machine according to VHDL models in FPGA basis are shown.

Conclusions. The experiments carried out confirmed the sufficiency of the operational table of transitions for describing a microprogram finite state machine with operational transformation of state codes for the purpose of further synthesizing its logic circuit. Prospects for further research are the use of the proposed operational table of transitions in the development of various methods for the synthesis and optimization of microprogram finite state machine with operational transformation of state codes.

KEYWORDS: finite state machine, datapath of transitions, table of transitions, synthesis of logical circuit, graph-scheme of algorithm.

ABBREVIATIONS

FSM is a finite state machine;
DT is a datapath of transitions;
GSA is a graph-scheme of algorithm;
OTT is an operational table of transitions;
BMO is a block of microoperations;
PLD is a programmable logic device.

DOI 10.15588/1607-3274-2022-3-11

NOMENCLATURE

a_m is a current FSM state;

$K(a_m)$ is a current FSM state code;

a_s is a transition state;

$K(a_s)$ is a transition state code;

X is a set of logic conditions;

x_i is an element of set X;

L is a number of logic conditions (power of set X);

Y is a set of microoperations;

y_i is an element of set Y;

N is a number of microoperations (power of set Y);

Φ_h is a set of input memory functions for switching FSM memory from state a_m to state a_s;

h is a number of FSM transition;

A is a set of FSM states;

a_i is an element of set A;

M is a number of FSM states (power of set A);

B is a number of transitions of FSM;

O is a set of transition operations;

O_i is an element of se O;

P is a number of transition operations (power of set P);

W is e set of signals of code of transitions operation;

w_i is an element of set W;

R_W is a digit capacity of code of transitions operation (power of set W);

I is a number of interpretations of state codes of FSM;

T is a set of signals of curretnt state code of FSM;

T_i is an element of set T;

R is a digit capacity of state code (power of set T).

INTRODUCTION

Digital systems are widely used in various fields of activity [1]. Structurally, the digital system can be considered as a combination of operational unit and control unit [2–3]. The control unit is based on a formal description of behavior and can be implemented in the form of a finite state machine model [4–5]. There are two models of finite state machines – the Mealy machine and the Moore machine [4–5]. The logic circuit of any FSM model is characterized by such parameters as hardware expenses, clock frequency and power consumption. As follows from [6], there is a direct relationship between these characteristics. Optimization of the characteristics of FSM circuits is an important scientific and practical problem, the solution of which is devoted to many scientific papers around the world [1–7]. One of such characteristics, which focuses on the finite state machine structure considered in this paper, is the hardware expenses of implementing the logic circuit of the FSM in a given element basis.

The object of study is the process of synthesis of the logic circuit of a finite state machine with operational transformation of state codes.

This process in the case of a canonical finite state machine is performed according to a direct structural table, which is a formal description of the behavior of the FSM and contains sufficient information for the synthesis of its circuit. In the case of operational transformation of state codes, this table requires modifications taking into account the processes of information transformation that takes place in this class of FSM.

The subject of study is the finite state machine with operational transformation of state codes, in which the transformation of state codes is carried out using a finite set of arithmetical and logical operations.

The purpose of the work is formalization of the description of the finite state machine with operational transformation of state codes in the form of the modified direct structural table.

1 PROBLEM STATEMENT

Suppose given finite state machine, characterized by the sets $A=\{a_1, ..., a_M\}$, $X=\{x_1, ..., x_L\}$, $Y=\{y_1, ..., y_N\}$ and implements a certain control algorithm. The synthesis of the logic circuit of the FSM provides for the implementation of the function of transitions $T=T(X, T)$ and the function of outputs $Y=Y(X, T)$ in the given elementary basis.

The paper solves the problem of synthesizing a finite state machine with datapath of transitions, in which the transition function $T=T(T, W)$ depends on the code of the current state and the code of transitions operation. To solve the problem, it is necessary to develop a formalized representation of the FSM with DT, which allows the following stages of the synthesis of the logic circuit of the FSM:

– synthesis of each structural blocks;

– construction of VHDL description of the synthesized FSM;

– FSM synthesis using Xilinx Vivado CAD using FPGA.

2 REVIEW OF THE LITERATURE

Various optimization methods for reducing of hardware expenses of FSM circuit are known today. Such methods include, for example, methods of structural decomposition [7]. Their use leads to FSM circuits with several levels of conversion of logic signals.

Another approach to reducing hardware amount, considered in this article, is to use the principle of operational transformation of state codes [8]. According to it, the transformation of state codes in an FSM is not carried out with the help of a canonical system of Boolean equations, but with using of a set of arithmetical and logicalal operations that form a special datapath of transitions. This structure of FSM with DT shows a fairly high efficiency in terms of hardware expenses [9].

In the work [10] considers an algebraic model of FSM with DT, according to which this FSM can be represented as a system of isomorphisms of partial algebras (transition algebras). Each transition algebra describes the rule of transformation of state codes for its subset of FSM transitions and assumes its own scalar or vector interpretation of state codes. The system of isomorphisms of algebras today is the only formal way to specify the FSM with DT. However, the synthesis of the FSM circuit directly by the system of isomorphisms of algebras is complicated due to different representations of FSM transitions.

DOI 10.15588/1607-3274-2022-3-11

A direct structural table is traditionally used to specify the FSM behavior [2, 7]. The method of synthesis of FSM according to the table of transitions is widely known and applied in practice [7]. This article proposes a new way to specify the FSM with DT, based on a modified table of transitions.

3 MATERIALS AND METHODS

The canonical FSM is usually set in the form of a direct structural table (table of transitions), the format of which is presented in Fig. 1 [2]. The purpose of columns of the table of transitions and its use for the synthesis of the FSM circuit are described in [7].

a_m	$K(a_m)$	a_s	$K(a_s)$	X_h	Y_h	Φ_h	h
...	...	...	...	...	...	...	...

Figure 1 – Structure of the table of transitions

Let the FSM be given in the form of a GSA *G*, [7], which is shown in Fig. 2, left. In the right part of Fig. 2 shows a description of GSA *G* in the *kiss* format, which is used to describe finite state machines in the test collection LGSynth93 [11]. GSA *G* is marked by the states of the Moore FSM, contains $M = 10$ states $a_0 - a_9$, $L = 3$ input signals $x_1 - x_3$, $N = 4$ output signals $y_1 - y_4$ and $B = 13$ FSM transitions. To encode 10 states, it is sufficient to use $R = 4$ binary digits.

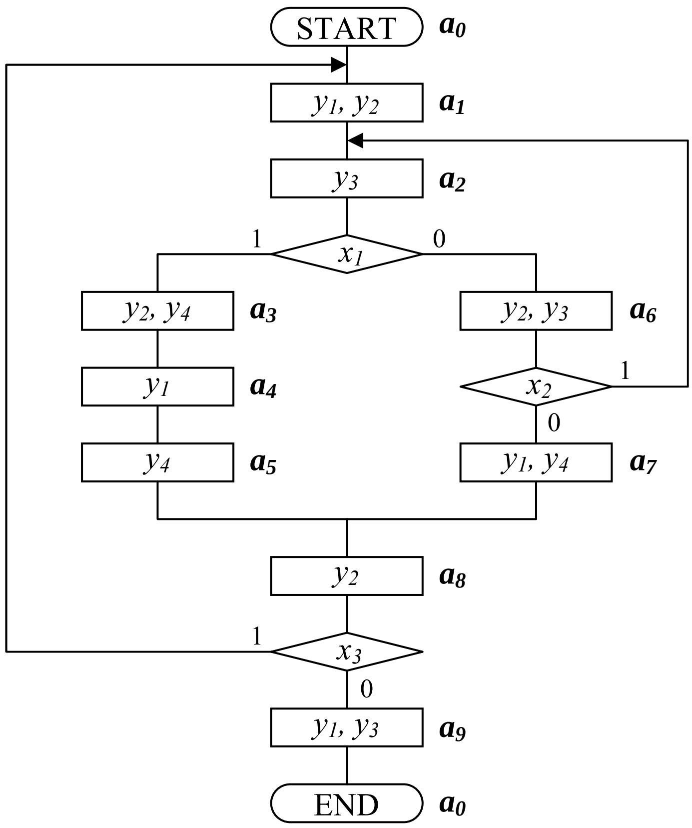
Let's synthesize for GSA *G* a finite state machine with datapath of transitions. Suppose the next transitions operations $O_1 - O_3$ are given:

$$O_1: \quad K(a_s) = K(a_m) + 9_{10}; \tag{1}$$

$$O_2: \quad K(a_s) = K(a_m) \ \& \ 1000_2; \tag{2}$$

$$O_3: \quad K(a_s) = K(a_m) \oplus 0100_2. \tag{2}$$

In operation O_1, a decimal constant 9 is added to the current state code $K(a_m)$. This means that a scalar decimal interpretation is used for the codes $K(a_m)$ and $K(a_s)$ when performing O_1. In operations O_2 and O_3 over the code $K(a_m)$ bitwise logical operations with binary constant are performed. Therefore, the codes $K(a_m)$ and $K(a_s)$ when performing operations O_2, O_3 are interpreted as binary vectors. Thus, when using operations (1)–(3) for state codes will be used two different interpretations: decimal number and binary vector. Note that the arguments and values of operations (1)–(3) are in the range [0; 15], i.e. in the range of numbers specified by four-digit binary codes. For example, in the circuit implementation of the operation "+9" the result is always modulo 16 (in fact, the lower four binary digits of the result are taken).




```
.i 3
.o 4
.p 12
.s 10
--- a0 a1 0000
--- a1 a2 1100
1-- a2 a3 0010
0-- a2 a6 0010
--- a3 a4 0101
--- a4 a5 1000
--- a5 a8 0001
-1- a6 a2 0110
-0- a6 a7 0110
--- a7 a8 1001
--1 a8 a1 0100
--0 a8 a9 0100
--- a9 a0 1010
```

Figure 2 – Graph-scheme of algorithm *G*

DOI 10.15588/1607-3274-2022-3-11

Let's perform an algebraic synthesis of FSM with DT, which is as follows:

– to each state of the FSM we will match the unique four-digit binary vector having the corresponding decimal interpretation;

– for each FSM transition we will match any operation from (1) – (3), which for the given state codes transforms the code $K(a_m)$ into the code $K(a_s)$.

The result of algebraic synthesis in graphical form is shown in Fig. 3. In each vertex, which is marked by the Moore FSM state, the state code is shown in scalar (decimal) and vector (binary) interpretations. Each FSM transition is marked by one of the operations (1) – (3): "+9", "& 1000" or "⊕ 0100". Since the operational transformation of state codes affects only the function of the FSM transitions and does not affect the function of the outputs, the microoperations in Fig. 3 are not shown, although they continue to correspond to Fig. 2.

As we can see, with the chosen values of state codes, all transitions within the GSA G are implemented using of operations (1)–(3). For example, the transition from state a_3 with code $K(a_3) = 9_{10} = 1001_2$ to state a_4 with code $K(a_4) = 2_{10} = 0010_2$ is performed using the operation "+9", and from the result $18_{10} = 10010_2$ the lower four digits 0010_2 were taken.

Let's modify the table of transitions of the canonical FSM as follows:

1. Instead of column $K(a_m)$ add columns $K_1(a_m)$, $K_2(a_m)$, ..., $K_I(a_m)$, which correspond to all used interpretations of state codes for all I interpretations.

2. Do the same with column $K(a_s)$, adding columns $K_1(a_s)$, $K_2(a_s)$, ..., $K_I(a_s)$ instead.

3. Add a column W_h containing information about the code of transitions operation that implements current FSM transition. The values of w_i specified in this column correspond to the values 1 in the binary code of the corresponding operation. Filling in this column is preceded by the step of encoding of transitions operations.

4. Remove the column Φ_h, because the conversion of state codes in the FSM with DT is carried out using a set of transitions operations, rather than a system of canonical equations of the transition function.

Let's call the received table as the operational table of transitions (OTT). In the general case, its structure corresponds to Fig. 4.

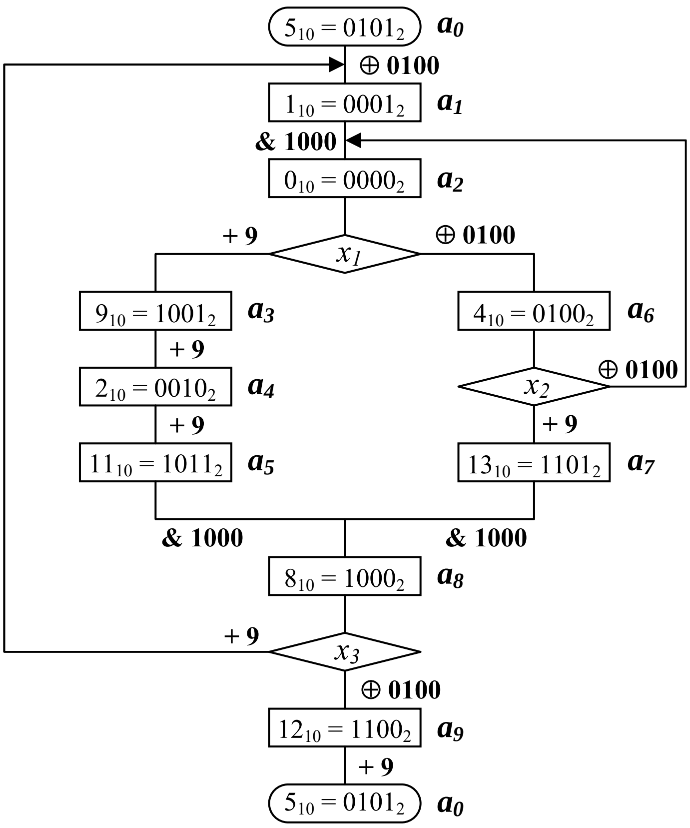


Figure 3 – Result of algebraic synthesis for GSA G and operations (1)–(3)

a_m	$K_1(a_m)$	...	$K_I(a_m)$	a_s	$K_1(a_s)$	...	$K_I(a_s)$	X_h	W_h	Y_h	h

Figure 4 – Structure of operational table of transitions

DOI 10.15588/1607-3274-2022-3-11

Let's present in the form of OTT the results of algebraic synthesis of the FSM, shown in Fig. 3. To do this, we encode the given transitions operations $O_1 - O_3$ with unique binary codes of bit size $R_W = \rceil \log_2 3 \lceil = 2$, which are formed by signals $W = \{w_1, w_2\}$. The result of coding is presented in Table 1. Taking into account the coding of transitions operations, the operational table of transitions, hich corresponds to GSA G and Fig. 3, is presented by Table 2.

Table 1 – Coding of operations $O_1 - O_3$

O_i	w_1 w_2
O_1	0 0
O_2	0 1
O_3	1 0

In this example, the number of interpretations of the state codes $I = 2$ (scalar decimal value and binary vector). When using the transition operation O_1, the code $K_1(a_m)$ is converted into the code $K_1(a_s)$; when using operations O_2 and O_3, the code $K_2(a_m)$ is converted into the code $K_2(a_s)$. For example, the transition $h = 5$ is realized by operation O_1. Therefore, in this transition, the conversion of scalar interpretations of codes is performed, i.e. the code $K_1(a_m) = 9_{10}$ into the code $K_1(a_s) = 2_{10}$. This transformation is performed using the adder circuit with the preservation of four lower digits of the result.

Note that the dash in the column W_h means that for the implementation of the corresponding transition, values $w_i = 1$ are not formed, which corresponds to the code of operation O_1 ($w_1 = 0$, $w_2 = 0$).

4 EXPERIMENTS

Let's show an example of Table 1, that the information contained in the OTT is sufficient for the synthesis of the logic circuit of the FSM with DT. The structural model of the FSM with DT, corresponding to the Moore FSM, is shown in Fig. 5 and contains the next synthesized blocks:

– the block W generates a set of signals W with digit capacity $R_W = \rceil P \lceil$ according to formula (4), where P is the number of transition operations; X is the set of L input signals of the FSM corresponding to the logical conditions $x_1, ..., x_l$ of the given GSA; T – state code of the FSM with digit capacity R

$$W = W(X, T); \quad (4)$$

– the block DT corresponds to the datapath of transitions and implements a set of transitions operations in the form of a set of separate combinational circuits, the outputs of which are multiplexed by the signal W and enter the FSM memory register that is part of the DT [8];

– the block BMO corresponds to the cicruit of formation of microoperations and implements the output function of the Moore FSM in the form of a set of microoperations $Y = \{y_1, ..., y_n\}$ according to formula (5) by analogy with [2, 7]

$$Y = Y(T). \quad (5)$$

Let's synthesize these blocks. Under the synthesis of the logic circuit of the machine we will understand the development of VHDL-model, which can be synthesized in the element basis of Xilinx FPGA [12].

Table 2 – Operational table of transitions (GSA G)

a_m	$K_1(a_m)$	$K_2(a_m)$	a_s	$K_1(a_s)$	$K_2(a_s)$	X_h	W_h	Y_h	h
a_0	5	0101	a_1	1	0001	1	w_1	–	1
a_1	1	0001	a_2	0	0000	1	w_2	y_1, y_2	2
a_2	0	0000	a_3	9	1001	x_1	–	y_3	3
			a_6	4	0100	$\bar{x}_1$	w_1		4
a_3	9	1001	a_4	2	0010	1	–	y_2, y_4	5
a_4	2	0010	a_5	11	1011	1	–	y_1	6
a_5	11	1011	a_8	8	1000	1	w_2	y_4	7
a_6	4	0100	a_2	0	0000	x_2	w_2	y_2, y_3	8
			a_7	13	1101	$\bar{x}_2$	–		9
a_7	13	1101	a_8	8	1000	1	w_2	y_1, y_4	10
a_8	8	1000	a_1	1	0001	x_3	–	y_2	11
			a_9	12	1100	$\bar{x}_3$	w_1		12
a_9	12	1100	a_0	5	0101	1	–	y_1, y_3	13

DOI 10.15588/1607-3274-2022-3-11

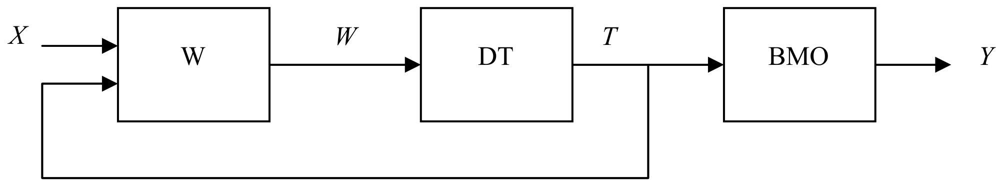


Figure 5 – Structural model of FSM with DT

Block W

This block implements a system of Boolean equations of function (4), in which each signal wi is formed in accordance with expression (6)

$$w_i = \bigvee_{h=1}^{H} C_{ih} A_m^h X_h \quad (i = \overline{1, R_W}) \tag{6}$$

In this formula, C_{ih} is a Boolean variable equal to 1, if and only if the function w_i is written in the OTT line number h; A_m^h is a conjunction of internal variables, corresponding to the binary representation of state code a_m from the line number h (for example, $A_5^7 = T_1\overline{T}_2T_3T_4$); X_h is a conjunction of signals of logical conditions, written in the column X_h of the line number h ($X_h = 1$ for unconditional transitions).

For this example, the system of equations of function (4) is the next:

$$\begin{cases} w_1 = \overline{T}_1T_2\overline{T}_3T_4 \vee \overline{T}_1\overline{T}_2\overline{T}_3\overline{T}_4\overline{x}_1 \vee T_1\overline{T}_2\overline{T}_3\overline{T}_4\overline{x}_3; \\ w_2 = \overline{T}_1\overline{T}_2\overline{T}_3T_4 \vee T_1\overline{T}_2T_3T_4 \vee \overline{T}_1T_2\overline{T}_3\overline{T}_4x_2 \vee T_1T_2\overline{T}_3T_4. \end{cases} \tag{7}$$

In the general case for the system (7) it is possible to carry out minimization in order to reduce the complexity of the circuit [4, p. 269].

System (7) can be described in VHDL in different ways, for example, as a separate process (Fig. 6) [12].

In this model, the description of the buses *T*, *X* and *W* corresponds to the same FSM signals and is discussed below in the description of the architecture block.

Block DT

This block includes an operational part that implements operations (1) – (3) and their multiplexing, as well as a memory register designed to store the current FSM state. The functional diagram of these nodes is shown in Fig. 7. Since for the considered example the circuit of DT consists of standard functional blocks, special synthesis of this circuit is trivial and is not required.

In Fig. 8 VHDL-model of OAP, in which the operational part and the memory register are represented by separate processes, is showed.

The first process corresponds to the operational part of DT. The absence of the synchronization signal *C* in the list of sensitivity of the process indicates the asynchronous nature of the operation of this block. Within the process transformation of the state code *T* is performed using one of three transitions operations depending on the value of the operation code *W*. As will be shown below, for the signal *T* used data type "unsigned", which allows you to interpret this signal simultaneously as an unsigned integer and as a binary vector.

The second process corresponds to the memory register. Receiving data in the register, as well as the switching to the initial state 0101_2 in the presence of the *Reset* signal are carried out synchronously on the leading edge of the *C* signal.

Block BMO

The synthesis of this block is performed in accordance with the contents of column Y_h of the operational table of transitions (Table 2). In this case, to obtain a synthesized VHDL model of the block, it is possible to use the method considered for block *W* (build a system of Boolean equations for generated microoperations), or use the *case* operator belonging to the synthesized subset of the VHDL language. We use the second method, as a result of which we obtain the VHDL model of the BMO block, shown in Fig. 9.

In this model, after the start of the process, all digits of the output bus *Y* are given zero values. Then, depending on the values of the signals on the bus *T*, the required discharges of the bus *Y* are set to 1. It is expected that state code values not provided by the case operator should not appear on the *T* bus.

```
process (T, X)    -- Block W
begin
    W(1) <= (not(T(1)) and T(2) and not(T(3)) and T(4)) or
       (not(T(1)) and not(T(2)) and not(T(3)) and not(T(4)) and not(X(1))) or
            (T(1) and not(T(2)) and not(T(3)) and not(T(4)) and not(X(3)));

    W(2) <= (not(T(1)) and not(T(2)) and not(T(3)) and T(4)) or
            (T(1) and not(T(2)) and T(3) and T(4)) or
            (not(T(1)) and T(2) and not(T(3)) and not(T(4)) and X(2)) or
            (T(1) and T(2) and not(T(3)) and T(4));
end process;
```

Figure 6 – VHDL model of block W

DOI 10.15588/1607-3274-2022-3-11

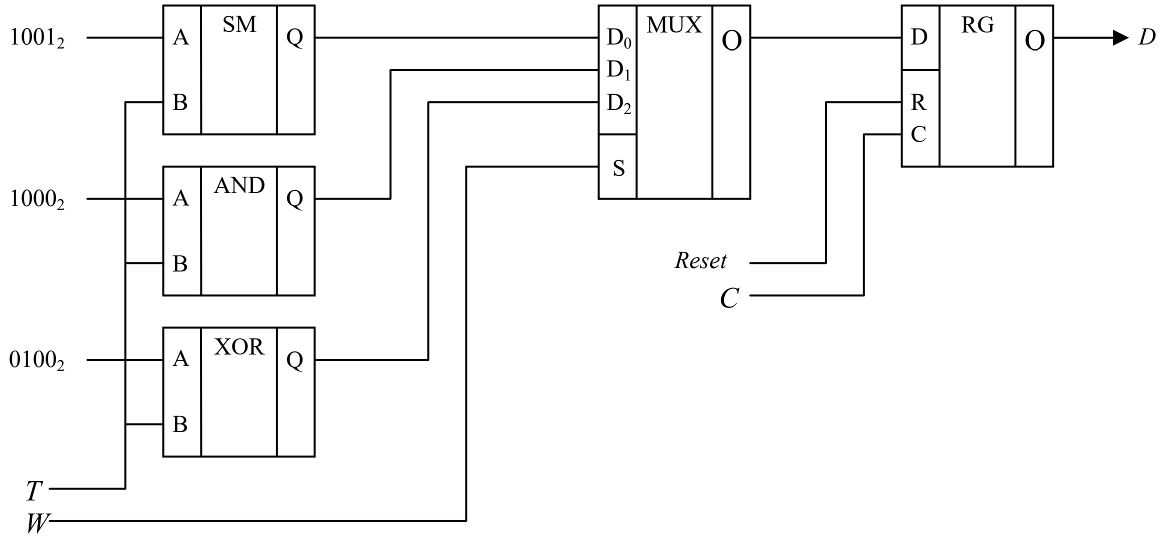


Figure 7 – Functional diagram of datapath of transitions

```
process (W, T)           -- Datapath of Transitions
begin
    if W = "00" then
        D <= T + 9;
    elsif W = "01" then
        D <= T and "1000";
    else
        D <= T xor "0100";
    end if;
end process;


process (C)              -- State Register
begin
    if rising_edge(C) then
       if Reset = '1' then
           T <= "0101";
       else
           T <= D;
       end if;
    end if;
end process;
```

Figure 8 – VHDL model of datapath of transitions

```
process (T)                    -- Microoperations
begin
    Y <= (others => '0');
    case T is
        when "0001" => Y(1) <= '1'; Y(2) <= '1';
        when "0000" => Y(3) <= '1';
        when "1001" => Y(2) <= '1'; Y(4) <= '1';
        when "0010" => Y(1) <= '1';
        when "1011" => Y(4) <= '1';
        when "0100" => Y(2) <= '1'; Y(3) <= '1';
        when "1101" => Y(1) <= '1'; Y(4) <= '1';
        when "1000" => Y(2) <= '1';
        when "1100" => Y(1) <= '1'; Y(3) <= '1';
        when others => null;
    end case;
end process;
```

Figure 9 – VHDL model of block BMO

DOI 10.15588/1607-3274-2022-3-11

Let's combine the considered VHDL descriptions into a single object entity, resulting in a synthesized model of FSM with DT (Fig. 10). A feature of this model is the presence of the output port *S*, which displays the code of the current state *T*. This is done in order to analyze functioning of the FSM in the process of behavioral modeling.

5 RESULTS

Synthesis of the VHDL model shown in Fig. 10, in CAD Xilinx Vivado 2021.1 allowed to obtain hardware expenses for implementation of the FSM, equal to 7 LUT-elements (based on FPGA xc7a12ticsg325-1L FPGA of Artix-7 series).

To test the correctness of VHDL model of the FSM with DT, a behavioral part was developed that implements the next functionality:

– single generation of the *Reset* signal in the range of 10–90 *ns* from the beginning of the simulation;

– regular signal *C* generation with a duration of 20 *ns* with an interval of 100 *ns*;

– regular generation of input signals $x_1 – x_3$ with different values of lengths of 1 and 0 levels.

Fig. 10 shows a fragment of the time diagram of the FSM in the process of behavioral modeling. This fragment demonstrates the correctness of the transition function and the output function of the FSM. For example, at

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use ieee.numeric_std.all;
-- ------------------------------------------------------------------------
entity OAP is
    generic (R: integer := 4;  L: integer := 3;
             RW: integer := 2; N: integer := 4);

    port (X: in std_logic_vector (1 to L);
          C: in std_logic;
          Reset: in std_logic;
          S: out unsigned (1 to R);
          Y: out std_logic_vector (1 to N));
end entity OAP;
-- ------------------------------------------------------------------------
architecture OAP_A of OAP is
signal T, D: unsigned (1 to R);
signal W: std_logic_vector (1 to RW);
begin
    process (C)            -- State register
    ...
    process (T, X)         -- Block W
    ...
    process (W, T)         -- Datapath of transitions
    ...
    process (T)            -- Block of microoperations
    ...
    S <= T;                -- For debugging
end architecture OAP_A;
```

Figure 10 – Synthesizable VHDL model of FSM with DT

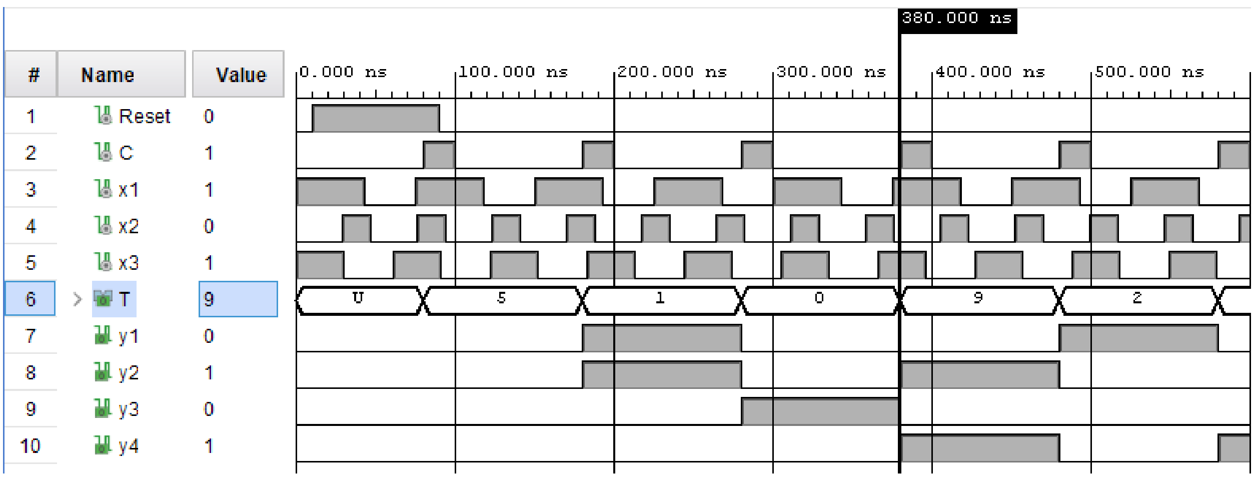


Figure 10 – Fragment of time diagram of behavioral modelling of work of FSM with DT (in interval from 0 to 600 ns)

DOI 10.15588/1607-3274-2022-3-11

time t = 380 *ns*, marked by a vertical marker, the FSM performs transition from the state with the code $T = 0$ (state a_2), analyzing value of signal x_1. Since at this point $x_1 = 1$, the transition is carried out to a state with code $T = 9$ (state a_3), which is consistent with Fig. 3. After the transition to state a_3, the formation of microoperations y_2 and y_4 is carried out, which is consistent with Fig. 2. Thus, the synthesis of FSM with DT for this example is performed correctly.

6 DISCUSSION

A finite state machine with datapath of transitions differs from a canonical finite state machine in that it uses a set of arithmetical and logical operations (transitions operations) to convert state codes, which form a datapath of transitions. Each transitions operation involves a certain interpretation of binary state codes and is formally specified on a set of interpreted values. One of the interpretations is a scalar representation in the form of an unsigned integer, which allows to define on the set of state codes operations of addition, subtraction, and so on. Interpretation of the state code in the form of a binary vector allows to specify over the state codes bitwise logical operations, shift operations and the others. Information about the used operations and methods of interpretation of binary state codes is necessary for the schematic implementation of the transition function of the FSM.

The input data for the synthesis of the logic circuit of the canonical FSM is a table of transitions (direct structural table), which contains information about the functions of transitions and outputs of the FSM. However, its use for FSM with DT is impossible due to the lack of information about the methods of interpretation of state codes and transitions operations. In this paper, it is proposed to use the so-called operational table of transitions for specification the FSM with DT, which provides extended information about the function of transitions of the FSM. The example considered in the paper showed that specification of the FSM with DT in the form of an operational table of transitions is sufficient for the synthesis of the logic circuit of the FSM in the form of VHDL model focused on the use element basis of FPGA-type.

The possibility of obtaining with the help of VHDL model numerical characteristics of hardware expenses for the implementation of the FSM circuit allows us to recommend the use of the operational table of transitions as a way to present the results of other known methods of hardware expenses in FSM circuit.

CONCLUSIONS

The article proposes a solution of scientific problem of formalizing the description of the processes of state codes transformation in a finite state machine with datapath of transitions, which allows to bring the description of this class of finite state machine in line with traditional description of other classes of finite state machines.

The scientific novelty of the work is to modify the direct structural table by adding to it information about algebraic interpretation and methods of transformation of state codes. The resulting operational table of transitions is proposed for the first time and contains sufficient information for the synthesis of the logic circuit of the FSM.

Practical use of the obtained results is possible in the development of formal methods of structural synthesis of finite state machines with operational transformation of state codes.

Prospects for further research are to develop methods of synthesis of FSM circuit, based on the formal representation of the FSM in the form of operational table of transitions.

ACKNOWLEDGEMENTS

The work is supported by the state budget scientific research project of Vasyl' Stus Donetsk National University "Methods, algorithms and tools of computer-aided design of control units of computing systems" (state registration number 0122U200085).

REFERENCES

1. Bailliul J., Samad T. Encyclopedia of Systems and Control. Springer, London, UK, 2015, 1554 p.
2. Sklyarov V., Sklyarova I., Barkalov A., Titarenko L. Synthesis and Optimization of FPGA-Based Systems; Volume 294 of Lecture Notes in Electrical Engineering. Springer, Berlin, Germany, 2014, 432 p.
3. Baranov S. Logic and System Design of Digital Systems. Tallin, TUTPress, 2008, 267 p.
4. Micheli, G. D. Synthesis and Optimization of Digital Circuits. McGraw-Hill, Cambridge, MA, USA, 1994, 579 p.
5. Minns P., Elliot I. FSM-Based Digital Design Using Verilog HDL. JohnWiley and Sons. Hoboken, NJ, USA, 2008, 408 p.
6. Grout I. Digital Systems Design with FPGAs and CPLDs. Elsevier Science, Amsterdam, The Netherlands, 2011, 784 p.
7. Baranov S. Logic Synthesis for Control Automata. Dordrecht, Kluwer Academic Publishers, 1994, 312 p.
8. Barkalov A. A., Babakov R. M. Operational formation of state codes in microprogram automata, *Cybernetics and Systems Analysis*, 2011, Volume 47 (2), pp. 193–197.
9. Barkalov A. A., Babakov R. M. Determining the Area of Efficient Application of a Microprogrammed Finite-State Machine with Datapath of Transitions, *Cybernetics and Systems Analysis,* 2019,Volume 54 (3), pp. 366–375.
10. Barkalov A. A., Babakov R. M. Algebraic Interpretation of a Microprogram Finite-State Machine with Datapath of Transitions, *Cybernetics and Systems Analysis,* 2016, Volume 54 (3), pp. 366–375.
11. McElvain, K. LGSynth93 Benchmark; Mentor Graphics. Wilsonville, OR, USA, 1993.
12. Xilinx. XST UserGuide. V.11.3. Available online: https://www.xilinx.com/support/documentation/sw_manuals /xilinx11/xst.pdf (accessed on 07 June 2022).

Received 17.06.2022.
Accepted 12.08.2022.

DOI 10.15588/1607-3274-2022-3-11

УДК 004.2 : 004.94

СИНТЕЗ МІКРОПРОГРАМНОГО АВТОМАТА З ОПЕРАЦІЙНИМ АВТОМАТОМ ПЕРЕХОДІВ ЗА ОПЕРАЦІЙНОЮ ТАБЛИЦЕЮ ПЕРЕХОДІВ

Баркалов О. О. – д-р техн. наук, професор, професор Інституту комп'ютерних наук та електроніки університету Зеленогурського, м. Зельона Гура, Польща.

Тітаренко Л. О. – д-р техн. наук, професор, професор Інституту комп'ютерних наук та електроніки університету Зеленогурського, м. Зельона Гура, Польща.

Бабаков Р. М. – д-р техн. наук, доцент, доцент кафедри інформаційних технологій Донецького національного університету імені Василя Стуса, м. Вінниця, Україна.

АНОТАЦІЯ

Актуальність. Розглянуто задачу формалізації опису мікропрограмного автомата, заснованого на принципі операційного перетворення кодів станів, за допомогою модифікованої таблиці переходів. Об'єктом дослідження була модель мікропрограмного автомата з операційним автоматом переходів. Мета роботи – розробка та дослідження способу формального завдання мікропрограмного автомата з операційним автоматом переходів у вигляді модифікованої таблиці переходів, що містить достатню інформацію для синтезу логічної схеми автомата в базисі програмувальних логічних пристроїв.

Метод. Запропоновано новий спосіб представлення формального рішення задачі алгебраїчного синтезу мікропрограмного автомата з операційним автоматом переходів у вигляді операційної таблиці переходів. Ця таблиця є модифікацією прямої структурної таблиці, зазвичай використовуваної при синтезі мікропрограмних автоматів. Використання раніше відомого представлення формального вирішення задачі алгебраїчного синтезу у вигляді системи ізоморфізмів автоматних алгебр є надто формалізованим і ускладнює синтез логічної схеми автомата внаслідок роздільного опису функцій переходів і виходів. Показано, що структура мікропрограмного автомата з операційним автоматом переходів вимагає внесення до традиційної таблиці переходів інформації про множину інтерпретацій кодів станів і використовувані операції переходів. Зазначено, що запропонована операційна таблиця переходів містить достатню інформацію для синтезу логічної схеми автомата. Наведено приклад побудови операційної таблиці переходів для автомата, заданого граф-схемою імплементованого алгоритму керування. У прикладі показані різні методи інтерпретації кодів станів. Запропоновано порядок синтезу схеми формування кодів операцій переходів та схеми формування мікрооперацій за операційною таблицею переходів.

Результати. Розглянуто приклад виконання основних етапів синтезу мікропрограмного автомата з операційним автоматом переходів за операційною таблицею переходів. Наведено приклади моделей синтезованого автомата мовою VHDL, які враховують особливості представлення моделей кінцевих автоматів у САПР Xilinx Vivado. Показано результати синтезу автомата за VHDL-моделями у базисі ПЛІС FPGA.

Висновки. Проведені експерименти підтвердили достатність операційної таблиці переходів для опису мікропрограмного автомата з операційним перетворенням станів кодів з метою подальшого синтезу його логічної схеми. Перспективи подальших досліджень полягають у використанні запропонованої операційної таблиці переходів при розробці різних методів синтезу та оптимізації мікропрограмних автоматів з операційним перетворенням кодів станів.

КЛЮЧОВІ СЛОВА: мікропрограмний автомат, операційний автомат переходів, таблиця переходів, синтез логічної схеми, граф-схема алгоритму.

УДК 004.2 : 004.94

СИНТЕЗ МИКРОПРОГРАММНОГО АВТОМАТА С ОПЕРАЦИОННЫМ АВТОМАТОМ ПЕРЕХОДОВ ПО ОПЕРАЦИОННОЙ ТАБЛИЦЕ ПЕРЕХОДОВ

Баркалов А. А. – д-р техн. наук, профессор, профессор Института компьютерных наук и электроники университета Зеленогурского, г. Зелена Гура, Польша.

Титаренко Л. А. – д-р техн. наук, профессор, профессор Института компьютерных наук и электроники университета Зеленогурского, г. Зелена Гура, Польша.

Бабаков Р. М. – д-р техн. наук, доцент, доцент кафедры информационных технологий Донецкого национального университета имени Васыля Стуса, г. Винница, Украина.

АННОТАЦИЯ

Актуальность. Рассмотрена задача формализации описания микропрограммного автомата, основанного на принципе операционного преобразования кодов состояний, с помощью модифицированной таблицы переходов. Объектом исследования являлась модель микропрограммного автомата с операционным автоматом переходов. Цель работы – разработка и исследование способа формального задания микропрограммного автомата с операционным автоматом переходов в виде модифицированной таблицы переходов, содержащей достаточную информацию для синтеза логической схемы автомата в базисе программируемых логических устройств.

Метод. Предложен новый способ представления формального решения задачи алгебраического синтеза микропрограммного автомата с операционным автоматом переходов в виде операционной таблицы переходов. Данная таблица является модификацией прямой структурной таблицы, традиционно используемой при синтезе микропрограммных автоматов. Использование ранее известного представления формального решения задачи алгебраического синтеза в виде системы изоморфизмов автоматных алгебр является слишком формализованным и затрудняет синтез логической схемы автомата по причине раздельного описания функций переходов и выходов. Показано, что структура микропрограммного автомата с

DOI 10.15588/1607-3274-2022-3-11

операционным автоматом переходов требует внесения в традиционную таблицу переходов информации о множестве интерпретаций кодов состояний и используемых операциях переходов. Отмечено, что предложенная операционная таблица переходов содержит достаточную информацию для синтеза логической схемы автомата. Приведен пример построения операционной таблицы переходов для автомата, заданного граф-схемой имплементируемого алгоритма управления. В примере продемонстрированы различные способы интерпретации кодов состояний. Предложен порядок синтеза схемы формирования кодов операций переходов и схемы формирования микроопераций по операционной таблице переходов.

Результаты. Рассмотрен пример выполнения основных этапов синтеза микропрограммного автомата с операционным автоматом переходов по операционной таблице переходов. Даны примеры моделей синтезированного автомата на языке VHDL, которые учитывают особенности представления моделей конечных автоматов в САПР Xilinx Vivado. Показаны результаты синтеза автомата по VHDL-моделям в базисе ПЛИС FPGA.

Выводы. Проведенные эксперименты подтвердили достаточность операционной таблицы переходов для описания микропрограммного автомата с операционным преобразованием кодов состояний с целью дальнейшего синтеза его логической схемы. Перспективы дальнейших исследований заключаются в использовании предложенной операционной таблицы переходов при разработке различных методов синтеза и оптимизации микропрограммных автоматов с операционным преобразованием кодов состояний.

КЛЮЧЕВЫЕ СЛОВА: микропрограммный автомат, операционный автомат переходов, таблица переходов, синтез логической схемы, граф-схема алгоритма.

ЛІТЕРАТУРА / ЛИТЕРАТУРА

1. Bailliul J. Encyclopedia of Systems and Control / J. Bailliul, T. Samad. – Springer : London, UK, 2015. – 1554 p.
2. Synthesis and Optimization of FPGA-Based Systems; Volume 294 of Lecture Notes in Electrical Engineering / [V. Sklyarov, I. Sklyarova, A. Barkalov, L. Titarenko]. – Springer : Berlin, Germany, 2014. – 432 p.
3. Baranov S. Logic and System Design of Digital Systems / S. Baranov. – Tallin : TUTPress, 2008. – 267 p.
4. Micheli G. D. Synthesis and Optimization of Digital Circuits / G. D. Micheli. – McGraw-Hill : Cambridge, MA, USA, 1994. – 579 p.
5. Minns, P. FSM-Based Digital Design Using Verilog HDL / P. Minns, I. Elliot. – JohnWiley and Sons: Hoboken, NJ, USA, 2008. – 408 p.
6. Grout I. Digital Systems Design with FPGAs and CPLDs / I. Grout. – Elsevier Science : Amsterdam, The Netherlands, 2011. – 784 p.
7. Baranov S. Logic Synthesis for Control Automata / S. Baranov. – Dordrecht : Kluwer Academic Publishers, 1994. – 312 p.
8. Barkalov A. A. Operational formation of state codes in microprogram automata / A. A. Barkalov, R. M. Babakov // Cybernetics and Systems Analysis. – 2011. – Volume 47 (2). – P. 193–197.
9. Barkalov A. A. Determining the Area of Efficient Application of a Microprogrammed Finite-State Machine with Datapath of Transitions / A. A. Barkalov, R. M. Babakov // Cybernetics and Systems Analysis. – 2019. – Volume 54 (3). – P. 366–375.
10. Barkalov A. A. Algebraic Interpretation of a Microprogram Finite-State Machine with Datapath of Transitions / A. A. Barkalov, R. M. Babakov // Cybernetics and Systems Analysis. – 2016. – Volume 54 (3). – P. 366–375.
11. McElvain K. LGSynth93 Benchmark; Mentor Graphics / K. McElvain. – Wilsonville, OR, USA, 1993.
12. Xilinx. XST UserGuide. V.11.3. Available online: https://www.xilinx.com/support/documentation/sw_manuals/xilinx11/xst.pdf (accessed on 07 June 2022).

DOI 10.15588/1607-3274-2022-3-11

UDC 621.327:681.5

METHODOLOGY OF INCREASING THE RELIABILITY OF VIDEO INFORMATION IN INFOCOMMUNICATION NETWORKS AEROSEGMENT

Karlov D. V. – Dr. Sc., Professor, Head of the Department of Aviation Radio Engineering Navigation and Landing Systems of Ivan Kozhedub Kharkiv National Air Force University, Kharkiv, Ukraine.

Tupitsya I. M. – Lecturer of the Department of Combat Application and Operation of Automated Control Systems of Ivan Kozhedub Kharkiv National Air Force University, Kharkiv, Ukraine.

Parkhomenko M. V. – PhD, Lecturer of the Department of Combat Application and Operation of Automated Control Systems of Ivan Kozhedub Kharkiv National Air Force University, Kharkiv, Ukraine.

ABSTRACT

Context. The problem of localization of the effect of errors in data transmission channels when using compression and noise-immune coding methods in the conditions of compliance with the speed of data delivery in infocommunication systems of the aerosegment. The object of the study is coding methods for increasing the reliability of video information resources in infocommunication networks using airmobile platforms.

Objective. The goal of the work is to methodology development of increasing the reliability of video information in the infocommunication networks of the aerosegment.

Method. The use of noise-immune coding methods to ensure the required level of reliability of video information transmitted in infocommunication systems of the aerosegment has a number of significant disadvantages: it leads to a significant increase in the bit volume of compactly presented video data; the time delay for the delivery of video information is growing, which is critical in the conditions of using airmobile platforms. An increase in time delays in the process of delivering video information leads to the fact that the video information will not be transmitted in full and, as a consequence, in the conditions of aeromonitoring, to the loss of data reliability; time for processing video data increases. The advantage of using compression coding technologies to solve the problem of increasing the reliability of video information transmitted in infocommunication systems of the aerosegment is to reduce the bit volume of the video information resource. However, the existing video processing technologies are based on the use of statistical coding methods and the identification of a series of identical sequences of repeating elements. But the use of such technologies does not provide the required level of error localization. Restructuring method was developed based on identifying patterns in the internal binary structure of message elements by a quantitative attribute. The sign of the number of series of units in the binary structure of message elements is used as a tool for restructuring. Distinctive features of the method are that the restructuring of the information space is carried out without loss of integrity on the basis of structural features by the number of binary series.

Results. The analysis of existing directions for solving the problem of increasing the level of reliability of video information transmitted in the infocommunication systems of the aerosegment was carried out. A method of internal data restructuring has been developed, which allows obtaining the following results: conditions are provided for additional reduction of structural redundancy of code representation of information due to significant reduction of information space capacity as a result of using internal data restructuring on the basis of the number of series of units; conditions are created for localization of errors in the process of reconstruction of video information resources; conditions are created to reduce the time for data processing, due to the fact that the developed method of data restructuring does not require transformations over the elements of the message.

Conclusions. It is necessary to improve the existing compression coding technologies in the direction of identifying patterns, taking into account which will allow localizing the destructive effect of errors arising in the communication channel.

KEYWORDS: video information resource, coding, reliability, efficiency, communication channel, aerosegment, compression technology.

ABBREVIATIONS

CS is crisis situations;
UAV is unmanned aerial vehicles;
VIR is video information resources;
BCH is Bose-Chowdhury-Hawkingham codes;
SU is series of units.

NOMENCLATURE

f_{cl} is a message clustering functionality $U(\theta)$ of the plural $U(\lambda_i)$ on the basis of λ, $\lambda = \overline{\lambda_1, \lambda_{|\Lambda|}}$;

$f_{cl}(u_\xi, \lambda_i)$ is a element clustering functionality u_ξ video sequences $U(\theta)$ with the same amount λ_i SU;

K' is a number distorted pixels that affect the quality of visual perception, %;

$P(\varepsilon)$ is a probability of bit errors under conditions of interference and interference in the communication channel;

$P(u_\xi)$ is a probability of the appearance of elements u_ξ in video sequence $U(\theta)$;

$PSNR_\varepsilon$ is a peak signal-to-noise ratio under conditions of interference and interference in communication channels, dB;

$PSNR_{\varepsilon nec}$ is a necessary value of peak signal-to-noise ratio;

DOI 10.15588/1607-3274-2022-3-12

$PSNR_{\varepsilon_{dm}}$ is a value of peak signal-to-noise ratio for development method;.

$q_{\alpha-1}$, q_{α} are previous and current binary digits, which specify the internal binary structure of the element u_{ξ} message $U(\theta)$;

T_{del} is a delays time, ms;

T_{tr} is a transmission time, s;

$U(\theta)$ is a video sequence;

u_{ξ} is an element of video sequences;

$U(\lambda_i)$ is a set (cluster) of elements u_{ξ}, the binary representation of which has the same value of the sign λ, i.e. $\lambda = \lambda_i$;

$U(\vartheta)$ is an alphabet of video sequences $U(\theta)$;

V_{tr} is a transmission speed, Mbps;

λ is a quantitative sign;

λ_{α} is a value of quantity λ SU on α-th step, $\alpha = \overline{1, |u_{\xi}|_2}$.

INTRODUCTION

Significant rates of digitalization of society are accompanied by increasing the role of information space at both state and regional levels [1–3]. Especially in the field of security, the main objects of which are the population and the system of critical infrastructure of the city [3–5].

In this direction, video services play an important role as a means to improve the efficiency of departmental bodies and line ministries in terms of timely detection, prompt coordinated response to threats to critical infrastructure of the city, and prompt decision-making on their location [6–8]. In turn, the threats that may arise in modern society and the system of critical infrastructure can lead to crises of man-made, natural, criminogenic, terrorist nature [9–11].

It should be borne in mind that the critical infrastructure system is characterized by such properties as complexity, scale and mobility [12]. In this regard, the use of infocommunication systems of the aerosegment segment has become especially important [13–15]. Thus, unmanned aerial vehicles (complexes, drones) are actively used for mobile monitoring of remote objects and areas [16–18]. This allows for timely detection and prompt response to crisis (emergency) situations.

Thus, the use of remote infotainment technologies based on airborne means in order to increase the effectiveness of actions aimed at prevention and timely response to crisis situations (CS) is a very important issue.

Thus, the use of unmanned aerial vehicles provides the following tasks [19–21]:

– obtaining up-to-date information about the object (territory) of observation – video surveillance in real time;

– monitoring of remotely remote objects of observation;

– simultaneous monitoring of a significant number of observation objects;

– monitoring of large areas;

– monitoring of dynamic objects.

For the transmission of video images in the infocommunication systems of the aerosegment, wireless communication channels are used, which are characterized by the presence of a whole range of interference (interference) of natural and artificial nature [22–24]. In this regard, it is necessary to take into account the following problematic aspects of the implementation of wireless technologies [24]:

– low resistance to errors in data transmission channels;

– the need to take into account the electromagnetic compatibility of on-board radio equipment with wireless devices.

This leads to a significant loss of authenticity of video information due to:

– receiving a video information resource with a significant delay;

– loss of relevance of the received video data;

– complete destruction of video information in semantic content;

– Inability to identify image objects.

Thus, the use of unmanned aerial vehicles on the one hand allows increasing the level of informatization of control systems, and on the other is leads to a significant loss of reliability of video information.

The purpose of the work is development of a methodology to increase the reliability of compactly presented video images in infocommunication networks in terms of ensuring the required level of efficiency.

1 PROBLEM STATEMENT

Suppose we have a video sequence $U(\theta)$ consisting of elements u_{ξ} ($\xi = \overline{1, \theta}$) and transmitted over a data channel under error conditions $P(\varepsilon)$. The task is to develop a method of internal data $U(\theta)$ restructuring by identifying patterns in the internal binary structure $[u_{\xi}]_2$ of the elements u_{ξ} of the video sequence $U(\theta)$ on a quantitative sign λ for further clustering.

Restructuring of the information space should provide the required level of reliability of the reconstructed video image, the quantitative assessment of which is the peak signal-to-noise ratio $PSNR_{\varepsilon}$.

For the developed method, the peak signal-to-noise ratio at a given error level ($P(\varepsilon) = 10^{-4}$, $P(\varepsilon) = 10^{-5}$) in the data channel should have the following value:

$$PSNR_{\varepsilon} \geq 15 - 25\,\text{dB}.$$

2 REVIEW OF THE LITERATURE

Analysis of recent scientific publications shows that currently actively used UAVs, which are characterized by

DOI 10.15588/1607-3274-2022-3-12

an increase in maximum flight altitude to 7000 m [25–27]. In this regard, the requirements for the resolution of on-board video surveillance equipment installed on board the UAV are increasing.

To obtain video information using the air segment, airborne means are actively used, which are characterized by the following features [28–30]:

– image resolution – not worse than 4000 × 3000 pixels [28];

– video resolution – not worse than Full HD (1920 × 1080 pixels) [29];

– image format – JPEG / DNG (RAW) [29–30];

– video format – MP4 / MOV (MPEG-4 AVC / H.264, HEVC / H.265) [28–30];

– frame rate per second – 24… 60 [30].

In turn, the increase in the resolution of video images in terms of increasing the height of aerial monitoring leads to the fact that:

– requirements for ensuring the appropriate level of reliability of video information resources (VIR) are increased. This is due to the fact that the number of elements (pixels) that describe the object of observation decreases, resulting in an increase in their semantic load. Accordingly, the loss of the element due to errors in the communication channel leads to a decrease in the level of reliability of the VIR;

– the volume of the video image increases. This leads to an increase in the number of video processing operations. Accordingly, with the same computational complexity and increasing volumes of video information, it is necessary to look for opportunities to reduce the complexity of the processing of VIR to ensure the desired level of efficiency of delivery to the final destination.

So increase the reliability of compactly presented video images in infocommunication networks is **an urgent scientific and applied task**.

3 MATERIALS AND METHODS

Let's break feature space into rectangular regions limiting the range of values of each feature by its minimum and maximum values. Then the partition projections into feature axis allow allocating feature intervals for each of the rectangular block. The intervals can be formed as cluster projections or as a regular grid, or on the basis of class boundaries in sample one-dimensional projections on the feature axes [24].

The use of the aerosegment involves the implementation of a number of requirements for VIR from the standpoint of ensuring the required level of reliability, which are shown in Table 1. The requirements specified in Table 1 are formed taking into account:

– analysis of QoS (quality of service) indicators [31–33];

– requirements for providing video information in critical infrastructure systems [34];

– results of practical research [35–37].

The scheme of formation of these requirements, taking into account the peculiarities of the use of wireless communication channels for the delivery of video information is shown in Fig. 1.

Quantitative indicators of these requirements are given in Table 1.

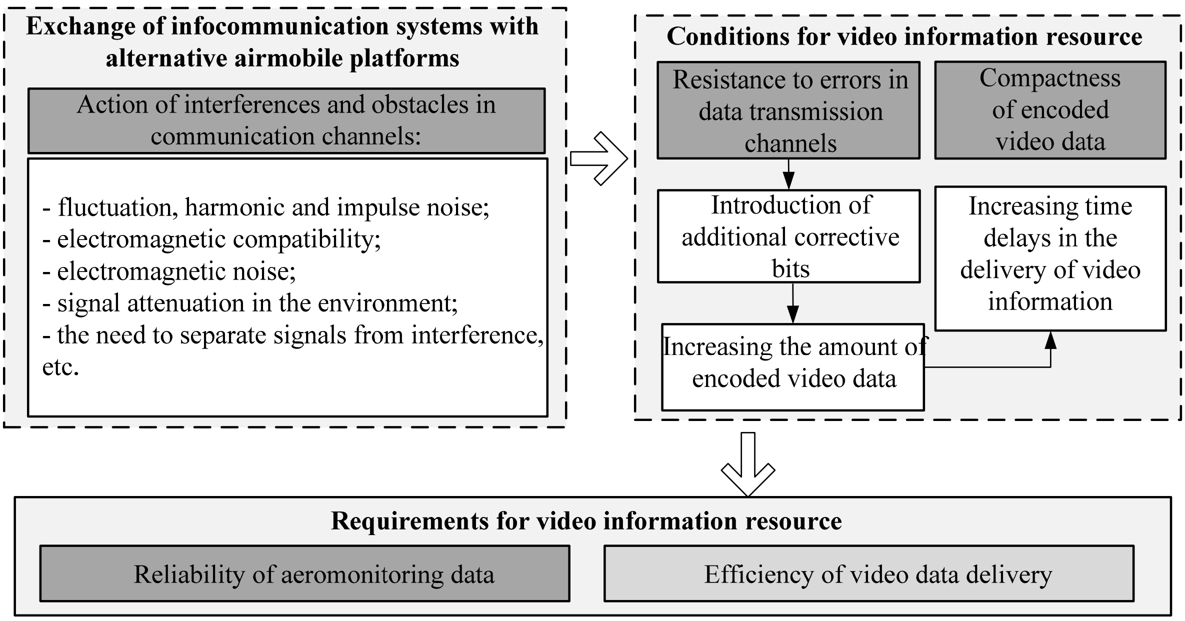


Figure 1 – The scheme of formation of requirements to video information resources in the conditions of use aerosegment

DOI 10.15588/1607-3274-2022-3-12

Table 1 – Requirements for video information resources in the conditions of using the air segment

№ s/n	Indicator	Necessary value
1.	Peak signal-to-noise ratio $PSNR_{\varepsilon}$ under conditions of interference and interference in communication channels, dB	> 15–25
2.	Number K' distorted pixels that affect the quality of visual perception, %	7–8
3.	Probability of bit errors under conditions of interference and interference in the communication channel, $P(\varepsilon)$	$10^{-5} \ldots 10^{-4}$
4.	Time delays T_{del}, ms	120

Analysis of the data listed in table 1 show that the main requirements for VIR in the use of the aerosegment are the following:

1. Ensuring the reliability of the video information resource in the conditions of errors in the communication channel, which is determined by the following quantitative indicators:

– peak signal-to-noise ratio $PSNR_{\varepsilon}$ under conditions of interference and interference in communication channels, which should not be less than 15–25 dB;

– number K' distorted pixels in the reconstructed video image that affect the quality of visual perception should not exceed 7–8%;

2. Probability of bit errors under conditions of interference and interference in the communication channel, $P(\varepsilon)$.

3. The required level of time delay T_{del} in the process of data delivery of video information resource using wireless communication technologies. Time delays T_{del} should not exceed 120 ms. Otherwise, in the conditions of air monitoring, significant time delays lead to the fact that video information will not be transmitted in full. So this is what will lead to the loss of its credibility.

On the other hand, infocommunication systems using airmobile platforms are characterized by the presence of interference and interference in communication channels due to the following factors [38–40]: electromagnetic compatibility of airborne radio navigation equipment with wireless devices; low resistance of wireless communication technologies to errors; electromagnetic noise; the need to separate signals from interference; mutual electromagnetic interference of devices of one channel (or adjacent frequencies); signal attenuation in the environment.

These factors pose a threat to the loss of information, the destruction of video images, reducing the efficiency of their delivery due to the need to re-record and transmit video data.

There are the following areas of solving the problem of increasing the reliability of video information transmitted in infocommunication networks using airmobile platforms, namely [41–44]:

1) using existing noise-tolerant coding technologies;

2) compression coding technologies with localization of errors that occur in the communication channel.

Estimation of efficiency of use of existing technologies of noise-tolerant coding of video data. One of the ways to solve the problem increase the reliability of video information is the use of existing noise-tolerant coding technologies. The most common are the Bose-Chowdhury-Hawkingham (BCH) codes and the Reed-Solomon codes. The essence of the use of these noise-tolerant coding technologies is the need to introduce additional corrective (verification) bits in the process of forming code structures assigned to the data VIR.

Fig. 2–3 shows estimates of video transmission time in a given level of errors ($P(\varepsilon) = 10^{-4} \ldots 10^{-5}$) at a given baud rate $V_{tr} = 5$ $Mbps$, $V_{tr} = 40$ $Mbps$) and pixel depth (8 bits) using standard noise-tolerant encoding methods.

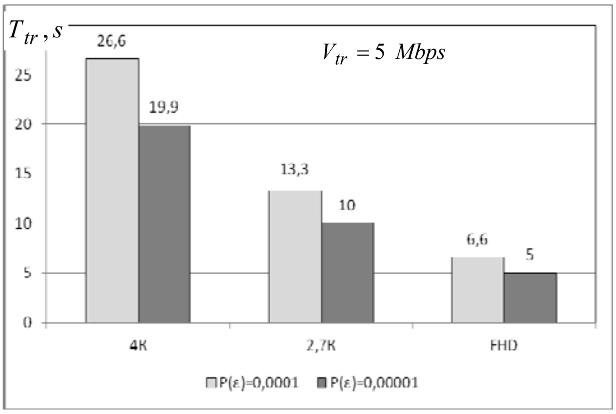


Figure 2 – Diagram of the dependence of the transmission time of the video frame on the resolution at a given transmission speed, $V_{tr} = 5$ Mbps

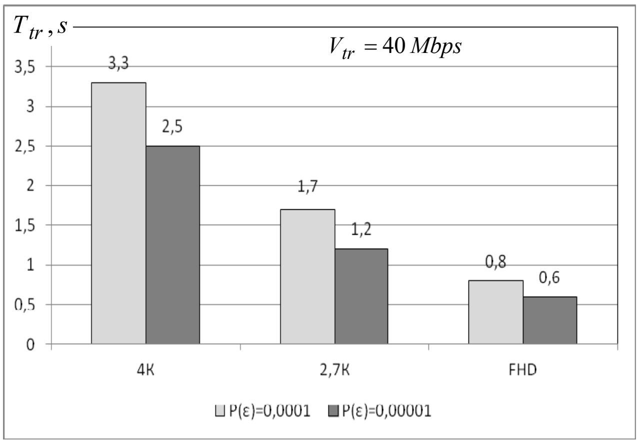


Figure 3 – Diagram of the dependence of the transmission time of the video frame on the resolution at a given transmission speed, $V_{tr} = 40$ Mbps

Analysis of the data shown in Fig. 2–3 shows that:

1. Given the average recording frequency of 30 frames per second, it can be concluded that the on-board equipment for video surveillance and video transmission on the studied airborne means does not provide full registration and transmission of all video information generated per second.

DOI 10.15588/1607-3274-2022-3-12

2. The minimum time for transmitting video information from the UAV is:

– at a video transfer rate of 5 Mbps and error rates in the data channel $P(\varepsilon)=10^{-4}$ – from 3 to 40 minutes;

– at a video data transfer rate of 40 Mbps and a given level of errors in the data channel $P(\varepsilon)=10^{-4}$ – from 24 s to 5 min;

– at a video transfer rate of 5 Mbps and a given level of errors in the data channel $P(\varepsilon)=10^{-5}$ – from 2.5 to 30 minutes;

– at a video data transfer rate of 40 Mbps and a given level of errors in the data channel $P(\varepsilon)=10^{-5}$ – from 18 s to 3.5 minutes

3. Increasing the pixel depth to 24 bits increases the transmission time of video information by 3 times.

Increasing time delays in the process of delivering video information leads to the fact that the video information will not be transmitted in full and, as a consequence, in the conditions of aeromonitoring to the loss of reliability of VIR data.

Thus, we can conclude that the use of existing noise-tolerant coding technologies can increase the resilience of VIR to errors in data channels due to the use of additional corrective digits, but has a number of significant disadvantages:

– leads to a significant increase in the bit size of video data;

– there is a growing delay in the delivery of video information, which is critical in the use of airmobile platforms;

– video processing time is increasing.

In the conditions of aerial monitoring, significant time delays lead to the fact that video information will not be transmitted in full. So this is what will lead to the loss of its credibility.

On the other hand, it is possible to localize the effect of errors on the quality of recoverable images due to the existing psycho visual redundancy. But it is on the elimination of psycho visual redundancy in the existing methods of compression coding is achieved the basic value of the level of compression of video images.

Therefore, it is necessary to increase the reliability of the video information resource based on the study of the possibilities of reducing the negative impact of errors in the process of compression and reconstruction of video frames.

Estimation of efficiency of use of existing technologies of compression coding of video data from a position of maintenance of necessary level of reliability. Analysis of the effectiveness of existing compression video coding technologies from the standpoint of ensuring the required level of reliability in the event of errors in the reconstruction process shows that existing video coding algorithms, built on processing technologies implemented on JPEG platform, have significant shortcomings. This is due to the fact that the action of errors for existing approaches has an avalanche effect, which leads to the destruction of video images in the reconstruction process.

Estimation of reliability of video images in infocommunication systems of an aerosegment on an indicator of a peak ratio signal / noise $PSNR_{\varepsilon}$ at a given error rate ($P(\varepsilon)=10^{-4}... 10^{-5}$) in data transmission channels is shown in Fig. 4.

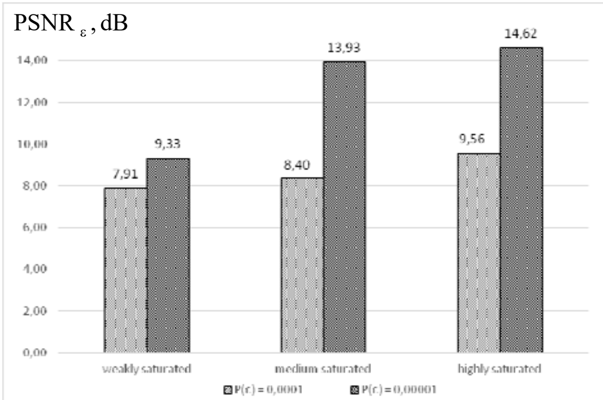


Figure 4 – Diagram of the dependence of the peak signal-to-noise ratio $PSNR_{\varepsilon}$ from mistakes $P(\varepsilon)$ in the communication channel

Analysis of the data shown in Fig. 4 shows that the use of existing compression coding technologies does not provide an appropriate level of reliability of video information for infocommunication systems of the air segment. This is due to the fact that the value of the peak signal-to-noise ratio $PSNR_{\varepsilon}$ at a given level of errors in the process of reconstruction of the video information resource does not meet the required $PSNR_{\varepsilon nec}$, i.e. the following condition is not met:

$$PSNR_{\varepsilon} \le PSNR_{\varepsilon nec},\ PSNR_{\varepsilon} \le 15-25\,\text{dB}.$$

So at the level of errors $P(\varepsilon)$ in the communication channel equal to $P(\varepsilon)=10^{-5}$ the value of the peak signal-to-noise ratio $PSNR_{\varepsilon}$ less than the minimum required value of 3% for high-saturated video to 38% for low-saturated video.

In turn, at the level of errors $P(\varepsilon)$ in the communication channel equal to $P(\varepsilon)=10^{-4}$ the value of the peak signal-to-noise ratio $PSNR_{\varepsilon}$ less than the minimum required value of 36% for high-saturated video to 47% for low-saturated video.

Hence the use of existing compression coding technologies leads to a significant loss of reliability of information, namely the video resource can be obtained with a significant delay and will be in this case irrelevant, or completely destroyed in semantic content, ie not subject to reconstruction. This is due to the fact that existing compression coding technologies are based on the use of sta-

DOI 10.15588/1607-3274-2022-3-12

tistical coding methods and the detection of a series of identical sequences of repeating elements. This situation leads to a number of significant shortcomings, namely:

1. Low resistance to errors in data transmission channels, as statistical coding methods are based on non-uniform code constructions.

2. Due to the formation of non-uniform structural characteristics (lengths of series of transformant components with zero values) to reduce redundancy in transformants, the occurrence of errors in the reconstruction process can lead to a shift in the data of the video information resource.

Therefore, this direction does not provide the required level of localization of errors. Therefore, it is necessary to improve existing compression coding technologies in the direction of identifying patterns, taking into account which will localize the destructive effects of errors that occur in the communication channel.

To solve the problems associated with improving the reliability of VIR data while ensuring the required level of efficiency in the use of compression coding technologies, it is necessary to ensure the following conditions:

1) to ensure the localization of errors to reduce their impact on destructive action in the process of reconstruction of video data;

2) due to the restructuring of the information space, within which unlikely elements will be coded u_ξ ($P(u_\xi) \to 0$), increase the value of probabilities $P(u_\xi)$ by dividing the power of the original alphabet and the length of the video sequence into local sets.

To do this, it is proposed to use a fundamentally new approach – internal restructuring, which takes into account the laws of internal binary structure of the element u_ξ ($u_\xi \in U(\theta)$) on a quantitative sign λ.

Structural and functional scheme of clustering of clustering of elements u_ξ message $U(\theta)$ on a quantitative basis λ presented in Fig. 5.

Video sequence clustering $U(\theta)$ on a quantitative sign λ of the set $U(\lambda_i)$ is given by the following expression:

$$U(\theta) \xrightarrow{f_{cl}} \{U(\lambda_1), ..., U(\lambda_i), ..., U(\lambda_{|\Lambda|})\}. \quad (1)$$

For internal data restructuring as quantitative feature is proposed to use quantity λ series of units (SU) in the binary representation of elements. The advantages of using this quantitative feature are the simplicity of algorithmic implementation (the process of forming a quantitative feature involves the use of only arithmetic and logical operations).

Quantity formation cycle λ SU, in the binary representation of the element u_ξ, is given by the following system of expressions:

$$\lambda_\alpha = \begin{cases} \lambda_\alpha, \to q_{\alpha-1} = q_\alpha; \\ \lambda_\alpha + 1 \ \to q_{\alpha-1} \neq q_\alpha. \end{cases} \quad (2)$$

Clustering of elements u_ξ video sequences $U(\theta)$ with the same amount of SU is given by the following expression:

$$f_{cl}(u_\xi, \lambda_i): U(\theta) \xrightarrow{f_{cl}} \{U(\lambda_1); ...; U(\lambda_i); ...; U(\lambda_{|\Lambda|})\}, \quad (3)$$

Alphabet clustering $U(\vartheta)$ video sequences $U(\theta)$ based on the amount of SU in the binary representation of the elements u_ξ is given by the following expression:

$$f_{cl}(u_\xi, \lambda_i): U(\vartheta) \xrightarrow{f_{cl}} \{U(\lambda_1); ...; U(\lambda_i); ...; U(\lambda_{|\Lambda|})\}. \quad (4)$$

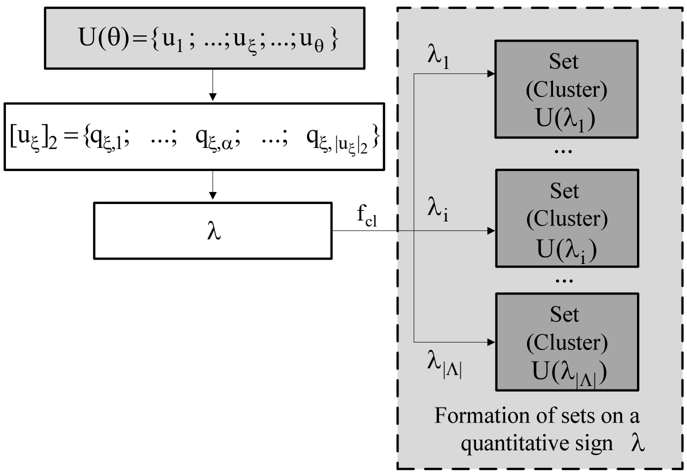


Figure 5 – Structural and functional scheme of clustering of elements u_ξ video sequences $U(\theta)$ on a quantitative basis λ of the set $U(\lambda_i)$

DOI 10.15588/1607-3274-2022-3-12

4 EXPERIMENTS

To assess the effectiveness of the developed method of internal restructuring of video information resource data, a number of experimental studies were conducted.

Test video images of different degrees of saturation were used as source data: low-, medium- and high-saturation.

A discrete symmetric channel without memory was used as the data channel.

The following error probability values were used to assess the effect of errors on the reliability of the reconstructed video image: $P(\varepsilon)=10^{-4}$, $P(\varepsilon)=10^{-5}$.

Modeling of experimental research results was performed using the developed software product.

Comparative assessment of the reliability of video information in infocommunication systems of the aerosegment by the indicator of the peak signal-to-noise ratio $PSNR_{\varepsilon_{dm}}$ at a given error rate ($P(\varepsilon)=10^{-4}... 10^{-5}$) in data transmission channels with algorithms of the JPEG family is shown in Fig. 6–7.

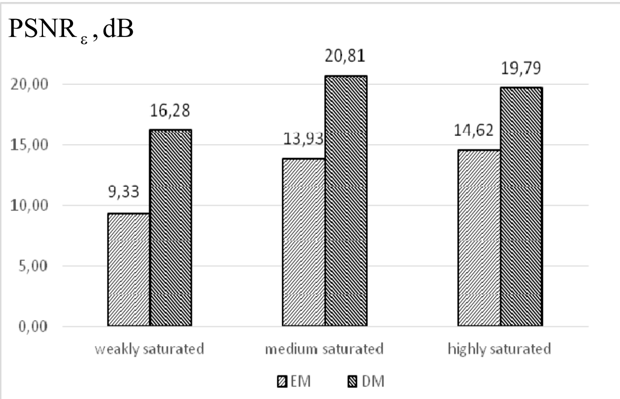


Figure 6 – Diagram of the dependence of the peak signal-to-noise ratio at a given error level $P(\varepsilon)$ from the degree of image saturation for developed and existing methods, $P(\varepsilon)=10^{-5}$

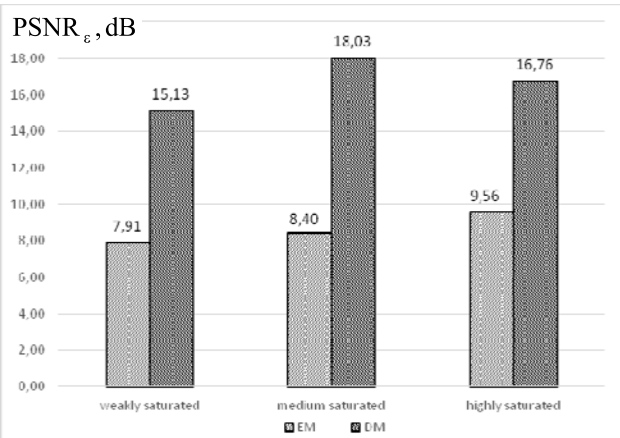


Figure 7 – Diagram of the dependence of the peak signal-to-noise ratio at a given error level $P(\varepsilon)$ from the degree of image saturation for developed and existing methods, $P(\varepsilon)=10^{-4}$

5 RESULTS

Analysis of the results of the reconstruction of VIR data at a given level of errors ($P(\varepsilon)=10^{-4}... 10^{-5}$) by the value of the peak signal / noise ratio is presented in Fig. 6–7 indicates that:

1. The use of the developed coding method allows to increase the level of reliability of VIR data at a given level of errors in the reconstruction process. So for the case when the error in the discrete symmetric data channel without memory is set to a value equal to $P(\varepsilon)=10^{-5}$ the use of the developed coding method allows to increase the reliability of video information, which is determined by the quantitative assessment of the peak signal-to-noise ratio, by an average of 53.08% compared to existing methods.

Accordingly, for the second case under study (when $P(\varepsilon)=10^{-4}$) the developed method allows to increase the reliability of video information by an average of 93.8%.

2. The developed method of compression coding of video information solves the scientific problem of increasing the reliability of video information in infocommunication systems of the aerosegment. This means that the following condition is met:

$$PSNR_{\varepsilon} \geq 15-25\,\text{dB}.$$

6 DISCUSSION

Analysis of the effectiveness of modern video coding technologies shows that existing approaches do not provide the required level of reliability in terms of ensuring prompt delivery. Thus, experimental studies of the impact of errors in the data transmission channel show that the use of noise-tolerant coding technologies leads to a significant increase in the amount of coded data. In turn, this leads to time delays in the delivery of video data (Fig. 2–3).

The use of existing compression coding technologies based on the JPEG platform allows more compact presentation of coded data [32, 33]. This allows ensuring compliance with the requirements for the efficiency of delivery of video information resources by eliminating psychovisual redundancy [11, 13]. However, it should be borne in mind that at the coding stage, the algorithms of this family use a statistical approach [1–4, 6]. This means that the result of statistical coding is the formation of non-uniform codes assigned to the elements of the video sequence [7–9]. In turn, the results of experimental studies presented in Fig. 4, indicate that the use of this approach in the process of forming code structures does not allow localizing the effect of errors in communication channels.

The use of the proposed method (Fig. 5), which is based on the restructuring of the information space in quantitative terms, allows without loss of integrity to provide additional reduction of the structural redundancy of the code representation of the video sequence. This is due to the fact that further coding of video sequence elements occurs in the statistical space of clusters. In turn, this allows to increase the level of reliability of video data due to the localization of errors within the sequence of code constructs assigned to one cluster (Fig. 6–7). Comparative

DOI 10.15588/1607-3274-2022-3-12

analysis of the developed method with the existing ones shows that the proposed approach allows providing the required level of reliability (Fig. 6–7).

CONCLUSIONS

1. For infocommunication systems using airmobile platforms are characterized by the presence of interference and interference in communication channels. This poses a threat to the loss of information, the destruction of video images, reducing the efficiency of their delivery due to the need to re-capture and transmit video data. In this connection the problem of increase needs to be solved reliability of video information transmitted in infocommunication networks using airmobile platforms. There are the following ways to solve this problem: use existing noise-tolerant coding technologies; compression coding technologies with localization of errors that occur in the communication channel.

2. A study of the use of noise-tolerant coding methods to ensure the required level of reliability of video information transmitted in infocommunication systems of the air segment shows that this direction increases the resistance of VIR to errors in data channels through the use of additional correction bits. But it has a number of significant disadvantages:

– leads to a significant increase in the bit size of compactly presented video data;

– there is a growing delay in the delivery of video information, which is critical in the use of airmobile platforms. Increasing time delays in the process of delivery of video information leads to the fact that video information will not be transmitted in full and, as a consequence, in the conditions of aerial monitoring to the loss of reliability of VIR data;

– video processing time is increasing.

3. The advantage of using compression coding technologies to solve the problem of increasing the reliability of video information transmitted in the infocommunication systems of the air segment is to reduce the bit size of the video information resource. However, existing video processing technologies are based on the use of statistical coding methods and the detection of a series of identical sequences of repeating elements. But the use of such technologies leads to a number of significant disadvantages, namely:

– low resistance to errors in data transmission channels, as statistical coding methods are based on non-uniform code constructions. This is due to the fact that the action of errors that occur in the communication channel leads to an avalanche effect;

– due to the formation of uneven structural characteristics (lengths of series of transformant components with zero values), to reduce redundancy in transformants, the occurrence of errors in the reconstruction process can lead to a shift in the data of the video information resource.

Therefore, this direction does not provide the required level of localization of errors.

DOI 10.15588/1607-3274-2022-3-12

4. It is substantiated that it is necessary to improve the existing compression coding technologies in the direction of identifying patterns, taking into account which will allow localizing the destructive effect of errors that occur in the communication channel.

5. Thus, for the first time a method of restructuring based on the detection of patterns in the internal binary structure of the message elements was developed. Distinctive features of the method are that the restructuring of the information space is carried out without loss of integrity on the basis of structural features by the number of binary series. This allows you to get the following results:

1) conditions are provided for additional reduction of structural redundancy of code representation of information due to significant reduction of information space capacity as a result of using internal data restructuring on the basis of the number of series of units.

2) conditions are created for localization of errors in the process of reconstruction of video information resources;

3) conditions are created to reduce the time for data processing, due to the fact that the developed method of data restructuring does not require transformations over the elements of the message.

ACKNOWLEDGEMENTS

The work was supported by Kharkiv National Air Force University named after Ivan Kozhedub, Kharkiv, Ukraine.

REFERENCES

1. Miano J. Compressed image file formats: JPEG, PNG, GIF, XBM, BMP/ by John Miano, 1999, 264 p.
2. Pratt W. K., Chen, W. H., Welch L. R. Slant transforms image coding, *Proc. Computer Processing in communications,* 1969, pp. 63–84.
3. Wallace G. K. The JPEG Still Picture Compression Standard, *Communication in ACM,* 1991, Vol. 34, No. 4, pp. 31–34.
4. Wallace G. K. Overview of the JPEG (ISO/CCITT) Still image compression: image processing algorithms and techniques, *Proc. of SPIE-IS&T Electronic Imaging (SPIE),* 1990, Vol. 1244, pp. 220–233.
5. Wang S., Zhang X., Liu X., Zhang J., Ma S., Gao W. Utility Driven Adaptive Preprocessing for Screen Content Video Compression, *IEEE Transactions on Multimedia*, 2017, Vol. 19, No. 3, pp. 660–667.
6. Gonzales R. C., Woods R. E. Digital image processing. Prentice Inc. Upper Saddle River, 2002, 779 p.
7. Dong W., Wang J. JPEG Compression Forensics against Resizing, *IEEE Trustcom/ BigDataSE/IѳSPA.* Tianjin, China, 2016, pp. 1001–1007. DOI: 10.1109/TrustCom.2016.0168.
8. Richter T. Error Bounds for HDR Image Coding with JPEG XT, *Data Compression Conference (DCC),* 2017, pp. 122–130. DOI: 10.1109/DCC.2017.7.
9. Xiao W., Wan N. A., Hong and Chen X. A Fast JPEG Image Compression Algorithm Based on DCT, *IEEE International Conference on Smart Cloud (SmartCloud),* 2020, pp. 106–110. DOI: 10.1109/ SmartCloud49737. 2020.00028.

10. Rippel O. Learned Video Compression, *IEEE/CVF International Conference on Computer Vision (ICCV),* 2019, pp. 3453–3462. DOI: 10.1109/ICCV. 2019.00355.
11. Bienik J., Uhrina M., Kuba M. and Vaculik M. Performance of H.264, H.265, VP8 and VP9 Compression Standards for High Resolutions, *19th International Conference on Network-Based Information Systems (NBiS),* 2016, pp. 246–252. DOI: 10.1109/NBiS. 2016.70.
12. Wang X., Xiao J., Hu R., Wang Z. Cruise UAV Video Compression Based on Long-Term Wide-Range Background, *Data Compression Conference (DCC),* 2017, pp. 466–467. DOI: 10.1109/DCC.2017.71.
13. Minallah N., Gul S., Bokhari M. Performance Analysis of H.265/HEVC (High-Efficiency Video Coding) with Reference to Other Codecs, *13th International Conference on Frontiers of Information Technology (FIT),* 2015, pp. 216–221. DOI: 10.1109/ FIT.2015.46.
14. Djelouah A., Campos J., Schaub-Meyer S., Schroers C. Neural Inter-Frame Compression for Video Coding, *IEEE/CVF International Conference on Computer Vision (ICCV),* 2019, pp. 6420–6428. DOI: 10.1109/ICCV.2019. 00652.
15. Tupitsya I. Methodology for restructuring information resource data to improve the efficiency of statistical coding, *Science-based technologies,* 2019, Vol. 42, No. 2, pp. 262 – 269 DOI: 10.18372/2310-5461.42.13801.
16. Narmatha C., Manimegalai P., Manimurugan S. A LS-compression scheme for grayscale images using pixel based technique, *International Conference on Innovations in Green Energy and Healthcare Technologies (IGEHT),* 2017, pp. 1–5, DOI: 10.1109/ IGEHT.2017.8093980.
17. Alam M. A., Faster Image Compression Technique Based on LZW Algorithm Using GPU Parallel Processing, *Joint 7th International Conference on Informatics, Electronics & Vision (ICIEV) and 2nd International Conference on Imaging, Vision & Pattern Recognition (icIVPR),* 2018, pp. 272–275, DOI: 10.1109/ICIEV.2018.8640956.
18. Barannik V., Tupitsya I., Barannik V., Shulgin S., Musienko A., Kochan, R., Veselska O. The Application of the Internal Restructuring Method of the Information Resource Data According to the Sign of the Number of Series of Units to Improve the Statistical Coding Efficiency, *10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS),* 2019, pp. 65–69. DOI: 10.1109/IDAACS.2019.8924460.
19. Barannik V., Tupytsya I., Parkhomenko M., Azatov A., Pershin A., Gurzhii P., Shaikhanova A., Karpiinski M. The concept of a quantitative sign formation for the internal restructuring of information resource data. Przetwarzanie, transmisja i bezpieczeństwo informacji' 2020, Monogrpah, 2020, pp.41–52.
20. Poolakkachalil T. K., Chandran S., Muralidharan R., Vijayalakshmi K. Comparative analysis of lossless compression techniques in efficient DCT-based image compression system based on Laplacian Transparent Composite Model and An Innovative Lossless Compression Method for Discrete-Color Images, *3rd MEC International Conference on Big Data and Smart City (ICBDSC),* 2016, pp. 1–6, DOI: 10.1109/ICBDSC.2016.7460360.
21. Wang Z., Liao R., Ye Y. Joint Learned and Traditional Video Compression for P Frame, *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW),* 2020, pp. 560–564. DOI: 10.1109/CVPRW50498.2020.00075.
22. Bui V., Chang L., Li D., Hsu L., Chen M. Comparison of lossless video and image compression codecs for medical computed tomography datasets, *IEEE International Conference on Big Data (Big Data),* 2016 pp. 3960–3962. DOI: 10.1109/BigData. 2016.7841075.
23. Akbari M., Liang J., Han J., Tu C. Learned Variable-Rate Image Compression With Residual Divisive Normalization, *IEEE International Conference on Multimedia and Expo (ICME),* 2020, pp. 1–6. DOI: 10.1109/ICME46284.2020.9102877.
24. Shinde T. Efficient Image Set Compression, *IEEE International Conference on Image Processing (ICIP),* 2019, pp. 3016–3017. DOI: 10.1109/ICIP. 2019.8803230.
25. Barannik V., Sidchenko S., Tarnopolov R., Tupitsya I. The process of forming layers of bit zones in the method of crypto-semantic presentation of images on the basis of the floating scheme, *1st International conference on advanced information and communication technologies (AICT): Conference Proceedings,* 2015, pp. 59–62.
26. Barannik V., Tupitsya I., Sidchenko S., Tarnopolov R. The Method of Crypto-Semantic Presentation of Images Based on the Floating Scheme in the Basis of the Upper Boundaries, *IEEE Problems of Infocommunications. Science and Technology (PICS&T): proceedings of International Scientific-Practical Conference,* 2015, pp. 248–251. DOI: 10.1109/INFOCOMMST.2015.7357326.
27. Lin J., Liu D., Li H., Wu F. M-LVC: Multiple Frames Prediction for Learned Video Compression. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR),* 2020, pp. 3543–3551. DOI: 10.1109/CVPR42600. 2020.00360.
28. Artuğer F., Özkaynak F. Fractal Image Compression Method for Lossy Data Compression, *International Conference on Artificial Intelligence and Data Processing (IDAP),* 2018, pp. 1–6. DOI: 10.1109/IDAP. 2018.8620735.
29. Arnob P., Tanvir Z.; Prajoy P., Rafi A., Muktadir Rahman, M., Mamdudul Haque, Kh., Iris image compression using wavelets transform coding. *2nd International Conference on Signal Processing and Integrated Networks (SPIN)*, 2015, pp. 544–548, DOI: 10.1109/SPIN.2015.7095407.
30. Zhu X., Liu L., Jin Na, Ai P. Morphological component decomposition combined with compressed sensing for image compression, *IEEE International Conference on Information and Automation (ICIA)*, DOI: 10.1109/ICInfA.2016.7832096.
31. Wang S., Kim S. M., Yin Z., & He T. Encode when necessary: Correlated network coding under unreliable wireless links, *ACM Transactions on Sensor Networks,* 2017, Vol. 13(1). DOI: 10.1145/ 3023953.
32. Phatak A. A Non-format Compliant Scalable RSA-based JPEG Encryption Algorithm, *International Journal of Image. Graphics and Signal Processing,* 2016, Vol. 8, No. 6, pp. 64–71. DOI: 10.5815/ijigsp. 2016.06.08.
33. Wu H., Sun X., Yang J., Zeng W., Wu F. Lossless Compression of JPEG Coded Photo Collections, *IEEE Transactions on Image Processing,* 2016, Vol. 25, No. 6, pp. 2684–2696. DOI: 10.1109/ TIP.2016.2551366.
34. Lee J., Cho S., Beack S.-K. Context-adaptive entropy model for end-to-end optimized image compression, 2018. arXiv: 1809.10452.
35. Chen C., Zhuo Y. A research on anti-jamming method based on compressive sensing for OFDM analogous system, *IEEE 17th International Conference on Communication Technology (ICCT),* 2017, pp. 655–659, DOI: 10.1109/ICCT.2017. 8359718.

DOI 10.15588/1607-3274-2022-3-12

36. Wang S., Kim S., Yin Z., He T. Encode when necessary: Correlated network coding under unreliable wireless links, *ACM Transactions on Sensor Networks,* 2017, Vol. 13, No.1, pp. 24 – 29, DOI: 10.1145/ 3023953.
37. Han S., Mao H., Dally W. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding, 2015. arXiv: 1510.00149.
38. Zhurakovskyi B., Boiko J., Druzhynin V., Zeniv I., & Eromenko O. Increasing the efficiency of information transmission in communication channels, *Indonesian Journal of Electrical Engineering and Computer Science,* 2020, Vol. 19(3), pp. 1306–1315. DOI: 10.11591/ijeecs.v19.i3.
39. Barannik, V., Sidchenko, S., Tupitsya, I., Stasev, S. The application for internal restructuring the data in the entropy coding process to enhance the information resource security. IEEE East-West Design and Test Symposium (EWDTS), 2016, pp. 1–4. DOI:10.1109/EWDTS.2016.7807749.
40. Barannik V., Tupitsya I., Gurzhii I., Barannik V., Sidchenko S., Kulitsa O. Two-Hierarchical Scheme of Statistical Coding of Information Resource Data with Quantitative Clustering, *IEEE International Conference on Advanced Trends in Information Theory (ATIT),* 2019, pp. 89–92. DOI: 10.1109/ATIT49449.2019.9030451.
41. Barannik V., Tupitsya I., Dodukh O., Barannik V., Parkhomenko M. The Method of Clustering Information Resource Data on the Sign of the Number of Series of Units as a Tool to improve the Statistical Coding Efficiency, *IEEE 15th International Conference on the Experience of Designing and Application of CAD Systems (CADSM),* 2019, pp. 32–35. DOI: 10.1109/CADSM.2019.8779243.
42. Barannik V., Tupitsya I., Kovalenko O., Sidchenko Y., Yroshenko V., Stepanko O. The analysis of the internal restructuring method efficiency used for a more compact representation of the encoded data, *Advanced Trends in Information Theory (ATIT'2020): proceedings of the Intern. Conf.,* 2020, pp. 48–51. DOI: 10.1109/ATIT50783.2020.9349307.
43. Yudin O., Artemov V., Krasnorutsky A., Barannik V. Tupitsya I. and Pris G. Creating a mathematical model for estimating the impact of errors in the process of reconstruction of non-uniform code structures on the quality of recoverable video images, *Advanced Trends in Information Theory (ATIT'2021): proceedings of the Intern. Conf.,* pp. 38–41. DOI: 10.1109/ATIT54053.2021.9678887.
44. Khmelevskiy S., Tupitsya I., Mahdi Q. A., Musienko O., Parkhomenko M., & Borovensky Y. Development of the external restructuring method to increase the efficiency of information resource data encoding, *Information Processing Systems,* 2021, No. 3(166), pp. 52–61. https://doi.org/10.30748/soi.2021.166.06.

Received 20.04.2022.
Accepted 26.08.2022.

УДК 621.327:681.5

МЕТОДОЛОГІЯ ПІДВИЩЕННЯ ДОСТОВІРНОСТІ ВІДЕОІНФОРМАЦІЇ В ІНФОКОМУНІКАЦІЙНИХ МЕРЕЖАХ АЕРОСЕГМЕНТУ

Карлов Д. В. – д-р техн. наук, професор, начальник кафедри авіаційних радіотехнічних систем навігації та посадки Харківського національного університету Повітряних Сил ім. І. Кожедуба, Харків, Україна.

Тупиця І. М. – викладач кафедри бойового застосування та експлуатації автоматизованих систем управління Харківського національного університету Повітряних Сил ім. І. Кожедуба, Харків, Україна.

Пархоменко М. В. – канд. техн. наук, старший викладач кафедри бойового застосування та експлуатації автоматизованих систем управління Харківського національного університету Повітряних Сил ім. І. Кожедуба, Харків, Україна.

АНОТАЦІЯ

Актуальність. Проблема локалізації впливу помилок у каналах передачі даних при використанні методів компресійного та завадостійкого кодування в умовах дотримання швидкості доставки даних в інфокомунікаційних системах аеросегменту. Об'єктом дослідження є методи кодування для підвищення надійності відеоінформаційних ресурсів в інфокомунікаційних мережах з використанням аеромобільних платформ.

Метод. Використання методів завадостійкого кодування для забезпечення необхідного рівня достовірності відеоінформації, що передається в інфокомунікаційних системах аеросегменту, має ряд істотних недоліків – призводить до значного збільшення бітового об'єму компактно представлених відеозображень; зростає часова затримка на доставку відеоінформації, що в умовах використання аеромобільних платформ є критичним. Підвищення часових затримок у процесі доставки відеоінформації призводить до того, що: відеоінформацію буде передано не в повному обсязі і, як наслідок, в умовах аеромоніторингу до втрати достовірності даних; збільшується час на обробку відеоданих. Перевагою використання технологій компресійного кодування для вирішення задачі підвищення достовірності відеозображень, що передаються в інфокомунікаційних системах аеросегменту, є зниження бітового об'єму відеоінформаційного ресурсу. Однак існуючі технології обробки відеоданих базуються на використанні методів статистичного кодування і виявленні серій однакових послідовностей елементів, що повторюються. Але використання таких технологій не забезпечує необхідного рівня локалізації дії помилок. Розроблений метод реструктуризації на основі виявлення закономірностей у внутрішній двійковій структурі елементів повідомлення за кількісною ознакою. Інструментом для реструктуризації є ознака кількості серій одиниць у двійковій структурі елементів повідомлення. Відмінні характеристики методу полягають у тому, що реструктуризація інформаційного простору здійснюється без втрати цілісності на основі структурної ознаки за кількістю двійкових серій.

Результати. Проведено аналіз існуючих напрямків вирішення проблеми підвищення рівня достовірності відеоінформації, що передається в інфокомунікаційних системах аеросегменту. Розроблений метод внутрішньої реструктуризації даних дозволяє отримати наступні результати: забезпечуються умови додаткового скорочення структурної надмірності кодового представлення інформації зарахунок суттєвого скорочення потужності інформаційного простору в результаті використання внутрішньої реструктуризації даних за ознакою кількості серій одиниць; створюються умови для локалізації дії помилок в процесі реконструкції відеоінформаційних ресурсів; створюються умови для скорочення часу на

DOI 10.15588/1607-3274-2022-3-12

обробку даних, в зв'язку з тим, що розроблений метод реструктуризації даних не вимагає проведення перетворень над елементами повідомлення.

Висновки. Необхідно вдосконалювати існуючі технології компресійного кодування у напрямі виявлення закономірностей, облік яких дозволить локалізувати руйнівну дію помилок, що виникають у каналі зв'язку.

КЛЮЧОВІ СЛОВА: відеоінформаційний ресурс, кодування, достовірність, оперативність, канал зв'язку, аеросегмент, технології стиснення.

УДК 621.327:681.5

МЕТОДОЛОГИЯ ПОВЫШЕНИЯ ДОСТОВЕРНОСТИ ВИДЕОИНФОРМАЦИИ В ИНФОКОМУНИКАЦИОННЫХ СЕТЯХ АЭРОСЕГМЕНТА

Карлов Д. В. – д-р техн. наук, профессор, начальник кафедры авиационных радиотехнических систем навигации и посадки Харьковского национального университета Воздушных Сил им. И. Кожедуба, Харьков, Украина.

Тупица И. М. – преподаватель кафедры боевого применения и эксплуатации автоматизированных систем управления Харьковского национального университета Воздушных Сил им. И. Кожедуба, Харьков, Украина.

Пархоменко М. В. – канд. техн. наук, старший преподаватель кафедры боевого применения и эксплуатации автоматизированных систем управления Харьковского национального университета Воздушных Сил им. И. Кожедуба, Харьков, Украина.

АННОТАЦИЯ

Актуальность. Проблема локализации влияния ошибок в каналах передачи данных при использовании методов компрессионного и помехоустойчивого кодирования в условиях соблюдения оперативности доставки данных в инфокоммуникационных системах аэросегмента. Объектом исследования являются методы кодирования для повышения достоверности видеоинформационных ресурсов в инфокоммуникационных сетях с использованием аэромобильных платформ.

Метод. Использование методов помехоустойчивого кодирования для обеспечения необходимого уровня достоверности видеоинформации, передаваемой в инфокоммуникационных системах аэросегмента, имеет ряд существенных недостатков: приводит к значительному увеличению битового объема компактно представленных видеоданных; растет временная задержка на доставку видеоинформации, что в условиях использования аэромобильных платформ является критическим. Повышение временных задержек в процессе доставки видеоинформации приводит к тому, что: видеоинформация будет передана не в полном объеме и, как следствие, в условиях аэромониторинга к потере достоверности данных; увеличивается время на обработку видеоданных. Преимуществом использования технологий компрессионного кодирования для решения задачи повышения достоверности видеоинформации, передаваемой в инфокоммуникационных системах аэросегмента, является снижение битового объема видеоинформационного ресурса. Однако существующие технологии обработки видеоданных базируются на использовании методов статистического кодирования и выявлении серий одинаковых последовательностей повторяющихся элементов. Но использование таких технологий не обеспечивает необходимый уровень локализации действия ошибок. Разработан метод реструктуризации на основе выявления закономерностей во внутренней двоичной структуре элементов сообщения по количественному признаку. Инструментом для реструктуризации является признак количества серий единиц в двоичной структуре элементов сообщения. Отличительные характеристики метода состоят в том, что реструктуризация информационного пространства осуществляется без потери целостности на основе структурного признака по количеству серий единиц.

Результаты. Проведен анализ существующих направлений решения проблемы повышения уровня достоверности передаваемой в инфокоммуникационных системах аэросегмента видеоинформации. Разработанный метод внутренней реструктуризации данных позволяет получить следующие результаты: обеспечиваются условия дополнительного сокращения структурной избыточности кодового представления информации за счет существенного сокращения мощности информационного пространства в результате использования внутренней реструктуризации данных по признаку количества серий единиц; создаются условия для локализации действия ошибок в процессе реконструкции видеоинформационных ресурсов; создаются условия для сокращения времени на обработку данных в связи с тем, что разработанный метод реструктуризации данных не требует проведения преобразований над элементами сообщения.

Выводы. Необходимо совершенствовать существующие технологии компрессионного кодирования в направлении выявления закономерностей, учет которых позволит локализовать разрушительное действие ошибок, возникающих в канале связи.

КЛЮЧЕВЫЕ СЛОВА: видеоинформационный ресурс, кодирование, достоверность, оперативность, канал связи, аэросегмент, технологии сжатия.

ЛІТЕРАТУРА / ЛИТЕРАТУРА

1. Miano J. Compressed image file formats: JPEG, PNG, GIF, XBM, BMP [Text] / by John Miano. – 1999. – 264 p.
2. Pratt W. K. Slant transforms image coding [Text] / W. K. Pratt, W. H. Chen, L. R. Welch // Proc. Computer processing in communications. – 1969. – P. 63–84.
3. Wallace G. K. The JPEG Still Picture Compression Standard [Text] / G. K. Wallace // Communication in ACM. – 1991. – Vol. 34, No. 4. – P. 31–34.
4. Wallace G.K. Overview of the JPEG (ISO/CCITT) Still image compression: image processing algorithms and techniques [Text] / G. K. Wallace // Proc. of SPIE-IS&T Electronic Imaging (SPIE). – 1990. – Vol. 1244. – P. 220–233.
5. Utility Driven Adaptive Preprocessing for Screen Content Video Compression [Text] / [S. Wang, X. Zhang, X. Liu et al.] // IEEE Transactions on Multimedia. – 2017. – Vol. 19, No. 3. – P. 660–667.

DOI 10.15588/1607-3274-2022-3-12

6. Gonzales R. C. Digital image processing. [Text] / R. C. Gonzales, R. E. Woods // Prentice Inc. Upper Saddle River, 2002. – 779 p.
7. Dong W. JPEG Compression Forensics against Resizing [Text] / W. Dong, J. Wang // IEEE Trustcom/ BigDataSE/IвSPA. – 2016. – P. 1001–1007. DOI: 10.1109/TrustCom.2016.0168.
8. Richter T. Error Bounds for HDR Image Coding with JPEG XT [Text] / T. Richter // Data Compression Conference (DCC). – 2017. – P. 122–130. DOI: 10.1109/DCC.2017.7.
9. A Fast JPEG Image Compression Algorithm Based on DCT [Text] / [W. Xiao, N. Wan, A. Hong, X. Chen] // IEEE International Conference on Smart Cloud (SmartCloud). – 2020. – P. 106–110. DOI: 10.1109/ SmartCloud49737. 2020.00028.
10. Rippel, O. Learned Video Compression [Text] / O. Rippel // IEEE/CVF International Conference on Computer Vision (ICCV). – 2019. – P. 3453–3462. DOI: 10.1109/ICCV. 2019.00355.
11. Performance of H.264, H.265, VP8 and VP9 Compression Standards for High Resolutions [Text] / [J. Bienik, M. Uhrina, M. Kuba, M. Vaculik] // 19th International Conference on Network-Based Information Systems (NBiS). – 2016. – P. 246–252. DOI: 10.1109/NBiS. 2016.70.
12. Cruise UAV Video Compression Based on Long-Term Wide-Range Background [Text] / [X. Wang, J. Xiao, R. Hu, Z. Wang] // Data Compression Conference (DCC). – 2017. – P. 466–467. DOI: 10.1109/DCC. 2017.71
13. Minallah, N. Performance Analysis of H.265/ HEVC (High-Efficiency Video Coding) with Reference to Other Codecs [Text] / N. Minallah, S. Gul, M. Bokhari // 13th International Conference on Frontiers of Information Technology (FIT). – 2015. – P. 216–221. DOI: 10.1109/ FIT.2015.46
14. Neural Inter-Frame Compression for Video Coding [Text] / [A. Djelouah, J. Campos, S. Schaub-Meyer, C. Schroers] // IEEE/CVF International Conference on Computer Vision (ICCV). – 2019. – P. 6420–6428. DOI: 10.1109/ICCV.2019. 00652.
15. Tupytsya I. Methodology for restructuring information resource data to improve the efficiency of statistical coding [Text] / I. Tupytsya // Science-based technologies. – 2019. – Vol. 42, No. 2. – P. 262–269. DOI: 10.18372/ 2310-5461.42.13801.
16. Narmatha C. A LS-compression scheme for grayscale images using pixel based technique [Text] / C. Narmatha, P. Manimegalai, S. Manimurugan // International Conference on Innovations in Green Energy and Healthcare Technologies (IGEHT). – 2017. – P. 1–5. DOI: 10.1109/ IGEHT.2017.8093980.
17. Alam M. A. Faster Image Compression Technique Based on LZW Algorithm Using GPU Parallel Processing [Text] / M. A. Alam // Joint 7th International Conference on Informatics, Electronics & Vision (ICIEV) and 2nd International Conference on Imaging, Vision & Pattern Recognition (icIVPR). – 2018. – P. 272–275. DOI: 10.1109/ICIEV.2018.8640956.
18. The Application of the Internal Restructuring Method of the Information Resource Data According to the Sign of the Number of Series of Units to Improve the Statistical Coding Efficiency [Text] / [V. Barannik, I. Tupitsya, V. Barannik et al.] // 10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS). – 2019. – P. 65–69. DOI: 10.1109/IDAACS. 2019.8924460.
19. The concept of a quantitative sign formation for the internal restructuring of information resource data [Text] / [V. Barannik, I. Tupytsya, M. Parkhomenko et al.] // Przetwarzanie, transmisja i bezpieczeństwo informacji : monogrpah, 2020. – P.41–52.
20. Comparative analysis of lossless compression techniques in efficient DCT-based image compression system based on Laplacian Transparent Composite Model and An Innovative Lossless Compression Method for Discrete-Color Images [Text] / [T. Poolakkachalil, S. Chandran, R. Muralidharan, K. Vijayalakshmi] // 3rd MEC International Conference on Big Data and Smart City (ICBDSC). – 2016. – P. 1–6. DOI: 10.1109/ ICBDSC.2016.7460360.
21. Wang Z. Joint Learned and Traditional Video Compression for P Frame [Text] / Z. Wang, R. Liao, Y. Ye // IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). – 2020. – P. 560–564. DOI: 10.1109/CVPRW50498.2020.00075.
22. Comparison of lossless video and image compression codecs for medical computed tomography datasets [Text] / [V. Bui, L. Chang, D. Li et al.] // IEEE International Conference on Big Data (Big Data). – 2016. – P. 3960–3962. DOI: 10.1109/BigData. 2016.7841075.
23. Learned Variable-Rate Image Compression With Residual Divisive Normalization [Text] / [M. Akbari, J. Liang, J. Han, C. Tu] // IEEE International Conference on Multimedia and Expo (ICME). – 2020. – P. 1–6. DOI: 10.1109/ICME46284.2020.9102877.
24. The process of forming layers of bit zones in the method of crypto-semantic presentation of images on the basis of the floating scheme [Text] / [V. Barannik, S. Sidchenko, R. Tarnopolov, I. Tupitsya] // 1st International conference on advanced information and communication technologies (AICT). – 2015. – P. 59–62.
25. The Method of Crypto-Semantic Presentation of Images Based on the Floating Scheme in the Basis of the Upper Boundaries / [V. Barannik, I. Tupitsya, S. Sidchenko, R. Tarnopolov] // IEEE Problems of Infocommunications. Science and Technology (PICS&T). – 2015. – P. 248–251. DOI: 10.1109/ INFOCOMMST.2015.7357326.
26. Shinde T. Efficient Image Set Compression. [Text] / T. Shinde // IEEE International Conference on Image Processing (ICIP). – 2019. – P. 3016– 017. DOI: 10.1109/ICIP. 2019.8803230.
27. M-LVC: Multiple Frames Prediction for Learned Video Compression [Text] / [J. Lin, D. Liu, H. Li, F. Wu] // IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). – 2020. – P. 3543–3551. DOI: 10.1109/CVPR42600. 2020.00360.
28. Artuğer, F. Fractal Image Compression Method for Lossy Data Compression [Text] / F. Artuğer, F. Özkaynak // International Conference on Artificial Intelligence and Data Processing (IDAP). – 2018. – P. 1–6. DOI: 10.1109/IDAP. 2018.8620735.
29. Iris image compression using wavelets transform coding [Text] / [P. Arnob, Z. Tanvir, P. Prajoy et al.] // 2nd International Conference on Signal Processing and Integrated Networks (SPIN). – 2015. – P. 544–548. DOI: 10.1109/SPIN.2015.7095407.
30. Morphological component decomposition combined with compressed sensing for image compression [Text] / [X. Zhu, L. Liu, Na Jin, P. Ai] // IEEE International Conference on Information and Automation (ICIA), DOI: 10.1109/ICInfA.2016.7832096.

DOI 10.15588/1607-3274-2022-3-12

31. Encode when necessary: Correlated network coding under unreliable wireless links [Text] / [S. Wang, M. Kim, Z. Yin, T. He] // ACM Transactions on Sensor Networks. – 2017. – Vol. 13(1). DOI: 10.1145/ 3023953.
32. Phatak A. A Non-format Compliant Scalable RSA-based JPEG Encryption Algorithm [Text] / A. Phatak // International Journal of Image, Graphics and Signal Processing. – 2016. – Vol. 8, No. 6. – P. 64–71. DOI: 10.5815/ijigsp.2016.06.08.
33. Lossless Compression of JPEG Coded Photo Collections [Text] / [H. Wu, X. Sun, J. Yang et al.] // IEEE Transactions on Image Processing. – 2016. – Vol. 25, No. 6. – P. 2684–2696. DOI: 10.1109/TIP. 2016.2551366.
34. Lee J. Context-adaptive entropy model for end-to-end optimized image compression [Text] / J. Lee, S. Cho, S. K. Beack. – 2018. Arxiv: 1809.10452.
35. Chen C. A research on anti-jamming method based on compressive sensing for OFDM analogous system [Text] / C. Chen, Y. Zhuo // 17th International Conference on Communication Technology (ICCT). – 2017. – P. 655–659. DOI:10.1109/ICCT.2017. 8359718.
36. Encode when necessary: Correlated network coding under unreliable wireless links [Text] / [S. Wang, S. Kim, Z. Yin, T. He] // ACM Transactions on Sensor Networks. – 2017. – Vol. 13, No. 1. – P. 24 – 29. DOI: 10.1145/ 3023953.
37. Han S. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding [Text] / S. Han, H. Mao, W. Dally. – 2015. arXiv: 1510.00149.
38. Increasing the efficiency of information transmission in communication channels [Text] / [B. Zhurakovskyi, J. Boiko, V. Druzhynin et al.] // Indonesian Journal of Electrical Engineering and Computer Science. – 2020. – Vol. 19(3). – P. 1306–1315. DOI: 10.11591/ijeecs.v19.i3.
39. The application for internal restructuring the data in the entropy coding process to enhance the information resource security [Text] / [V. Barannik, S. Sidchenko, I. Tupitsya, S. Stasev] // IEEE East-West Design and Test Symposium (EWDTS). – 2016. – P. 1–4. DOI: 10.1109/ EWDTS.2016.7807749.
40. Two-Hierarchical Scheme of Statistical Coding of Information Resource Data with Quantitative Clustering [Text] / [V. Barannik, I. Tupitsya, I. Gurzhii et al.] // IEEE International Conference on Advanced Trends in Information Theory (ATIT). – 2019. – P. 89–92. DOI: 10.1109/ ATIT49449.2019.9030451.
41. The Method of Clustering Information Resource Data on the Sign of the Number of Series of Units as a Tool to improve the Statistical Coding Efficiency [Text] / [V. Barannik, I. Tupitsya, O. Dodukh et al.] // IEEE 15th International Conference on the Experience of Designing and Application of CAD Systems (CADSM). – 2019. – P. 32–35. DOI: 10.1109/CADSM.2019.8779243.
42. The analysis of the internal restructuring method efficiency used for a more compact representation of the encoded data [Text] / [V. Barannik, I. Tupitsya, O. Kovalenko et al.] // Advanced Trends in Information Theory (ATIT’2020). – 2020. – P.89–92. DOI: 10.1109/ATIT49449.2019.9030451.
43. Creating a mathematical model for estimating the impact of errors in the process of reconstruction of non-uniform code structures on the quality of recoverable video images [Text] / [O. Yudin, V. Artemov, A. Krasnorutsky et al.] // Advanced Trends in Information Theory (ATIT’2021). – 2021. – P. 38–41. DOI: 10.1109/ATIT54053.2021.9678887.
44. Development of the external restructuring method to increase the efficiency of information resource data encoding / [S. Khmelevskiy, I. Tupitsya, Q. A. Mahdi et al.] // Information Processing Systems. – 2021. – No. 3(166). – P. 52–61. https://doi.org/10.30748/soi.2021.166.06.

DOI 10.15588/1607-3274-2022-3-12

UDC 004.75

MODULE HIGH-EFFICIENCY MULTIPROCESSOR SYSTEM WITH MULTIDIMENSIONAL AGGREGATING OF CHANNELS OF NETWORK INTERFACE

Moroz D. M. – Assistant of the Department of Computer Systems Software, Dnipro University of Technology, Dnipro, Ukraine.

ABSTRACT

Context. In modern terms problem of constructing of the multiprocessor systems the special value acquires the base of standard popular technologies and components. It is caused by that such systems became popular and cheap vehicle platforms for high-performance calculations. In addition, practice pulls out problems complete decision of which in most cases possibly only due to application of high-performance calculations. Consequently, a theme of constructing of the cluster multiprocessor systems for today is actual, interesting and is on the stage of the active development. At the same time, the new high-quality stage of development of the multiprocessor cluster systems lies in area of the use of new modern network technologies. Presently the problem of choice and analysis of network technologies for the module multiprocessor cluster systems did not get due development, as well as problem of reorganization of structure ofnetwork interfaceby aggregating of channels of network interface.

Objective. An aim is in-process put improvement of structure and increase of the productivity of the multiprocessor computer system by the multidimensional aggregating of channels of network interface, adapted to the decision of tasks of the investigated class.

Method. The task of increase of efficiency of the module multiprocessor computer system is decided due to multidimensional aggregating of channels of network interface. Offered approach allowed not only to promote efficiency of parallelization but also substantially to decrease time of calculations. Such results succeeded to be attained due to diminishing to time of border exchange of data between the calculable knots of the cluster system.

Results. A feature offered approach is that he allowed to realize a direct exchange data between main memory of knots of the multiprocessor system, that promotes the fast-acting of calculations and provides high-speed access to memory of her slave -nodes. Thus during an exchange by data between the knots of the system the system *CPU* gets unloaded and loading of channel which passes between the knots of the computer system goes down, that assists diminishing of time of border exchange of data between the calculable knots of the system.

Conclusions. The results of the conducted experiments showed that the worked out multiprocessor system was used for creation of new technological processes. So, she is used in a fluidizer intensification of the сфероидизирующего annealing of long-length steelwork. Directly the technological process of heat treatment of metal acquires such advantages, as a high yield, substantial mionectic energy consumption and allows to carry out control of technological parameters in the modes of unisothermal treatment of metal.

KEYWORDS: multiprocessor systems, network interface, aggregating, fast-acting, memory, knots, latentness.

ABBREVIATION

VLAN − Virtual Local Area Network;
IB − InfiniBand;
Eth − Ethernet;
PM − master node;
PN − slave node;
VPI − Virtual Protocol Interconnect;
QoS − quality of service;
DDR − Double Data Rate;
RDMA − remote direct memory access;
DHCP − Dynamic Host Configuration Protocol;
HDR − High Dynamic Range;
NVMe − Non-Volatile Memory Express;
SLI − Scalable Link Interface;
MPI – Message Passing Interface.

NOMENCLATURE

ATX − Advanced Technology Extended;
TCA − Target Channel Adapters;
RAID − Redundant Array of Independent Disks;
CPU − Central Processing Unit;
GPU − Graphics Processing Unit;
SSD − Solid state drive;
CUDA − Compute Unified Device Architecture;
HCA − Host Channel Adapters;
T_h − temperature of heating of surface of standard;
T_k − controlled by facilities of the multiprocessor system, temperature of phase transformation of metal.

INTRODUCTION

Today there is a swift height of number of the multiprocessor computer systems and their total productivity in the world. It is caused by that such systems became popular and cheap vehicle platforms for high-performance calculations. In addition, practice pulls out before applied scientists of different problems complete decision of which in most cases possibly only due to application of the multiprocessor computer systems. For today there are many different variants of construction of the module multiprocessor computer systems. One of perspective is a process of constructing of such systems on the basis of “blade”-technologies. One of basic features of their constructing is constrained with the use of network technology the choice of which depends foremost, from the class of the tasks decided by users. Consequently, there is a problem of design of architecture of the multiprocessor

DOI 10.15588/1607-3274-2022-3-13

module computer systems, sent to the decision of wide circle of the applied tasks.

Consequently, a theme of constructing of the cluster multiprocessor systems for today is actual, interesting and is on the stage of the active development. Clear and other, by means of the high-performance module systems the effective method of implementation of actual tasks of class was found.

At the same time, the new high-quality stage of development of the multiprocessor cluster systems lies in area of the use of new modern network technologies. Thus efficiency of necessary on this stage parallelization of calculations is predefined by many factors among which decision are a choice and organization of network interface.

A research object are informative computational processes in the multiprocessor computer systems.

The article of research *is* conception of construction of the new module multiprocessor computer systems on the basis of reorganization of structure of network interface, decision-oriented certain class of tasks.

An aim to perfect a structure and promote the productivity of the multiprocessor system of calculations by aggregating of channels of network interface, adapted to the decision of tasks of the investigated class is in-process put.

1 PROBLEM STATEMENT

Today, production practice raises various problems, the complete solution of which in most cases is possible only through the use of multiprocessor computing complexes. Solving applied problems with the help of well-known standard approaches is a complex problem that can only be overcome by using modern multiprocessor computer technologies. At the same time, one of the main features of the application of such technologies is to increase the speed and productivity of computers. High computer performance allows you to solve multidimensional tasks, as well as tasks that require a large amount of processor time. Speed makes it possible to effectively manage technological processes or, in general, to create prerequisites for the development of new promising technological processes.

When designing new multiprocessor systems, the following must be taken into account. Modern HDR expansion technology allows data exchange between computing nodes at a speed of up to 200 Gbit/s. On the other hand, according to the manufacturer's data, HDR4 technology has a delay of 0.4–0.5μs. In this case, it is necessary to increase the efficiency and speed of the multiprocessor system by means of multidimensional aggregation of network interface channels. At the same time, data exchange between computing nodes must be transferred to a separate network that works at the channel (second) level using channel bonding technology. Such an approach will be aimed at increasing the speed of data exchange between cluster nodes and reducing the load on the channel that connects the cluster nodes. At the same time, due to the use of HDR4x2 technology, the speed of data exchange will increase from 200 Gbit/s. up to 400 Gbit/s. On the other hand, due to the separation of aggregated channels for the symmetrical use of controllers with aggregated components, the latency will decrease to 0.1μs.

An Intel Core I5 processor was used to design the multiprocessor system (using DDR4 memory type with RDMA support, which allows the combination of RAM ports up to 4 channels), which allows direct access to memory with support for InfiniBand technology. Such an approach will reduce the congestion of the channel that passes between the nodes of the computing system. In addition, the use of RDMA technology in the system will make it possible to eliminate the delay in sending data directly to the InfiniBand adapter.

2 REVIEW OF THE LITERATURE

In the modern terms of the special weight acquires creation of the multiprocessor computer systems on the base of standard popular technologies and components [1−3]. Due to high demand and supply on bladed configuration in scientific practice a "blade"-cluster calculable complex is offered exactly for the decision of tasks with the up-diffused area of calculations [4, 5]. Becomes clear that through high-performance clusters the effective method of decision of wide class of actual tasks is found.

In the process of planning of the module multiprocessor system his construction [has a large value 6]. Exactly on the stage of constructing of the multiprocessor system it is necessary to foresee possibilities of her expansion or modification in the future [7]. We will mark that the most successful decision, placing of the multiprocessor system is considered in a bar [8]. Such arrangement appeared appropriate even for the small computer system. Into a bar there are knots, apparatus for effective connection of components, management facilities by the intranet of the system and under. Every blade works under the management of the copy of the standard operating system. Composition and power of knots of the multiprocessor system can be разным. In hired the homogeneous system is examined. The interaction between the nodes of the computing system is established with the help of specialized libraries at the level of the OS kernel and switching hardware.

However at planning and effective use of the multiprocessor system basic attention is spared to the interconnect network of the system and her topology [9, 10]. Topology of cluster and his fast-acting at the decision of calculable tasks, undoubtedly, problems are constrained.

To our opinion, the new high-quality stage of development of the multiprocessor cluster systems lies in area of the use of new modern network technologies [11]. Thus efficiency of parallelization of calculations depends on many factors, however one of qualificatory are a choice and organization of network interface. It is explained as follows. The network of the cluster computer system fundamentally differs from the network of the work stations, although for the construction of cluster ordinary network maps and switchboards which are used during organization of network of the work stations are

DOI 10.15588/1607-3274-2022-3-13

needed. However in case of the cluster computer system there is one fundamental feature. The network of cluster, first of all, is intended not for connection of computers, and for connection of calculable processes. In this connection, than higher there will be a carrying capacity of computer network of cluster, the user parallel tasks, executable on a cluster, will be considered quicker. Thus, technical descriptions of computer network acquire a primary value for the multiprocessor cluster systems.

Presently the problem of choice and analysis of network technologies for the module multiprocessor cluster systems did not get due development, as well as problem of reorganization of structure ofnetwork interfaceby aggregating of channels of network interface. In addition, works, sanctified to research of influence of network technologies on efficiency of parallelization in the module multiprocessor cluster systems, are practically absent. In this connection, the researches examined in hired are actual and, primary.

3 MATERIALS AND METHODS

The module of the high-efficiency multiprocessor system consists of master-node (PM001) and slave-nodes (PN001, PN002, PN003, ..., PN00N), calculable operations oriented to implementation; two guided switchboards (Switch IB1, Switch IB2) in the networks of exchange by data between calculable knots; hybrid sluice (SkyWay IB3), oriented to the management by the system, her loading and diagnostics; virtual local networks, intermediate buffers of memory of the guided switchboards; networks of exchange by data in the system of slave-nodes; mechanism of backuping of basic components of the system. Loading of knots of the system is produced through corresponding switched local networks. On a Fig. 1 her flow-chart is presented.

The structural feature of the system consists of the following. In the interconnect network of exchange by data between slave – by the knots of the multiprocessor system for application of technology of InfiniBand [12] copper cables are used. Function of network adapters to execute network maps which support work of the module of the multiprocessor system in the standard of InfiniBand.

For constructing of the multiprocessor system the processor of Intel Core I5 (this processor 10 generations, using type of memory of DDR4 with support of RDMA, allowing the association of ports of main memory to 4 channels with frequency to 4 GHz and by volume of by 128 Gigabyte of RAM) was used.

The computer system foresees vertical, parallel in relation to each other, arrangement of system boards. Just the same approach and answers the idea of constructing of "blade"-servers. After loading of OC access to the multiprocessor system takes place by the use of standard network protocols (telnet, ssh, rsh). At that rate for organization of parallel calculations with participation of working computer and multiprocessor system it is necessary to set network connection between them. Such connection gets organized by means of topology "point-point".

After the serve of signal of START from a control of master-node (PM001) panel his electric feed is produced by means of the corresponding power (Fig. 1) module.

The process of start and initialising of master-node (PM001) of the module multiprocessor system begins further. The system foresees two modes of loading OS, namely: from a hard disk or from certain external media. According to setting unique dynamic IP-adresses and corresponding local networks by default loading of blades of the system is answered by the special configuration script. He is started right after loading of the operating system. In addition, this script will realize tuning of DHCP server on the whole. Exactly on this stage of work of the system the amount of slave-nodes of the computer system (PN001, PN002, PN003, .., PN00N) is appointed, and if necessary tuning of access is executed to the external networks or to the environment of the Internet.

The certain parameters of the system and her tuning are thus determined. The start of slave-nodes of the computer system (PN001, PN002, PN003, .., PN00N) and loading OS is produced by a serve to them corresponding feed from blocks (fig. 1). After implementation of the transferred operations a configuration script completes work, and the computer system will be ready to realization of the functions.

The multiprocessor system includes six virtual local networks of VLAN: Net1 − Net6. Thus a virtual local network of Net1 is a network of the interchannel aggregating of network interface of the multiprocessor system, she is oriented to realization of the interchannel aggregating of network interface through ports of IB3.Agr1 and IB3.Agr2 of hybrid sluice of SkyWay IB3. A virtual local network of Net2 is a network of external and cloudy interfaces, she is oriented to loading of management data by the system, to her configuration, and also cloudy converting of heterogeneous data with the purpose of their further treatment. A virtual local network of Net3 is a system network which will realize the rapid and energyeffective method of storage and exchange information. A system network is oriented to the rapid loading of the multiprocessor system, her configuration. Through this network the process of tuning of files of configuration of OS will be realized, storage of kernel OS, elements of safety of firewall and other the Virtual local networks of Net1 − Net3 is formed on the base of new hybrid sluice of NVIDIA InfiniBand. Family of module switchboards of NVIDIA InfiniBand is provided the very tall productivity and closeness of port with a deblocking high carrying capacity in a demihull. Such switchboards are perspective in tasks high-performance, cloudy calculations and artificial intelligence at less expenses and complication. They differ in the scaleable hierarchical aggregating, possibility of self-healing of networks, assured quality of service. The hybrid sluice of NVIDIA Skyway InfiniBand to Ethernet Gateway is used in this development. Basic descriptions of such sluice are presented in a Table 1.

DOI 10.15588/1607-3274-2022-3-13

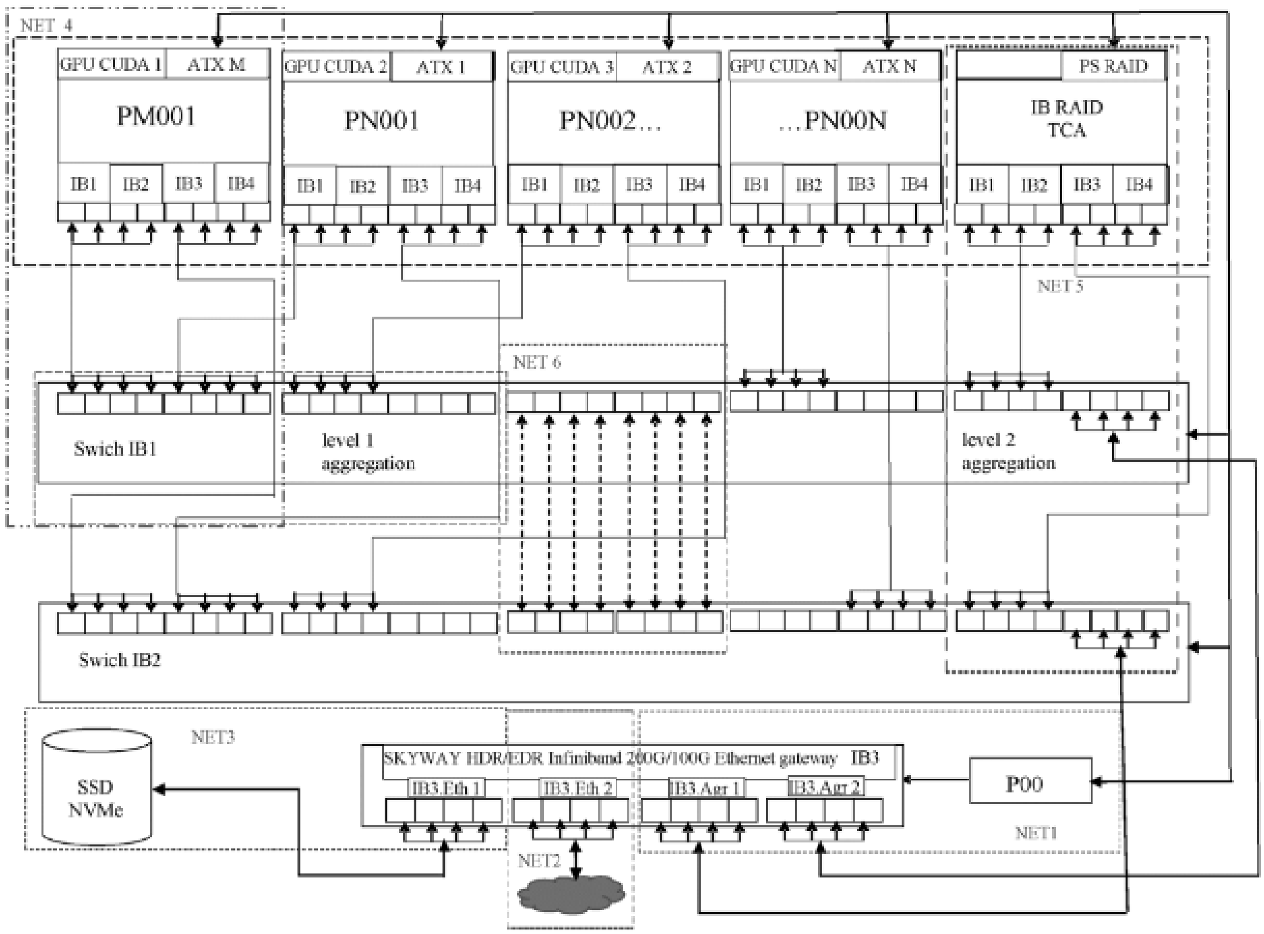


Figure 1 − Flow-chart of the module of the multiprocessor computer system with the multidimensional aggregating of channels of network interface

Table 1 − General descriptions of hybrid sluice NVIDIA Skyway InfiniBand to Ethernet Gateway

General carrying capacity	1.6 Tb/s
Common amount of ports	16
Amount of ports of IB	8x InfiniBand, 8x HDR HDR 200Gb/s
Amount of ports of Eth	8 ports of Ethernet, 8x 200/100Gb/s
Editing in a bar	2U

A virtual local network of Net4 is an interchannel network which will realize connection of the aggregated modules. In this connection she executes the functions of aggregating of channels. A virtual local network of Net5 is a network of technology of virtualization of data, she provides the increase of reliability of storage of data, and also serves for the rev-up of reading/notation of information. A virtual local network of Net6 is a network of the layer aggregating of data of network and interface.

The flow diagram of topology of network interface of the multiprocessor system is presented on a Fig. 2

Maximal efficiency of the module of the multiprocessor system is arrived at by reconfiguration of local network structure in accordance with the specific of the decided tasks (Fig. 2). Reconfiguration foresees the use of six modes of operations of network. Topology of local network of type a "star" answers the first mode, second, is a "line", third is a "complete count", fourth – "ring", fifth is a "grate" and finally sixth is the "reserved grate". Thus switching of the modes of topology of local network comes true by tuning of routers at hardwarily-programmatic level. To that end the modes of operations of ports of network maps change from full-duplex in half-duplex, and also in pairs to the symmetric aggregating of network ports of switchboards on a reception and/or communication of data. The concrete features of each of such modes of operations of reconfigurable network of the system in detail are reflected in literature [13, 14], where and the features of exchange open up by data between slave -nodes.

We will consider some fundamental advantages of process of calculations in the offered system. At first, in slave-nodes the process of acceptance/of communication of data is executed by facilities of the guided switch board IB without application of the mode of spooling. Secondly, intermediate and final calculations enter master-knot through switchboard Infiniband. Directly process of management and communication of the noted data from calculable slave-nodes will be realized by means of network adapters of *HCA*.

DOI 10.15588/1607-3274-2022-3-13

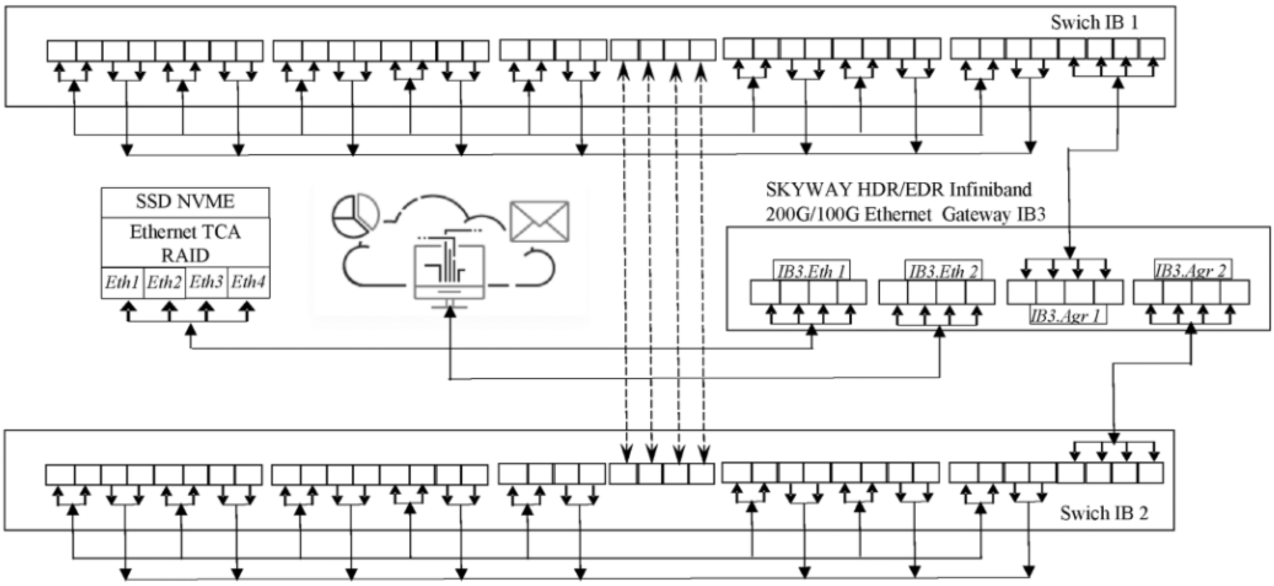


Figure 2 – Flow diagram of topology of network interface of the multiprocessor system

Finally, it is necessary to mark the fundamental feature of the use of technology of InfiniBand, in obedience to which all results of capture of data from input units, their treatment and transmission of the managing affecting out devices realized through the processor module with the interface of *TCA* (Fig. 2). In addition, storage of the indicated data with a management on a local network comes true on the hard disk of SSD with the interface of NVMe. Such approach allows substantially to promote efficiency of calculations.

For configuration of the considered network interaces the basic operations of tuning of the mode of Link Aggregation are executed. Directly procedure of "fastening of channels" or technology of fastening of channels, allowing to unite a few network adapters in an onespeed channel in full lighted up in-process [15]. It is here shown that exchange by data between the calculable knots of cluster comes true in a separate network, working on the basis of the mentioned technology of fastening of channels. Offered approach provides the rev-up of exchange data between the secondary sites of the multiprocessor system and to reduce loading on a channel, connecting the knots of the multiprocessor system.

For planning of the multiprocessor computer system the vehicle providing of corporation NVIDIA was used. In this connection possibility of the use of software and hardware architecture of parallel calculations appears on the basis of platform of *CUDA*. Such platform allows to use the resources of video cards for ungraphic calculations. And to date this technology becomes more actual.

The platform of *CUDA* is supported in all video cards of corporation NVIDIA, since the simplest user and to specialized powerful. An acceleration through the platform of *CUDA* is used by many user applications, for example, MatLab, TensorFlow, Keras of and other In this connection programming, with the use of resources of video card is actual and perspective. Thus get the large productivity for a moderate price.

Basic advantages of platform of *CUDA* are her free of chargeness (software for all basic platforms is freely got from the resource of developer.nvidia.com), simplicity and flexibility. Technology of NVIDIA CUDA is an environment which allows developers to create software for the decision of intricate calculable problems for less time, due to multinuclear calculable power of graphic processors.

Procedure of programming, applied in *CUDA* differs from traditional that fully hides a graphic conveyer from a programmer, allowing him to write the programs those in more usual for him "terms". In addition, *CUDA* gives to the programmer more comfortable model of work with memory. Here is not a necessity to keep data in 128-bit textures, as *CUDA* allows to read data straight from memory of video card.

For constructing of the multiprocessor computer system the video card of corporation NVIDIA GeForce GTX 1080 was used. Basic descriptions of such video card are presented in a Table 2.

Table 2 – General descriptions of video card NVIDIA GeForce GTX 1080

Amount of kernels	2560
Base frequency	1607 Mhz
Videomemory	8 Gbit
Carrying capacity of channel of memory	10 Gbps
Bit of interface	256-bit
Productivity	277.3 *Gflops*

DOI 10.15588/1607-3274-2022-3-13

On the basis of the use of technology *CUDA* were realized algorithms of calculation of process of heat transfer [16]. The analysis of time of implementation of parallel algorithms showed that application of technology of *CUDA* in times is abbreviated by time of processing of these experimental data. The practical testing showed, effectiveness of such calculations exceeded on an order statistics on similar calculations on central processing unit, even with the use of technology of OpenMP. In time is obvious also, that does not make sense to use technology of *CUDA* for work with the small volumes of data. For the small volumes of datains an acceleration practically is not observed.

On the other hand, it should be noted that the NVIDIA GeForce GTX 1080 video card was selected taking into account the compatibility of the scalable SLI communication interface. When creating the latest technological processes [16], this approach is extremely relevant. The proprietary SLI technology allows you to distribute the calculation between two video cards. Quad SLI extends this technology by allowing two dual-*GPU* graphics cards to use four *GPU*s simultaneously. In the above development of a multiprocessor system, two NVIDIA GeForce GTX 1080 video cards were "linked". This approach is aimed not only at a significant increase in computing performance, but also at a significant decrease in latency and a significant unloading of the system bus.

Installation of Quad SLI systems differs in sufficient simplicity. Inserting the indicated two video cards in sockets and connecting SLI-мост, after the start of the system new video cards are installed as an ordinary video adapter. Further after setting of drivers and finding out Quad SLI, рендеринг Quad SLI will be included by default. The new, additional tuning is not foreseen thus. We will notice that the variant of disconnecting of technology of Quad SLI is foreseen in a control of the multiprocessor system panel, that will allow two video cards of NVIDIA GeForce GTX 1080 to work independent of each other. This procedure is extraordinarily important for realization of new installation procedures with the video cards of the indicated type.

We will mark that for "fastening" of two video cards the bridge of SLI is used. Thus, a corporation Nvidia uses a physical socket for connection of video cards together, that allows to co-operate them with each other, not using the stripe of key-in in slots. Thus, one of two bridges of SLI may need: either standard bridge (for less powerful maps) or bridge with a high carrying capacity (for more powerful maps). Applied more powerful map (NVIDIA GeForce GTX 1080), a standard bridge can use, but it will not allow to provide the complete productivity of video cards. Descriptions of bridges of SLI are driven to the Table 3.

At installations SLI of the system the additional cooling of corps of multiprocessor system is provided. For such system a corps is used with a 120 mm by a ventilator, located opposite the sockets of video cards. Placing a 120 mm of ventilator opposite these two video cards allows considerably to reduce the temperature of video cards of *GPU*. Video cards are assembled so that they выдувают hot air through opening on the backplane of corps of the system. In this connection there is not a necessity to apply additional special cooling, it is necessary only to provide normal ventilation.

Table 3 – Descriptions of SLI of bridges

Bridge	Clock rate	Maximal carrying capacity
Standard bridge	400 Mhz	1 Gbps
Bridge with a high carrying capacity	650 Mhz	2 Gbps

In respect of energy consumption, then in the similar systems it is necessary to use the high-quality power modules. In the construction of the multiprocessor system the power of type of Corsair HX 1200 module was used watt which without problems provides the feed of Quad SLI systems. The offered multiprocessor system did not force the power module to work to capacity. The applied video cards work also quietly enough, thus even in Quad SLI configurations with noise for them problems are not present.

In addition, it is necessary to underline that graphic processor, possessing powerful calculable capabilities, however would not be able fully to replace activity of central processing unit absolute advantage of which is universality, but in his forces substantially to unload *CPU*, undertaking loading, presented by the most labour intensive and difficult tasks.

4 EXPERIMENTS

The worked out multiprocessor system is used in a fluidizer intensification of the spheroidizing annealing of long-length steelwork [16]. Setting of have for an object substantially to shorten duration of technological process of the spheroidizing annealing of metal due to the use of unisothermal self-control, that allows to improve technological properties of rolled metal with providing of high dispersion and homogeneity of structure of standard on all plane of his section. Directly the technological process of heat treatment of metal acquires such advantages, as a high yield, substantial mionectic energy consumption and allows to carry out control of technological parameters in the modes of unisothermal treatment of metal.

The task is achieved due to the fact that the installation for intensifying spheroidizing annealing of a long steel product is equipped with a multiprocessor computer system with specially oriented software installed on it. At the same time, the multiprocessor computing system is connected via an information bidirectional communication interface with a process control unit. The multiprocessor computing system is made as a separate module and allows using special software to set and control the necessary temperature conditions on the entire sectional plane of the sample during heating and holding the metal, as well as control the mode of non-isothermal annealing of steel, while the multiprocessor computing system aims to control the thermal regime processing is constant in the annealing temperature range.

DOI 10.15588/1607-3274-2022-3-13

The use of a multiprocessor computer system with its software makes it possible, on the basis of a mathematical model of the process of heating a sample, already under production conditions to control the heating of the sample to the transition to the austenitic region and the temperature of phase recrystallization on the entire plane of the cut of a long steel product, and then, having solved the inverse problem of heat conductivity, to control the necessary non-isothermal exposure mode in the annealing temperature range over the entire sample cut plane. The use of the installation for the implementation of the intensive mode of spheroidizing annealing predetermines the uniform distribution of cementite globules in the ferrite matrix, which provides the necessary mechanical properties of the metal required for further cold deformation.

5 RESULTS

On results experiments the crooked distributions of temperature of standard are got on the plane of his cut where determined: T_h − is a temperature of heating of surface of standard, T_k − controlled by facilities of the multiprocessor system temperature of phase transformation of metal. The design of such temperature fields comes true taking into account the change of thermophysical properties of material during his heating. The microstructure of standards after heat treatment purchased values 150−169 HB.

The performed spheroidization of the carbide phase of the metal under the conditions of the corresponding modes of heat treatment of the workpieces provides the material with the structure of granular perlite. Moreover, high-speed spheroidization predetermines a more uniform distribution of cementite globules in a ferrite matrix. After heat treatment, steel samples of almost the same hardness acquired a finely dispersed structure, which ensures a high level of metal ductility. Due to the rapid heating of the sample and incomplete austenitization of steel, certain changes occur in the morphology of the carbide phase from lamellar to finely dispersed globular.

The use of the proposed multiprocessor system can significantly improve the operational and technical characteristics of the technological process as a whole. This is due to the introduction of the following factors. Thus, in comparison with the well-known approach [17], due to the use of a processor module with a new generation *TCA* interface and an *SSD* hard disk with an NVMe interface, it was possible to reduce the operating system boot time on the main node by 180%, on the slave nodes by 320%; network interface software reorganization time decreased by 530%; the time for processing, sending and storing intermediate and final calculation results has decreased by 250%; 240% reduction in processing time for system statistics.

Through the use of VLANs and multidimensional network interface link aggregation, it was possible to increase the throughput of the network interface port from 200 Mb / s to 800 Mb / s, which increases the speed of data exchange between the nodes of a multiprocessor system four times.

Compared to the well-known approach [17], due to the use of the software and hardware architecture of parallel computing by NVIDIA Corporation based on the *CUDA* platform, it was possible to increase the amount of video memory by 16 GB on each computing node of the multiprocessor system, as well as increase the overall performance of the system node by 350 Gfl.

6 DISCUSSION

It was found out on the basis of analysis of methods of decision of the modern applied tasks, that application of the parallel computer systems – one of strategic directions of development of informative technique. It is possible explained by the permanent height of amount of the applied tasks, for the decision of which possibilities of present computing facilities failing. Obviously, that by means of the high-efficiency module systems the successful method of decision of actual tasks of wide class was found. For this reason in hired the problem of constructing of the high-efficiency multiprocessor system was lighted up.

This paper considers a complex formalized approach to designing a modular multiprocessor system with multidimensional network interface aggregation. At the same time, an analysis of approaches to the design of multiprocessor systems showed that recently computer equipment manufacturers have been offering devices based on blade technologies (blade technologies). Under such circumstances, by constructing a multiprocessor system based on blade servers, a turnkey solution is obtained, equipped with the necessary management tools and a network interface. We note some of the main advantages of such design solutions compared to others: blade systems are more compact and easy to maintain, their design features make it convenient to form the required configuration.

The analysis of constructing of the multiprocessor systems showed that the new high-quality stage of development of the multiprocessor computer systems was in the field of the use of new modern network technologies. It can be explained by substantial differences between the network of the cluster computer system and network of the work stations. Yes, the network of the computer system is intended not for connection of computers, and for connection of calculable processes. Then, than higher there will be a carrying capacity of computer network of the system, the set an user parallel tasks will be executed quicker. Consequently, technical descriptions of computer network acquire a primary value, when speech goes about the multiprocessor computer systems. Having regard to marked, it was made decision to apply technology of network interface of InfiniBand. Consequently, exchange between the knots of the multiprocessor system organized data by means of standard of InfiniBand. It is shown that by comparison to other multiprocessor systems, the worked out system has such fundamental differences: network loading of processors, maintenances of the mode of VLAN, mechanism of backuping of key constituents of the module, specially worked out mode of exchange by

DOI 10.15588/1607-3274-2022-3-13

data between the knots of the system in the network of switchboards of InfiniBand.

When designing a multiprocessor system, special attention was paid to traffic between neighboring nodes of a multiprocessor system, which is the slowest part of the algorithm of performed calculations and can significantly reduce the effect of increasing the number of processors involved. Exactly this circumstance and allows to talk that one of basic ways of increase of efficiency of the multiprocessor systems consists in aggregating of channels of network interface in the network of exchange by data between slave-nodes of the system. Thus, it is set that a theme of increase of efficiency of the multiprocessor systems due to reorganization of structure of network interface for today is actual, interesting, and her research is on the stage of active development.

A comparative analysis showed that presently the problem of aggregating of channels in the module multiprocessor cluster systems is not decided properly. Besides, critically small works in which influence of architecture of network of the cluster system would be investigated on efficiency of parallelization of calculations.

At the same time, multidimensional aggregation of channels of the network interface of a multiprocessor system is implemented on the basis of six VLANs. The interchannel aggregation network of the network interface of a multiprocessor system, external and cloud interfaces, as well as the control and diagnostics network are formed on the basis of the new NVIDIA InfiniBand hybrid gateway. On the other hand, the use of modular NVIDIA InfiniBand switches supports a standard set of network technologies, in particular virtual networks, traffic prioritization, aggregated links, and multicast traffic filtering. The family of such switches is promising in high-performance, cloud computing and artificial intelligence tasks at lower cost and complexity. In addition, they are distinguished by scalable hierarchical aggregation, self-healing networks, and guaranteed quality of service.

The technology of fastening of channels of network interface of the multiprocessor cluster system worked out in hired allows to unite the knots of cluster in a network so, if each of them was connected to the switchboard more than by one channel. The described technology is similar to the mode of tracking at connection of switchboards, due to which it is succeeded to rev up communication of data between two or by a few switchboards. Application of procedure of fastening of channels allows to attain the even partition of load (acceptance/of communication of data) between them in the multiprocessor system and to promote speed of exchange data between her knots.

Once again it is important to pay attention to main advantage of the mode aggregating of channels, due to which substantially speed of exchange rises by data, and also reliability of functioning of the cluster system indexes grow. So, in case of refuse adapter a traffic is sent to the next in good condition adapter without breaking of service. When an adapter again begins to work, then sending of data recommences through him.

To implement multidimensional link aggregation of the network interface, the advantage was given to the adapter from Mellanox. At the same time, network adapters of the MHQH29C – XTR type were chosen, which, supporting switching according to the virtual protocol VPI, provide flexibility in connections in computing systems. Under such conditions, a multiprocessor system provides high-quality computing, high-speed access to data storage resources, guaranteed high throughput, and low data transfer latency.

When designing a multiprocessor system, special attention was paid to its practical aspects of functioning. Thus, due to the use of the hybrid NVIDIA Skyway InfiniBand to Ethernet Gateway, a processor module with a new generation *TCA* interface, as well as the use of the *CUDA* platform, it was possible to significantly increase the computing power of a multiprocessor system without wasting time on reorganizing the network interface operating modes to solve the required class of applied tasks.

Computational experiments were carried out under the control of cluster operating systems using VLAN technology and a set of MPI libraries in the object-oriented programming environment of the C# language.

CONCLUSIONS

In the article the ways of increase of efficiency of the multiprocessor cluster system are shown due to reorganization of architecture of her network interface. Offered approach allowed not only to promote efficiency of распараллеливания but also substantially to decrease time of calculations. Such results succeeded to be attained due to diminishing to time of border exchange of data between the calculable knots of the cluster system.

The transferred signs of the worked out system allowed her to perfect with acquisition of certain differences from present to the system, namely:

– at first, due to realization of technology of InfiniBand were it is attained such advantages: subzero latentness and high fast-acting;

– secondly, possibility to change – interface configuration of local networks of the system through a control or WEB stand, adapting their structures to the decision of tasks of that or other type;

– thirdly, on the basis of principle of RDMA of technology of InfiniBand a direct exchange comes true by data between main memory of knots of the multiprocessor system, which promotes the fast-acting of calculations and provides high-speed access to memory of her slave - nodes systems, and also exchange by data between them, off-loading the system *CPU* during an exchange by data and reducing loading of channel which passes between the knots of the computer system;

– fourthly, application of multichannel hybrid sluice of NVIDIA Skyway InfiniBand in a copula with the processor module of *TCA* with the interface of NVMe and hard disk of *SSD* creates fundamentally new possibilities of “connectivity” of such system with other calculable environments; dirigibility of the system allows substantially to promote; in particular, to unload central process-

DOI 10.15588/1607-3274-2022-3-13

ing unit (due to maintenance of traffic of InfiniBand); to reduce time on switching of the modes of operations of virtual networks, collection, transmission and storage of results of calculations and, as a result, promote efficiency of all multiprocessor system on the whole;

– fifthly, due to module principle of construction to simplify the processes of planning, increase or replacement of those calculable knots which broke ranks, and also on the whole exploitation of all constructed system.

The scientific novelty. For the first time, a procedure for multidimensional aggregation of network interface channels of a multiprocessor computing system was proposed, which made it possible to increase its performance and speed.

The practical significance. The results of the experiments carried out make it possible to recommend the developed multiprocessor system for creating new technological processes.

Prospects for further research. The developed multiprocessor system can be used to solve a wide range of applied problems as a highly efficient platform for high performance computing.

REFERENCES

1. Robey R., Zamora Y. Parallel and High Performance Computing. 2021. 704 p.
2. Voevodin Vl. V., Zhumatiy S. A. Computing and cluster systems. Moscow, 2007, 150 p.
3. Bashkov E. O., Ivashchenko V. P., Shvachich G. G. A highly productive rich processor system based on a personal enumeration cluster, *Problems of modeling and automation of design*, 2011, No. 9(179). pp. 312–324. DOI 10.31474/2074-7888.
4. Shvachych G., Pobochii I., Ivaschenko O., Busygin V. Research of compatibility in the multi-processing compound systems, Science Review. Poland, 2018, No. 2(9), Vol. 1. pp. 15–19. Access mode: https://ws-conference.com/.
5. Zhou D., Lo V. Cluster Computing on the Fly: resource discovery in a cycle sharing peer-to-peer system, *IEEE International Symposium on Cluster Computing and the Grid,* 2004, pp. 66–73. DOI: 10.1109/CCGrid.2004.1336550.
6. Ivashchenko V. P., Bashkov E. O., Shvachych G. G., Tkach M. O., Shcherbyna P. O., Volnyanskyi V. V. Patent 105438, Ukraine, IPC G06F 15/16 (2006.01) Module of a highly efficient multiprocessor system with an extended computing area /; patent owners: National Metallurgical Academy of Ukraine, Donetsk National Technical University. No. u 201507019; statement 14.07.2015; published 25.03.2016. Bul. No. 6.
7. Shvachych G., Pobochii I., Khokhlova T., Kholod O., Busygin V., Moroz D. Multiprocessor Computing based Parallel Structures of Mathematical Models of Tridiagonal Systems, *2020 International Conference on Inventive Computation Technologies (ICICT)*, 2020, pp. 1031–1035. DOI: 10.1109/ICICT48043.2020.9112512.
8. Latsis A. O. How to build and use a supercomputer. Moscow, 2003, 240 p.
9. Glushan V. M., Lavrik P. V. Network technologies as a tool for distributed electronic design, *Software Journal: Theory and Applications,* 2013, No. 1, pp. 1–12. DOI: 10.15827/2311-6749.7.119.
10. Ivashchenko V. P., Bashkov E. A., Shvachich G. G., Tkach M. A. Modern communication technologies in modular multiprocessor systems: experience of use, study of efficiency estimates, prospects for application. Dnepropetrovsk, 2012, 140 p.
11. Besta M., Hoefler T. Slim fly: A cost effective low-diameter network topology, *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, IEEE Press*, 2014, pp. 348–359. DOI: 10.1109/SC.2014.34.
12. Shvachich G. G., Kholod E. G., Tkach M. A. Peculiarities of the use of the INFINIBAND seismic interface in multiprocessor systems with a distributed computing area, *Information processing systems*, 2016, No. 3(140), pp. 175–181.
13. Gergel V. P., Strongin R. G. Fundamentals of parallel computing for multiprocessor computing systems, 2003, 184 p.
14. Maksimov N. V., Popov I. I. Computer networks. Moscow, 2011, 464 p.
15. Moroz D. M. Aggregation arrangement features of network interface channels in multiprocessor computing systems, *System technologies*, 2022, No. 2(139), pp. 111–121. DOI 10.34185/1562-9945-2-139-2022-11.
16. Shlomchak G., Shvachych G., Moroz B., Fedorov E., Kozenkov D. Automated control of temperature regimes of alloyed steel products based on multiprocessors computing systems, *Metalurgija*, 2019, No. 58 (3–4), pp. 299–302. Access mode: https://pubweb.carnet.hr/metalurg/arhiva/1983.
17. Shvachych G., Moroz B., Khylko M., Perepolkina O., Busygin V. Distributed Data Register Technology as the Main Component of Economic Decentralization, *Applied Computational Technologies*. Springer, 2022, pp. 533–542. Access mode: https://doi.org/10.1007/978-981-19-2719-5.

Received 20.06.2022.
Accepted 29.08.2022.

УДК 004.75

МОДУЛЬНА ВИСОКОЕФЕКТИВНА БАГАТОПРОЦЕСОРНА СИСТЕМА З БАГАТОВИМІРНОЮ АГРЕГАЦІЄЮ КАНАЛІВ МЕРЕЖЕВОГО ІНТЕРФЕЙСУ

Мороз Д. М. – асистент кафедри програмного забезпечення комп’ютерних систем, НТУ «Дніпровська політехніка», Дніпро, Україна.

АНОТАЦІЯ

Актуальність. У сучасних умовах проблема конструювання багатопроцесорних систем базі стандартних загальнодоступних технологій і компонентів набуває особливе значення. Це викликано тим, що такі системи стали загальнодоступними і дешевими апаратними платформами для високопродуктивних обчислень. Крім того, практика висуває проблеми, повне вирішення яких у більшості випадків можливо лише за рахунок застосування високопродуктивних обчислень. Отже, тема конструювання кластерних багатопроцесорних систем на сьогодні є актуальною, цікавою і знаходиться на етапі свого активного розвитку. В той же час, новий якісний етап розвитку багатопроцесорних кластерних систем лежить в області викорис-

DOI 10.15588/1607-3274-2022-3-13

тання нових сучасних мережевих технологій. Нині проблема вибору та аналізу мережевих технологій для модульних багатопроцесорних кластерних систем не отримала належного розвитку, як і проблема реорганізації структури мережевого інтерфейсу шляхом агрегації каналів мережевого інтерфейсу.

Мета. У роботі поставлена мета удосконалення структури та підвищення продуктивності багатопроцесорної обчислювальної системи шляхом багатовимірної агрегації каналів мережевого інтерфейсу, адаптованого до розв'язку задач досліджуваного класу.

Метод. Розв'язана задача підвищення ефективності модульної багатопроцесорної обчислювальної системи за рахунок багатовимірної агрегації каналів мережевого інтерфейсу. Запропонований підхід дозволив не лише підвищити ефективність розпаралелювання, але й істотно зменшити час обчислень. Таких результатів вдалося досягти за рахунок зменшення часу граничного обміну даними між обчислювальними вузлами кластерної системи.

Результати. Особливістю запропонованого підходу є те, що він дозволив реалізувати прямий обмін даними між оперативною пам'яттю вузлів багатопроцесорної системи, що підвищує швидкодію обчислень і забезпечує високошвидкісний доступ до пам'яті її slave-вузлів. При цьому під час обміну даними між вузлами системи розвантажується система *CPU* і знижується завантаження каналу, який проходить між вузлами обчислювальної системи, що сприяє зменшенню часу граничного обміну даними між обчислювальними вузлами системи.

Висновки. Результати проведених експериментів показали, що розроблена багатопроцесорна система використовується для створення нових технологічних процесів. Так, вона застосовується в установці для інтенсифікації сфероїдизуючого відпалу довгомірного сталевого виробу. Безпосередньо технологічний процес термічної обробки металу придбаває такі переваги, як висока продуктивність, істотне зниження енергоспоживання і дозволяє здійснювати контроль технологічних параметрів в режимах неізотермічної обробки металу.

КЛЮЧОВІ СЛОВА: багатопроцесорні системи, мережевий інтерфейс, агрегація, швидкодія, пам'ять, вузли, латентність.

УДК 004.75

МОДУЛЬНАЯ ВЫСОКОЭФФЕКТИВНАЯ МНОГОПРОЦЕССОРНАЯ СИСТЕМА С МНОГОМЕРНОЙ АГРЕГАЦИЕЙ КАНАЛОВ СЕТЕВОГО ИНТЕРФЕЙСА

Мороз Д. М. – ассистент кафедры программного обеспечения компьютерных систем, НТУ «Днепровская политехника», Днепр, Украина.

АННОТАЦИЯ

Актуальность. В современных условиях проблема конструирования многопроцессорных систем базе стандартных общедоступных технологий и компонентов приобретает особое значение. Это вызвано тем, что такие системы стали общедоступными и дешевыми аппаратными платформами для высокопроизводительных вычислений. Кроме того, практика выдвигает проблемы, полное решение которых в большинстве случаев возможно лишь за счет применения высокопроизводительных вычислений. Следовательно, тема конструирования кластерных многопроцессорных систем на сегодня является актуальной, интересной и находится на этапе своего активного развития. Вместе с тем, новый качественный этап развития многопроцессорных кластерных систем лежит в области использования новых современных сетевых технологий. В настоящее время проблема выбора и анализа сетевых технологий для модульных многопроцессорных кластерных систем не получила должного развития, как и проблема реорганизации структуры сетевого интерфейса путем агрегации каналов сетевого интерфейса.

Цель. В работе поставлена цель усовершенствование структуры и повышение производительности многопроцессорной вычислительной системы путем многомерной агрегации каналов сетевого интерфейса, адаптированного к решению задач исследуемого класса.

Метод. Решена задача повышения эффективности модульной многопроцессорной вычислительной системы за счет многомерной агрегацией каналов сетевого интерфейса. Предложенный подход позволил не только повысить эффективность распараллеливания, но и существенно уменьшить время вычислений. Таких результатов удалось достичь за счет уменьшения времени граничного обмена данных между вычислительными узлами кластерной системы.

Результаты. Особенностью предложенного подхода является то, что он позволил реализовать прямой обмен данными между оперативной памятью узлов многопроцессорной системы, что повышает быстродействие вычислений и обеспечивает высокоскоростной доступ к памяти ее slave -узлов. При этом во время обмена данными между узлами системы разгружается система *CPU* и снижается загрузка канала, который проходит между узлами вычислительной системы, что способствует уменьшению времени граничного обмена данных между вычислительными узлами системы.

Выводы. Результаты проведенных экспериментов показали, что разработанная многопроцессорная система используется для создания новых технологических процессов. Так, она применяется в установке для интенсификации сфероидизирующего отжига длинномерного стального изделия. Непосредственно технологический процесс термической обработки металла приобретает такие преимущества, как высокая производительность, существенное сниженное энергопотребление и позволяет осуществлять контроль технологических параметров в режимах неизотермической обработки металла.

КЛЮЧЕВЫЕ СЛОВА: многопроцессорные системы, сетевой интерфейс, агрегация, быстродействие, память, узлы, латентность.

ЛІТЕРАТУРА / ЛИТЕРАТУРА

1. Robey R. Parallel and High Performance Computing / R. Robey, Y. Zamora. – Publisher : Manning, 2021. – 704 p.
2. Воеводин Вл. В. Вычислительное дело и кластерные системы / Вл. В. Воеводин, С. А. Жуматий. – М. : Изд.-во МГУ, 2007. – 150 с.
3. Башков Є. О. Високопродуктивна багатопроцесорна система на базі персонального обчислювального класте-

DOI 10.15588/1607-3274-2022-3-13

ра / Є. О. Башков, В. П. Іващенко, Г. Г. Швачич // Проблеми моделювання та автоматизації проектування. – 2011. – № 9(179). – С. 312–324. DOI 10.31474/2074-7888.

4. Research of compatibility in the multi-processing compound systems / [G. Shvachych, I. Pobochii, O. Ivaschenko, V. Busygin] // Science Review. – 2018. – № 2(9), Vol. 1. – P. 15–19. Access mode: https://ws-conference.com/.
5. Zhou D. Cluster Computing on the Fly: resource discovery in a cycle sharing peer-to-peer system / D. Zhou, V. Lo // IEEE International Symposium on Cluster Computing and the Grid, Apr. 19 2004 to Apr. 22 2004. – Chicago, 2004. – P. 66–73. DOI: 10.1109/CCGrid.2004.1336550.
6. Патент 105438, Україна, МПК G06F 15/16 (2006.01) Модуль високоефективної багатопроцесорної системи з розширюваною областю обчислень / В. П. Іващенко, Є. О. Башков, Г. Г. Швачич, М. О. Ткач, П. О. Щербина, В. В. Волнянський В. В.; патентовласники: Національна металургійна академія України, Донецький національний технічний університет. – № u 201507019; заявл. 14.07.2015; опубл. 25.03.2016. Бюл. №6.
7. Shvachych G. Multiprocessor Computing based Parallel Structures of Mathematical Models of Tridiagonal Systems / [G. Shvachych, I. Pobochii, T. Khokhlova et al.] // International Conference on Inventive Computation Technologies (ICICT), Feb. 26 2020 to Feb. 28 2020. – Coimbatore, 2020. – P. 1031–1035. DOI: 10.1109/ICICT48043.2020.9112512.
8. Лацис А. О. Как построить и использовать суперкомпьютер / А. О. Лацис. – М. : Бестселлер, 2003. – 240 с.
9. Glushan V. M. Network technologies as a tool for distributed electronic design / V. M. Glushan, P. V. Lavrik // Software Journal: Theory and Applications. – 2013. – № 1. – P. 1–12. DOI: 10.15827/2311-6749.7.119.
10. Современные коммуникационные технологии в модульных многопроцессорных системах: опыт использования, исследование оценок эффективности, перспективы применения / [В. П. Иващенко, Е. А. Башков, Г. Г. Швачич, М. А.Ткач]. – Днепропетровск : ИМА-пресс, 2012. – 140 с.
11. Besta M. Slim fly: A cost effective low-diameter network topology / M. Besta, T. Hoefler // Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, IEEE Press, Nov. 16 2014 to Nov. 21 2014. New Orleans, 2014. – P. 348–359. DOI: 10.1109/SC.2014.34.
12. Швачич Г. Г. Особенности применения сетевого интерфейса INFINIBAND в многопроцессорных системах с распределенной областью вычислений / Г. Г. Швачич, Е. Г. Холод, М. А. Ткач // Системи обробки інформації. – 2016. – № 3(140). – С. 175–181. Режим доступу: https://journal-hnups.com.ua/index.php/soi.
13. Гергель В. П. Основы параллельных вычислений для многопроцессорных вычислительных систем / В. П. Гегель, Р. Г. Стронгин. – Нижний Новгород : Изд.-во ННГУ, 2003. – 184 с.
14. Максимов Н. В. Компьютерные сети / Н. В. Максимов, И. И. Попов. – М. : Форум, 2007. – 352 с.
15. Moroz D. M. Aggregation arrangement features of network interface channels in multiprocessor computing systems / D. M. Moroz // System technologies. – 2022. – № 2(139). – P. 111–121. DOI 10.34185/1562-9945-2-139-2022-11.
16. Automated control of temperature regimes of alloyed steel products based on multiprocessors computing systems / [G. Shlomchak, G. Shvachych, B. Moroz et al.] // Metalurgija. – 2019. – № 58 (3–4). – P. 299–302. Access mode: https://pubweb.carnet.hr/metalurg/arhiva/1983.
17. Distributed Data Register Technology as the Main Component of Economic Decentralization / [G. Shvachych, B. Moroz, M. Khylko et al.] // Applied Computational Technologies. Springer. – 2022. – P. 533–542. Access mode: https://doi.org/10.1007/978-981-19-2719-5.

DOI 10.15588/1607-3274-2022-3-13

UDC 004.412:519.237.5

NONLINEAR REGRESSION MODELS FOR ESTIMATING THE DURATION OF SOFTWARE DEVELOPMENT IN JAVA FOR PC BASED ON THE 2021 ISBSG DATA

Prykhodko S. B. – Dr. Sc., Professor, Head of the Department of Software of Automated Systems, Admiral Makarov National University of Shipbuilding, Mykolaiv, Ukraine.

Pukhalevych A. V. – PhD, Lecturer of the Department of Software of Automated Systems, Admiral Makarov National University of Shipbuilding, Mykolaiv, Ukraine.

Prykhodko K. S. – PhD, Associate Professor of the Department of Information Systems and Technologies, Admiral Makarov National University of Shipbuilding, Mykolaiv, Ukraine.

Makarova L. M. – PhD, Associate Professor, Associate Professor of the Department of Software of Automated Systems, Admiral Makarov National University of Shipbuilding, Mykolaiv, Ukraine.

ABSTRACT

Context. The problem of estimating the duration of software development in Java for personal computers (PC) is important because, first, failed duration estimating is often the main contributor to failed software projects, second, Java is a popular language, and, third, a personal computer is a widespread multi-purpose computer. The object of the study is the process of estimating the duration of software development in Java for PC. The subject of the study is the nonlinear regression models to estimate the duration of software development in Java for PC.

Objective. The goal of the work is to build nonlinear regression models for estimating the duration of software development in Java for PC based on the normalizing transformations and deleting outliers in data to increase the confidence of the estimation in comparison to the ISBSG model for the PC platform.

Method. The models, confidence, and prediction intervals of nonlinear regressions to estimate the duration of software development in Java for PC are constructed based on the normalizing transformations for non-Gaussian data with the help of appropriate techniques. The techniques to build the models, confidence, and prediction intervals of nonlinear regressions are based on normalizing transformations. Also, we apply outlier removal for model construction. In general, the above leads to a reduction of the mean magnitude of relative error, the widths of the confidence, and prediction intervals in comparison to nonlinear models constructed without outlier removal application in the model construction process.

Results. A comparison of the model based on the decimal logarithm transformation with the nonlinear regression models based on the Johnson (for the S_B family) and Box-Cox transformations as both univariate and bivariate ones has been performed.

Conclusions. The nonlinear regression model to estimate the duration of software development in Java for PC is constructed based on the decimal logarithm transformation. This model, in comparison with other nonlinear regression models, has smaller widths of the confidence and prediction intervals for effort values that are bigger than 900 person-hours. The prospects for further research may include the application of bivariate normalizing transformations and data sets to construct the nonlinear regression models for estimating the duration of software development in other languages for PC and other platforms, for example, mainframe.

KEYWORDS: duration, software development, Java, personal computer, nonlinear regression model, normalizing transformation, non-Gaussian data, ISBSG.

ABBREVIATIONS

COCOMO is a constructive cost model;

ISBSG is the International Software Benchmarking Standards Group;

KLOC is kilo lines of code (one thousand lines of code);

MMRE is a mean magnitude of relative error;

MRE is a magnitude of relative error;

PC is a personal computer;

PRED is a percentage of prediction;

SMD is a squared Mahalanobis distance.

NOMENCLATURE

$\hat{b}_0$ is an estimator of the parameter defined by the intercept of the true regression line for normalized data;

$\hat{b}_1$ is an estimator of the parameter defined by the slope of the true regression line for normalized data;

N is a number of data points;

$N\left(0,\sigma_\varepsilon^2\right)$ is a Gaussian distribution with zero mathematical expectation and variance σ_ε^2;

P is a non-Gaussian random vector;

R^2 is a multiple coefficient of determination;

T is a Gaussian random vector;

$t_{\alpha/2,N-2}$ is a quantile of student's t-distribution with $N-2$ degrees of freedom and $\alpha/2$ significance level;

X_1 is an effort of software development;

Y is the duration of software development;

Z_1 is a Gaussian variable that is obtained by transforming variable X_1;

Z_Y is a Gaussian variable that is obtained by transforming variable Y;

$\overline{Z}_Y$ is a sample mean of the Z_Y values;

$\hat{Z}_Y$ is a prediction result by linear regression equation for normalized data;

DOI 10.15588/1607-3274-2022-3-14

α is a significance level;

β_1 is a multivariate skewness;

β_2 is a multivariate kurtosis;

ε is a Gaussian random variable that defines residuals;

σ_ε is a standard deviation of ε;

$\boldsymbol{\psi}$ is a vector of bivariate normalizing transformation.

INTRODUCTION

Estimation of duration, effort, and the cost is a very important and integral part of the software development life cycle [1–3]. It is important to do an accurate estimation as much as possible because failed estimation (including duration estimation) is often the main contributor to failed software projects.

Today estimation of duration in software development is mostly based on heuristic approaches like expert judgment and planning poker. In absence of the experts for estimating, it is very difficult to estimate software development duration. That is why there is a need for algorithmic methods and mathematical models that can do accurate estimates.

For many years the most famous models are regression equations such as COCOMO and ISBSG. These models are similar in structure (both are effort dependent and constructed based on decimal logarithm). Wherein there only is one ISBSG model for estimating the duration of software development for the PC platform. However, there are no models which additionally take into account the programming language. In this paper, we demonstrated the need to take into account the programming language for the ISBSG model. We practiced the calibration of the ISBSG model using the ISBSG data set (D&E Corporate Release May 2021 R1) collected from the software development projects in Java for PC. We used the ISBSG data set because for many years the ISBSG repository is applied as a foundation of the software project estimation process [4]. Also, we constructed other models based on the normalizing transformations such as the Box-Cox and Johnson using the above data set.

The object of study is the process of estimating the duration of software development in Java for PC.

The subject of study is the regression models to estimate the duration of software development in Java for PC.

The purpose of the work is to increase the confidence in estimating the duration of software development in Java for PC.

1 PROBLEM STATEMENT

Suppose given the original sample as the bivariate non-Gaussian data set: actual duration (in months) Y and effort (in person-hours) X_1 of software development in Java for PC. Suppose that there is a mutually inverse normalizing transformation of non-Gaussian random vector $\mathbf{P}=\{Y, X_1\}^T$ to Gaussian random vector $\mathbf{T}=\{Z_Y, Z_1\}^T$ is given by

$$\mathbf{T}=\boldsymbol{\psi}(\mathbf{P}) \quad (1)$$

and the inverse transformation for (1)

$$\mathbf{P}=\boldsymbol{\psi}^{-1}(\mathbf{T}). \quad (2)$$

It is required to build the nonlinear regression model in the form $Y = Y(X_1, \varepsilon)$ based on transformations (1) and (2).

2 REVIEW OF THE LITERATURE

Although the first models for estimating the duration of software development were built in the 1970–1980 years [1, 5], research in this area is still ongoing [4, 6–10].

Most often these models enable estimating the duration of software development depending on the development effort. Building such models requires the presence of corresponding datasets. Firstly it was government organization datasets (NASA etc.). For at least 25 years many such researchers are used data from the different ISBSG repository releases [6–10].

The COCOMO models were built using project size as the data clustering criteria [1]. Software development projects were split by their size into 3 types: organic (2–50 KLOC), semi-detached (50–300 KLOC), and embedded (larger than 300 KLOC). Then each of these types was built in separate models.

The ISBSG models are similar in structure to COCOMO models the only difference is that they were built for such platforms as mainframe, mid-range, and personal computers based on the 1996 ISBSG repository data.

In all models from [1, 2, 6] the decimal logarithm transformation was used to normalize empirical data. But as it was clear from [6], the above transformation is not always acceptable for empirical data normalization. In [6] a linear regression was performed on the Log10-transformed values of duration and effort for the 39 PC software development projects $R^2 = 0.140$. It is very low and means that there is no correlation between dependent and independent variables.

In the nonlinear regression model for estimating the duration of software development for PC [7], the Johnson univariate transformation was used to normalize empirical data values of duration and effort. This transformation enables to build of valid models in some cases but as will be shown in this research this transformation gives average model quality with the 2021 ISBSG repository data for software developed in Java for PC. Therefore, it is also required to apply bivariate transformation and remove outliers from the empirical data to build a high-quality model according to [11].

DOI 10.15588/1607-3274-2022-3-14

A normalizing transformation is often a good way to construct nonlinear regression models [11–17]. According to [14], transformations are made for essentially four purposes, two of which are: firstly, to obtain approximate normality for the distribution of the error term (residuals), secondly, to transform the response and/or the predictor in such a way that the strength of the linear relationship between new variables (normalized variables) is better than the linear relationship between initial dependent and independent variables.

According to [11], there may be data sets on which the results of building nonlinear regression models depend, firstly, which normalizing transformation is used, univariate, or multivariate, and, secondly, are there any outliers in the data set. That is why in [11] the technique was considered to build nonlinear regression models based on the multivariate normalizing transformations and prediction intervals. In this technique, in addition to the technique for detecting outliers in multivariate non-Gaussian data [18], the prediction intervals of nonlinear regressions are used to detect the outliers in the process of constructing the nonlinear regression models. We apply the above technique [11] for building the nonlinear regression models with one predictor (effort) to estimate the duration of software development in Java for PC.

3 MATERIALS AND METHODS

According to [11], the technique to build nonlinear regression models based on the normalizing transformations and prediction intervals consists of four steps. In the first step, multivariate non-Gaussian data are normalized using a multivariate normalizing transformation (1).

In the second step, the nonlinear regression model is constructed based on the multivariate normalizing transformation (1). Before that, we first determine whether one data point of a multivariate non-Gaussian data set is a multidimensional outlier. To do this, we apply the statistical technique based on the normalizing transformations and the Mahalanobis squared distance (MSD) as in [18, 19]. If there is a two-dimensional outlier in a bivariate non-Gaussian data set, then we discard the one, and return to step 1, else build the linear regression model for normalized data based on the transformation (1) in the form

$$Z_Y = \hat{Z}_Y + \varepsilon = \hat{b}_0 + \hat{b}_1 Z_1 + \varepsilon , \qquad (3)$$

ε is a Gaussian random variable that defines residuals, $\varepsilon \sim N\left(0, \sigma_\varepsilon^2\right)$.

After that, the nonlinear regression model is built based on the linear regression model (3) and the transformations (1) and (2) as

$$Y = \psi_Y^{-1}\left(\hat{Z}_Y + \varepsilon\right) . \qquad (4)$$

In the third step, the prediction interval of nonlinear regression is defined [11]

$$\psi_Y^{-1}\left(\hat{Z}_{Y_i} \pm t_{\alpha/2,N-2} S_Z \sqrt{1 + \frac{1}{N} + \frac{\left(Z_{1_i} - \bar{Z}_1\right)^2}{S_{Z_1 Z_1}}}\right) , \qquad (5)$$

where $t_{\alpha/2,N-2}$ is a student's *t*-distribution quantile with $\alpha/2$ significance level and $N-2$ degrees of freedom; $S_Z^2 = \frac{1}{N-2}\sum_{i=1}^{N}\left(Z_{Y_i} - \hat{Z}_{Y_i}\right)^2$, $S_{Z_1 Z_1} = \sum_{i=1}^{N}\left(Z_{1_i} - \bar{Z}_1\right)^2$; $\bar{Z}_1 = \frac{1}{N}\sum_{i=1}^{N} Z_{1_i}$.

In the fourth step, we check if there are data that are out of the bounds of the prediction interval. And if we detect the outliers, we discard them and repeat all the steps starting with the first for new data without discarded outliers, else nonlinear regression model construction is completed.

To normalize the data according to (1), we applied the decimal logarithm transformation with components Z_1

$$Z_1 = \lg X_1 \qquad (6)$$

and Z_Y

$$Z_Y = \lg Y . \qquad (7)$$

Also, to normalize the data, we used the univariate and bivariate Box-Cox transformations [16] with components Z_1

$$Z_1 = x(\lambda_1) = \begin{cases} \left(X_1^{\lambda_1} - 1\right)/\lambda_1, & \text{if} \quad \lambda_1 \neq 0; \\ \ln(X_1), & \text{if} \quad \lambda_1 = 0 \end{cases} \qquad (8)$$

and Z_Y, which is defined analogously to (8) with the only difference that instead of Z_1, X_1, and λ_1 should be put respectively Z_Y, *Y*, and λ_Y. Here Z_1 and Z_Y are Gaussian variables, λ_1 and λ_Y parameters of the bivariate Box-Cox transformation.

Furthermore, to normalize the data, we used the univariate and bivariate Jonson transformations for the S_B family [11] with component Z_1

$$Z_1 = \gamma_1 + \eta_1 \ln \frac{X_1 - \varphi_1}{\varphi_1 + \lambda_1 - X_1} \qquad (9)$$

and Z_Y, which is defined analogously to (9) with the only difference that instead of Z_1, X_1, γ_1, η_1, φ_1, and λ_1 should be put respectively Z_Y, *Y*, γ_Y, η_Y, φ_Y, and λ_Y. Here Z_1 and Z_Y are Gaussian variables with zero mathematical expectation and unit variance; γ_Y, η_Y,

DOI 10.15588/1607-3274-2022-3-14

φ_Y, λ_Y, γ_1, η_1, φ_1, and λ_1 are parameters of the Johnson transformation for the S_B family.

The nonlinear regression model based on the linear regression model (3) for the normalized data and the decimal logarithm transformation for (6) and (7) has the form

$$Y = 10^{\varepsilon + \hat{b}_0} X_1^{\hat{b}_1} . \qquad (10)$$

The nonlinear regression model based on the bivariate Box-Cox transformation has the form [20]

$$Y = \left[\hat{\lambda}_Y \left(\hat{Z}_Y + \varepsilon\right) + 1\right]^{1/\hat{\lambda}_Y} . \qquad (11)$$

According to [20], the nonlinear regression model based on the Johnson bivariate transformation for the S_B family has the form

$$Y = \hat{\varphi}_Y + \hat{\lambda}_Y / \left\{1 + \exp\left[-\left(\hat{Z}_Y + \varepsilon - \hat{\gamma}_Y\right)/\hat{\eta}_Y\right]\right\}. \qquad (12)$$

In (10)–(12) as and in (3), ε is a Gaussian random variable which defines residuals, $\varepsilon \sim N\left(0, \sigma_\varepsilon^2\right)$.

The confidence interval of nonlinear regression is defined analogously to (5) with the only difference that in the sum under the square root, there will not be leading 1.

4 EXPERIMENTS

Before building a nonlinear regression model based on the multivariate normalizing transformation, we constructed a nonlinear regression equation to estimate the duration Y (in months) of software development for the PC platform depending on the effort X_1 (in person-hours) based on the decimal logarithm transformation of 243 software projects data with Data Quality Rating A from the 2021 ISBSG database (see Fig. 1).

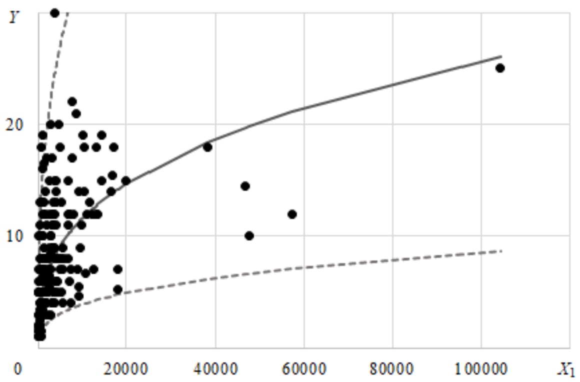


Figure 1 – Nonlinear regression (solid line) and its prediction intervals (dash lines) of the duration depending on the effort, which was constructed by the decimal logarithm using 243 software projects data (dots) with Quality Rating A (highest) from the ISBSG database (D&E Corporate Release May 2021 R1)

Fig. 1 contains nonlinear regression (solid line) for which the equation is:

$$Y = 0.4902 X_1^{0.3437} . \qquad (13)$$

Also, Fig. 1 contains prediction intervals bounds (dash lines) of nonlinear regression of the duration depending on the effort, which was constructed using the decimal logarithm (Log10) by (5) for a significance level of 0.05.

The values of R^2, MMRE, and PRED(0.25) equal respectively 0.2971, 0.4840, and 0.3457 for equation (13). These values are less than acceptable ones and indicate the unsatisfactory accuracy of duration prediction by equation (1). That is why we apply the appropriate technique [11] to build a nonlinear regression model for estimating the duration of software development in Java for PC.

To construct a nonlinear regression model for estimating the duration of software development in Java for PC we use the above technique for the 39 software projects data with Data Quality Rating A from the ISBSG database (D&E Corporate Release May 2021 R1). The above data are shown in Fig. 2 as dots.

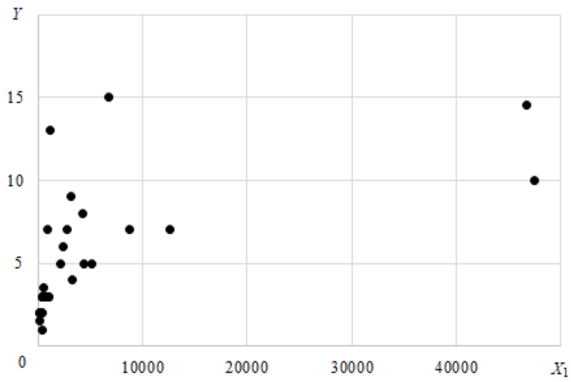


Figure 2 – Scatter plot of effort X_1 vs. duration Y for 39 software projects in Java for PC

We checked the bivariate data from Fig. 2 for multivariate outliers. But before that, we tested the normality of multivariate data from Fig. 2 because well-known statistical methods (for example, multivariate outlier detection based on the squared Mahalanobis distance (SMD)) are used to detect outliers in multivariate data under the assumption that the data is described by a multivariate Gaussian distribution [16, 18, 19]. We applied a multivariate normality test proposed by Mardia and based on measures of the multivariate skewness β_1 and kurtosis β_2 [21, 22]. According to this test, the distribution of bivariate data from Fig. 2 is not Gaussian since the test statistic for multivariate skewness $N\beta_1/6$ of this data, which equals 119.61, is greater than the quantile of the Chi-Square distribution, which is 14.86 for 4 degrees of freedom and 0.005 significance level. Similarly, the test statistic for multivariate kurtosis β_2, which equals 24.60,

DOI 10.15588/1607-3274-2022-3-14

is greater than the value of the Gaussian distribution quantile, which is 11.30 for 8 mean, 1.641 variance, and 0.005 significance level.

Therefore, we used the statistical technique [18] to detect multivariate outliers in the bivariate non-Gaussian data from Fig. 2 based on the multivariate normalizing transformations and the SMD for normalized data. To normalize the data from Fig. 2, we applied three univariate and two bivariate transformations (see Table 1).

The parameter estimates of the univariate and bivariate Box-Cox transformations for the data from Fig. 2 are calculated by the maximum likelihood method according to [16]. The parameter estimates of the univariate Box-Cox transformation are $\hat{\lambda}_Y = -0.295079$ and $\hat{\lambda}_1 = -0.244234$. The parameter estimates of the bivariate Box-Cox transformation are $\hat{\lambda}_Y = -0.207256$ and $\hat{\lambda}_1 = -0.212036$.

Table 1 – SMD values for normalized data

No	Project ID	Univariate			Bivariate	
		Log10	Box-Cox	Johnson	Box-Cox	Johnson
1	10248	1.37	1.55	1.04	1.60	1.27
2	10868	8.74	**14.69**	**14.02**	**12.95**	10.56
3	11641	**11.81**	8.18	8.20	9.10	10.23
4	11802	1.12	1.26	1.12	1.23	1.19
5	12636	5.60	3.98	4.78	4.42	5.30
6	12857	0.98	0.98	0.89	0.97	0.96
7	14345	0.43	0.39	0.26	0.40	0.31
8	14487	2.33	1.90	1.93	1.98	2.20
9	14883	0.50	0.49	0.31	0.49	0.38
10	14937	6.61	4.09	5.57	4.39	5.73
11	14953	0.52	0.61	0.51	0.59	0.53
12	16032	1.24	1.61	1.23	1.57	1.30
13	18271	0.15	0.13	0.12	0.15	0.16
14	19256	0.73	0.85	0.49	0.83	0.59
15	21372	0.42	0.35	0.21	0.34	0.27
16	21719	0.29	0.21	0.16	0.22	0.20
17	22359	1.86	1.92	1.36	2.01	1.68
18	22404	3.43	2.63	2.52	2.77	2.91
19	23094	1.11	1.28	1.05	1.26	1.11
20	23265	0.28	0.20	0.16	0.21	0.20
21	24483	2.23	2.01	1.76	2.07	1.98
22	25081	0.16	0.09	0.10	0.11	0.13
23	25342	1.54	1.86	1.13	1.93	1.47
24	25480	0.16	0.09	0.10	0.11	0.13
25	25663	1.49	2.26	1.66	2.17	1.76
26	25931	0.41	0.63	0.37	0.65	0.50
27	26422	0.25	0.17	0.14	0.18	0.17
28	26695	2.17	2.77	2.46	2.62	2.34
29	28504	0.28	0.42	0.26	0.43	0.34
30	28519	2.73	5.90	9.75	5.45	6.30
31	29310	7.92	4.10	6.06	4.45	6.44
32	29311	1.37	1.37	1.27	1.38	1.35
33	29398	1.83	3.18	2.49	3.00	2.52
34	29471	0.14	0.12	0.11	0.13	0.14
35	29537	0.23	0.43	0.27	0.41	0.27
36	30243	0.40	0.61	0.36	0.63	0.48
37	30658	0.18	0.24	0.17	0.25	0.22
38	31895	3.86	3.17	2.57	3.31	3.26
39	31999	1.11	1.28	1.05	1.26	1.11

The parameter estimates of the univariate and bivariate Jonson transformation for the S_B family for the data from Fig. 2 are calculated by the maximum likelihood method according to [20]. The parameter estimates of the univariate Jonson transformation for the S_B family are $\hat{\gamma}_Y = 3.8025$, $\hat{\gamma}_1 = 3.02479$, $\hat{\eta}_Y = 1.4727$, $\hat{\eta}_1 = 0.59352$, $\hat{\varphi}_Y = 0.65925$, $\hat{\varphi}_1 = 97.897$, $\hat{\lambda}_Y = 76.853$, and $\hat{\lambda}_1 = 207294.1$. The parameter estimates of the bivariate Jonson transformation for the S_B family are $\hat{\gamma}_Y = 10.939$, $\hat{\gamma}_1 = 4.42142$, $\hat{\eta}_Y = 1.19033$, $\hat{\eta}_1 = 0.54626$, $\hat{\varphi}_Y = 0.23642$, $\hat{\varphi}_1 = 81.576$, $\hat{\lambda}_Y = 5937.939$, and $\hat{\lambda}_1 = 1484909.7$.

Table 1 contains the SMD for normalized data. The SMD values from Table 1 indicate there is one multivariate outlier in bivariate non-Gaussian data for four transformations (all univariate transformations and the bivariate Box-Cox transformation) since the SMD values for row 3 for decimal logarithm and row 2 for two univariate transformations (Box-Cox and Jonson) and the bivariate Box-Cox transformation are greater than the quantile of the Chi-Square distribution, which equals to 10.60 for the 0.005 significance level and 2 degrees of freedom. In Table 1, the SMD values, which are greater than the above quantile, are highlighted in bold.

For example, a scatter plot of normalized effort Z_1 vs. normalized duration Z_Y (using the bivariate Box-Cox transformation) for the data from Fig. 2 is shown in Fig. 3. Here the above outlier (Project 10868) is marked as an “outlier”.

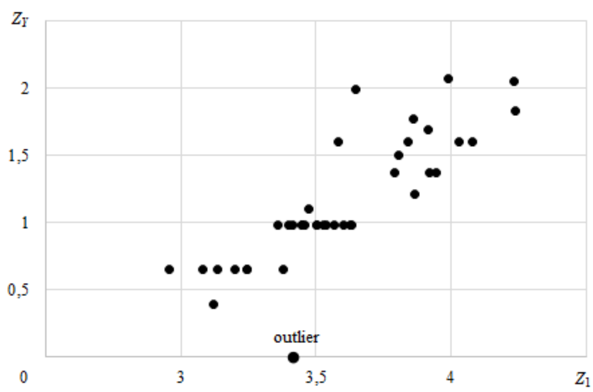


Figure 3 – Scatter plot of normalized effort Z_1 vs. normalized duration Z_Y (using the bivariate Box-Cox transformation) for the data from Fig. 2

Only the SMD values from Table 1 for bivariate Jonson transformations for the S_B family indicate there are no multivariate outliers in bivariate non-Gaussian data from Fig. 2 since all SMD values, in this case, are less than the above quantile value.

The reason for such different results in outlier detection is that only the data normalized using the bivariate Jonson transformation for the S_B family passes a multivariate normality test proposed by Mardia [21]. As a note, the above, Mardia’s test is based on measures of the multivariate skewness β_1 and kurtosis β_2 [21].

DOI 10.15588/1607-3274-2022-3-14

According to Mardia's test, the bivariate distribution of data (from Fig. 2) normalized using the bivariate Jonson transformation for the S_B family is approximately Gaussian since the test statistic for multivariate skewness $N\beta_1/6$ of this data, which equals 2.08, is less than the quantile of the Chi-Square distribution, which is 14.86 for 4 degrees of freedom and 0.005 significance level. Also, the test statistic for multivariate kurtosis β_2, which equals 10.71, is less than the value of the Gaussian distribution quantile, which is 11.30 for 8 mean, 1.641 variance, and 0.005 significance level.

Therefore, we decide, that there are no multivariate outliers in bivariate non-Gaussian data from Fig. 2 (39 data points). And we go to step 2 of the first iteration.

We constructed the nonlinear regression model (10), for which the estimate $\hat{\sigma}_\varepsilon$ is 0.1552, parameters estimates are $\hat{b}_0 = -0.500291$ and $\hat{b}_1 = 0.357533$.

Next, we calculated the nonlinear regression prediction interval by (5) for a significance level of 0.05. In the first iteration, $t_{\alpha/2,N-2} = 2.026$; $S_Z = 0.15726$; $\overline{Z}_1 = 3.025$; $S_{Z_1Z_1} = 15.968$ for the data normalized by the Log10 transformation of 39 data points from Fig. 2.

There are two outliers (data for software projects 10868 and 11641) since their Y values are out of the prediction interval computed by (5) for a significance level of 0.05. We discarded data of software projects 10868 and 11641. The first iteration is completed. The above 37 data points are shown in Fig. 4.

In the second iteration, there are no multivariate outliers in bivariate non-Gaussian data from Fig. 4 (37 data points). And we go to step 2 of the second iteration.

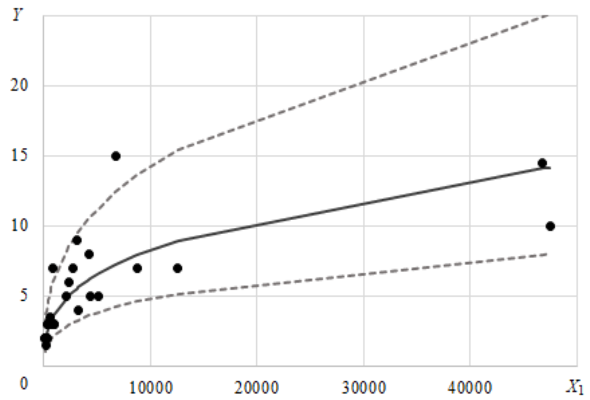


Figure 4 – Nonlinear regression $\hat{Y}$ (solid line) and its prediction intervals (dash lines) of the duration depending on the effort, which is constructed by the decimal logarithm transformation of 37 data points

We constructed the nonlinear regression model (10), for which the estimate $\hat{\sigma}_\varepsilon$ is 0.1104, parameters estimates are $\hat{b}_0 = -0.468116$ and $\hat{b}_1 = 0.346194$.

Next, we calculated the nonlinear regression prediction interval by (5) for a significance level of 0.05. In the second iteration, $t_{\alpha/2,N-2} = 2.030$; $S_Z = 0.1120$; $\overline{Z}_1 = 3.035$; $S_{Z_1Z_1} = 15.816$ for the data normalized by the Log10 transformation of 37 data points from Fig. 4.

There are two outliers (data for software projects 12636 and 31895) since their Y values are out of the prediction interval computed by (5) for a significance level of 0.05. We discarded data from software projects 12636 and 31895. The second iteration is completed.

In the third iteration, we used data from the remaining 35 projects (see Fig. 5). There are no multivariate outliers in bivariate non-Gaussian data from Fig. 5 (35 data points). And we go to step 2 of the third iteration.

Next, we used 35 data points from Fig. 5 to construct the model in form (10) with the following parameters estimates: $\hat{b}_0 = -0.439718$, $\hat{b}_1 = 0.330913$, $\hat{\sigma}_\varepsilon = 0.0828$.

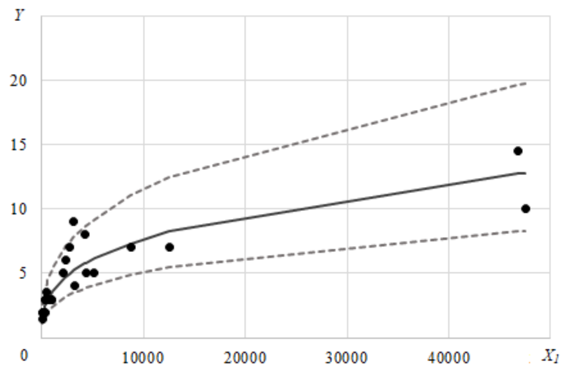


Figure 5 – Nonlinear regression $\hat{Y}$ (solid line) and its prediction intervals (dash lines) of the duration depending on the effort, which is constructed by the decimal logarithm transformation of 35 data points

After constructing a model (10), we have to find the nonlinear regression prediction interval by (5) for a significance level of 0.05 (see Fig. 5). In the third iteration, $t_{\alpha/2,N-2} = 2.035$; $S_Z = 0.0840$; $\overline{Z}_1 = 3.016$; $S_{Z_1Z_1} = 15.158$ for the data normalized by the Log10 transformation of 35 data points from Fig. 5.

There is one outlier (data for software project 14487) since its Y value is out of the prediction interval computed by (5) for a significance level of 0.05. We discarded data from software project 14487. The third iteration is completed.

In the fourth iteration, we used data from the remaining 34 projects (see Fig. 6). There are no multivariate outliers in bivariate non-Gaussian data from Fig. 6 (34 data points). And we go to step 2 of the fourth iteration.

We used 34 data points from Fig. 6 to construct the model in form (10) with the following parameters estimates: $\hat{b}_0 = -0.423179$, $\hat{b}_1 = 0.323072$, $\hat{\sigma}_\varepsilon = 0.07253$.

Next, we calculated the nonlinear regression prediction interval by (5) for a significance level of 0.05. In the fourth iteration, $t_{\alpha/2,N-2} = 2.037$; $S_Z = 0.07366$;

DOI 10.15588/1607-3274-2022-3-14

$\overline{Z}_1 = 3.002$; $S_{Z_1 Z_1} = 14.923$ for the data normalized by the Log10 transformation of 34 data points from Fig. 6.

There are no outliers since all Y values are not out of the bounds of the prediction interval computed by (5) for a significance level of 0.05. The model construction is completed.

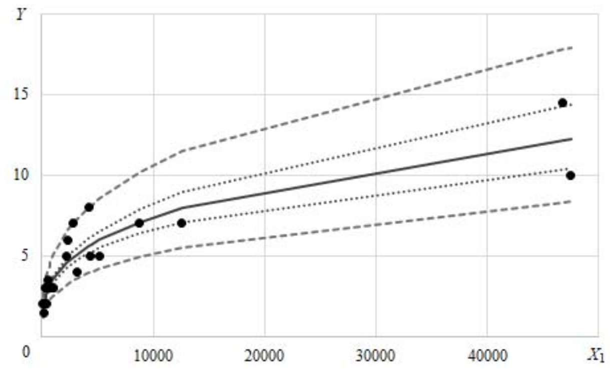


Figure 6 – Nonlinear regression $\hat{Y}$ (solid line), its confidence (dot lines) and prediction (dash lines) intervals of the duration depending on the effort, which is constructed by the decimal logarithm transformation of 34 data points

Also, we calculated the confidence intervals of nonlinear regression $\hat{Y}$ constructed by the decimal logarithm transformation of 34 data points (see Fig. 6).

The computer program implementing the constructed models (10), (11), and (12) was developed to conduct experiments. The program was written in the sci-language for the Scilab system. Scilab (https://www.scilab.org/) is free and open-source software, the alternative to commercial packages for system modeling and simulation packages such as MATLAB and MATRIXx [23].

5 RESULTS

The prediction results $\hat{Y}$ (solid line) of nonlinear regression models (11) and (12), its confidence (dot lines) and prediction (dash lines) intervals of the duration (in months) depending on the effort (in person-hours) are defined for both univariate and multivariate transformations (see figures 7–10) to compare with prediction results for model (10).

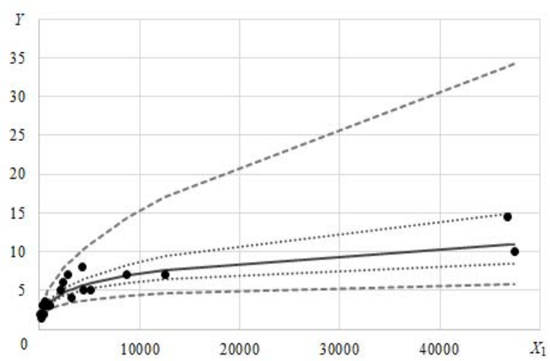


Figure 7 – Nonlinear regression $\hat{Y}$ (solid line), its confidence (dot lines) and prediction (dash lines) intervals of the duration depending on the effort, which is constructed by univariate Box-Cox' transformation of 34 data points

To evaluate the prediction accuracy of the nonlinear regression models we applied the metrics R^2, MMRE, and PRED(0.25). MMRE and PRED(0.25) are accepted as standard evaluations of prediction results by regression models.

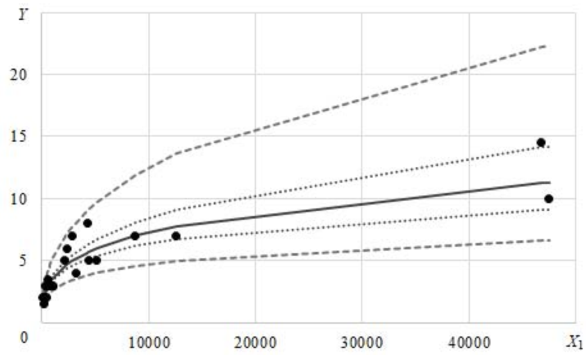


Figure 8 – Nonlinear regression $\hat{Y}$ (solid line), its confidence (dot lines) and prediction (dash lines) intervals of the duration depending on the effort, which is constructed by bivariate Box-Cox' transformation of 34 data points

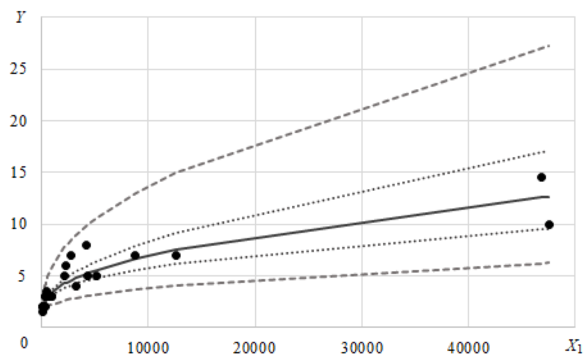


Figure 9 – Nonlinear regression $\hat{Y}$ (solid line), its confidence (dot lines) and prediction (dash lines) intervals of the duration depending on the effort, which is constructed by univariate Johnson' transformation of 34 data points

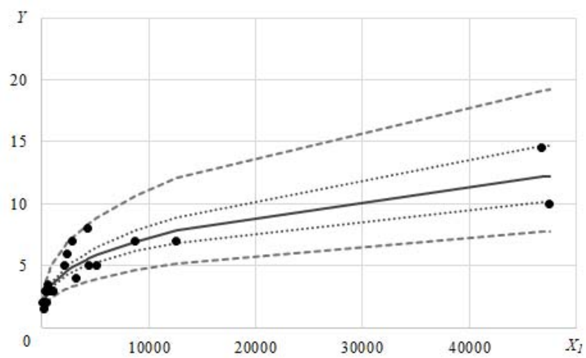


Figure 10 – Nonlinear regression $\hat{Y}$ (solid line), its confidence (dot lines) and prediction (dash lines) intervals of the duration depending on the effort, which constructed by bivariate Johnson' transformation of 34 data points

DOI 10.15588/1607-3274-2022-3-14

These metrics are applied in software engineering too [24, 25]. The acceptable values of MMRE and PRED(0.25) are not more than 0.25 and not less than 0.75 respectively. The values of R^2, MMRE and PRED(0.25) are shown in Table 2 for models (10)–(12) for both univariate and multivariate transformations. The values of these metrics are acceptable and approximately the same for all models. These values indicate good prediction accuracy of the models (10)–(12) for estimating the duration of software development in Java for PC.

Table 2 – The prediction accuracy metrics of the nonlinear regression models

Metrics	Univariate			Bivariate	
	Log10	Box-Cox	Johnson	Box-Cox	Johnson
R^2	0.8817	0.8704	0.8763	0.8709	0.8807
MMR_{min}	0.0010	0.0008	0.0006	0.0086	0.0001
MMR_{max}	0.3147	0.3001	0.2953	0.3539	0.3108
MMRE	0.1352	0.1333	0.1346	0.1356	0.1353
PRED(0.25)	0.8529	0.8529	0.8529	0.8235	0.8529

Also, Table 2 contains minimum and maximum values of MRE denoted MMR_{min} and MMR_{max}, respectively.

The model (12) based on the Johnson bivariate transformation for the S_B family has smaller MRE values for bigger numbers of data points in comparison to other models. Such, the MRE values for the model (12) based on the Johnson bivariate transformation are smaller than for the model (11) with parameters estimates for both the univariate and bivariate Box-Cox transformations for 21 from 34 data points. The MRE values for the model (12) based on the Johnson bivariate transformation are smaller than for the model (10) for 18 from 34 data points. The MRE values for the model (12) based on the Johnson bivariate transformation are smaller than for the model (12) with parameter estimates for the Johnson univariate transformation for 19 from 34 data points. Also, the last result indicates the advantage of using the bivariate transformation in comparison to the univariate one.

6 DISCUSSION

We apply bivariate normalizing transformations to build the nonlinear regression model for estimating the duration of software development in Java for PC by appropriate techniques [11] since the error distribution of the linear regression model is not Gaussian what the chi-squared test result indicates. Also, there are no outliers in the data. Moreover, the bivariate distribution of the data is not Gaussian which the Mardia multivariate normality test based on measures of the multivariate skewness and kurtosis indicates. Because we use the statistical technique [18] to detect multivariate outliers in the bivariate non-Gaussian data based on the bivariate normalizing transformations and the SMD for normalized data. Note, that we have other bivariate outliers for the data from Table 1 without applying normalization compared to outlier detection results using the above technique [18].

Also note that in our case, the poor normalization of bivariate non-Gaussian data using the Box-Cox and Johnson univariate transformations lead to an increase in the widths of the confidence and prediction intervals of nonlinear regression for a larger number of data rows compared to the Box-Cox and Johnson bivariate transformations. The above indicates the advantage of using the bivariate transformation in comparison to the univariate one.

The nonlinear regression model (10), in comparison with other nonlinear regression models (11) and (12), has smaller widths of the confidence and prediction intervals for effort values that are bigger than 900 person-hours. These results and the values of the prediction accuracy metrics from Table 2 indicate the preference for using a more simple model (10) for estimating the duration of software development in Java for PC.

CONCLUSIONS

The important problem of increase of confidence in estimating the duration of software development in Java for PC is solved.

The scientific novelty of obtained results is that nonlinear regression models to estimate the duration of software development in Java for PC are firstly constructed based on the Box-Cox and Johnson bivariate transformations. These models, in comparison with other nonlinear regression models, have smaller widths of the confidence and prediction intervals for effort values that are smaller than 900 person-hours.

The practical significance of obtained results is that the software realizing the constructed model is developed in the sci-language for Scilab. The experimental results allow for the recommendation of the constructed model for use in practice.

Prospects for further research may include the application of bivariate normalizing transformations and data sets to construct the nonlinear regression models for estimating the duration of software development in other languages for PC and other platforms, for example, mainframe.

ACKNOWLEDGEMENTS

This research was made possible by the ISBSG data set (D&E Corporate Release May 2021 R1). ISBSG (www.isbsg.org) provided the repository data subscription for Admiral Makarov National University of Shipbuilding at a heavily discounted price for academic purposes.

REFERENCES

1. Boehm B. W. Software engineering economics. Englewood Cliffs, NJ, Prentice Hall, 1981, 768 p.
2. Boehm B. W., Abts C., Brown A. W. et al. Software cost estimation with COCOMO II. Upper Saddle River, NJ: Prentice Hall PTR, 2000, 506 p.
3. Owais M., Ramakishore R. Effort, duration and cost estimation in agile software development, *2016 Ninth International Conference on Contemporary Computing (IC3),* 2016, pp. 1–5, DOI: 10.1109/IC3.2016.7880216.
4. Abran A. Data collection and industry standards: the ISBSG repository, *Software Project Estimation: The Fundamentals for Providing High Quality Information to Decision Makers,*

DOI 10.15588/1607-3274-2022-3-14

IEEE, 2015, pp. 161–184, DOI: 10.1002/9781118959312.ch8.

5. Putnam L. H. A general empirical solution to the macro-software sizing and estimating problem, *IEEE Transactions on Software Engineering,* 1978, Vol. 4, No. 2, July, pp. 345–361.
6. Oligny S., Bourque P. , Abran A. , Fournier B. Exploring the relation between effort and duration in software engineering projects, *Proceedings of the World Computer Congress,* Aug. 2000, P. 175–178.
7. Prykhodko S. B., Pukhalevich A. V. Developing PC Software Project Duration Model based on Johnson transformation, *Proceedings of the 12th International Conference Modern Problems of Radio Engineering, Telecommunications and Computer Science TCSET'2014, Lviv-Slavske, Ukraine*. Lviv, Polytechnic National University, 2014, pp. 114–116.
8. Prykhodko S.B., Pukhalevich A. V. Confidence interval estimation of PC software project duration regression based on Johnson transformation, *Radioelectronic and Computer Systems*. Kharkiv, 2014, No. 2 (66), pp. 104–107. ISSN: 1814-4225
9. López-Martín C., Abran A. Neural networks for predicting the duration of new software projects, *Journal of Systems and Software,* 2015, Vol. 101, pp. 127–135. DOI: 10.1016/J.JSS.2014.12.002
10. Pospieszny P., Czarnacka-Chrobot B., Kobylinski A. An effective approach for software project effort and duration estimation with machine learning algorithms, *Journal of Systems and Software,* 2018, pp. 184–196. DOI: 10.1016/J.JSS.2017.11.066
11. Prykhodko S., Prykhodko N. Mathematical modeling of non-Gaussian dependent random variables by nonlinear regression models based on the multivariate normalizing transformations, *Mathematical Modeling and Simulation of Systems : 15th International Scientific-practical Conference MODS'2020*. Chernihiv, Ukraine, June 29 – July 01, 2020, selected papers. Springer, Cham, 2021, P. 166–174. (Advances in Intelligent Systems and Computing, Vol. 1265). DOI: 10.1007/978-3-030-58124-4_16
12. Bates D. M., Watts D. G. Nonlinear regression analysis and its applications. New York, John Wiley & Sons, 1988, 384 p. DOI:10.1002/9780470316757
13. Seber G.A.F., C. J. Wild Nonlinear regression. New York, John Wiley & Sons, 1989, 768 p. DOI: 10.1002/0471725315
14. Ryan T. P. Modern regression methods. New York, John Wiley & Sons, 1997, 529 p. DOI: 10.1002/9780470382806
15. Drapper N. R., Smith H. Applied regression analysis. New York, John Wiley & Sons, 1998, 736 p.
16. Johnson R. A., Wichern D. W. Applied multivariate statistical analysis, Pearson Prentice Hall, 2007, 800 p.
17. Chatterjee S., Simonoff J. S. Handbook of regression analysis. New York, John Wiley & Sons, 2013, 236 p. DOI: 10.1002/9781118532843
18. Prykhodko S., Prykhodko N., Makarova L., et al. Detecting Outliers in Multivariate Non-Gaussian Data on the basis of Normalizing Transformations, *Electrical and Computer Engineering : the 2017 IEEE First Ukraine Conference (UKRCON) «Celebrating 25 Years of IEEE Ukraine Section», Kyiv, Ukraine, May 29 – June 2, 2017 : proceedings.* Kyiv, IEEE, 2017, pp. 846–849. DOI: 10.1109/UKRCON.2017.8100366
19. Prykhodko S., Prykhodko N., Makarova L. et al. Application of the Squared Mahalanobis Distance for Detecting Outliers in Multivariate Non-Gaussian Data, *Radioelectronics, Telecommunications and Computer Engineering : 14th International Conference on Advanced Trends (TCSET).* Lviv-Slavske, Ukraine, February 20–24, 2018 : proceedings, pp. 962–965. DOI: 10.1109/TCSET.2018.8336353
20. Prykhodko S., Prykhodko N., Knyrik K. Estimating the efforts of mobile application development in the planning phase using nonlinear regression analysis, *Applied Computer Systems*, 2020, Vol. 25, No. 2, pp. 172–179. DOI: 10.2478/acss-2020-0019
21. Mardia K.V. Measures of multivariate skewness and kurtosis with applications, *Biometrika,* 1970, Vol. 57, pp. 519–530. DOI: 10.1093/biomet/57.3.519
22. Mardia K.V. Applications of some measures of multivariate skewness and kurtosis in testing normality and robustness studies, *Sankhya: The Indian Journal of Statistics, Series B (1960–2002),* 1974, Vol. 36, Issue 2, pp. 115–128.
23. Campbell S. L., Chancelier J.-P., Nikoukhah R.Modeling and simulation in Scilab/Scicos. Springer, 2005, 313 p.
24. Foss T., Stensrud E., Kitchenham B., Myrtveit I. A simulation study of the model evaluation criterion MMRE, *IEEE Transactions on software engineering,* 2003, Vol. 29, Issue 11, pp. 985–995. DOI: 10.1109/TSE.2003.1245300
25. Port D., Korte M. Comparative studies of the model evaluation criterions MMRE and PRED in software cost estimation research, *Empirical Software Engineering and Measurement, the 2nd ACM-IEEE International Symposium ESEM, Kaiserslautern, Germany, October, 2008 : proceedings.* New York, ACM, 2008, pp. 51–60

Received 01.07.2022.
Accepted 27.08.2022.

УДК 004.412:519.237.5

НЕЛІНІЙНІ РЕГРЕСІЙНІ МОДЕЛІ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ НА JAVA ДЛЯ ПК ЗА ДАНИМИ ISBSG 2021 РОКУ

Приходько С. Б. – д-р техн. наук, професор, завідувач кафедри програмного забезпечення автоматизованих систем Національного університету кораблебудування імені адмірала Макарова, Миколаїв, Україна.

Пухалевич А. В. – канд. техн. наук, викладач кафедри програмного забезпечення автоматизованих систем Національного університету кораблебудування імені адмірала Макарова, Миколаїв, Україна.

Приходько К. С. – канд. техн. наук, доцент кафедри інформаційних систем і технологій Національного університету кораблебудування імені адмірала Макарова, Миколаїв, Україна.

Макарова Л. М. – канд. техн. наук, доцент кафедри програмного забезпечення автоматизованих систем Національного університету кораблебудування імені адмірала Макарова, Миколаїв, Україна.

DOI 10.15588/1607-3274-2022-3-14

АНОТАЦІЯ

Актуальність проблеми оцінювання тривалості розробки програмного забезпечення (ПЗ) на Java для персональних комп'ютерів (ПК) обумовлена наступним чинниками: по-перше, невдале оцінювання тривалості часто є основною причиною невдалої реалізації програмних проектів; по-друге, Java є популярною мовою; і, по-третє, ПК є широко поширеним багатоцільовим комп'ютером. Об'єктом дослідження є процес оцінювання тривалості розробки ПЗ на Java для ПК. Предметом дослідження є моделі нелінійної регресії для оцінювання тривалості розробки ПЗ на Java для ПК.

Мета. Метою роботи є побудова нелінійних регресійних моделей для оцінювання тривалості розробки ПЗ в Java для ПК на основі нормалізуючого перетворення у вигляді десяткового логарифму та видалення викидів у даних для підвищення достовірності оцінювання порівняно з моделлю ISBSG. для платформи ПК.

Метод. За допомогою відповідних методів на основі нормалізуючих перетворень для негаусових даних побудовано моделі, довірчі інтервали та інтервали прогнозування нелінійних регресій для оцінки тривалості розробки ПЗ на Java для ПК. Методи побудови моделей, довірчих інтервалів та інтервалів прогнозування нелінійних регресій базуються на нормалізуючих перетвореннях. Також ми застосовуємо видалення викидів для побудови моделей. Загалом, вищезазначене призводить до зменшення середньої величини відносної похибки, ширини довірчих інтервалів та інтервалів прогнозування порівняно з нелінійними моделями, побудованими без застосування видалення викидів у процесі побудови моделей.

Результати. Проведено порівняння побудованої на основі десяткового логарифму моделі з моделями нелінійної регресії на основі перетворень Джонсона (для сімейства S_B) та Бокса-Кокса як одновимірних, так і двовимірних.

Висновки. Модель нелінійної регресії для оцінювання тривалості розробки ПЗ на Java для ПК побудована на основі перетворення десяткового логарифма. Ця модель, порівняно з іншими моделями нелінійної регресії, має менші значення ширини довірчих інтервалів та інтервалів прогнозування для трудовитрат, які перевищують 900 людино-годин. Перспективи подальших досліджень можуть передбачати застосування двовимірних нормалізуючих перетворень і наборів даних для побудови нелінійних регресійних моделей для оцінювання тривалості розробки ПЗ іншими мовами для ПК та інших платформ, наприклад, мейнфреймів.

КЛЮЧОВІ СЛОВА: тривалість, розробка програмного забезпечення, Java, персональний комп'ютер, нелінійна регресійна модель, нормалізуюче перетворення, негаусові дані, ISBSG.

УДК 004.412:519.237.5

НЕЛИНЕЙНЫЕ РЕГРЕССИОННЫЕ МОДЕЛИ ДЛЯ ОЦЕНИВАНИЯ ПРОДОЛЖЕННОСТИ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ НА JAVA ДЛЯ ПК ПО ДАННЫМ ISBSG 2021 года

Приходько С. Б. – д-р техн. наук, профессор, заведующий кафедрой программного обеспечения автоматизированных систем Национального университету кораблестроения имени адмирала Макарова, Николаев, Украина.

Пухалевич А. В. – канд. техн. наук, преподаватель кафедры программного обеспечения автоматизированных систем Национального университету кораблестроения имени адмирала Макарова, Николаев, Украина.

Приходько Е. С. – канд. техн. наук, доцент кафедры информационных систем и технологий Национального университету кораблестроения имени адмирала Макарова, Николаев, Украина.

Макарова Л. Н. – канд. техн. наук, доцент кафедры программного обеспечения автоматизированных систем Национального университету кораблестроения имени адмирала Макарова, Николаев, Украина.

АННОТАЦИЯ

Актуальность проблемы оценивания продолжительности разработки программного обеспечения (ПО) на Java для персональных компьютеров (ПК) обусловлена следующими факторами: во-первых, неудачное оценивание продолжительности часто является основной причиной неудачной реализации программных проектов; во-вторых, Java является популярным языком; и, в-третьих, ПК является широко распространенным многоцелевым компьютером. Объектом исследования является процесс оценки продолжительности разработки программного обеспечения в Java для ПК. Предметом исследования являются модели нелинейной регрессии для оценки продолжительности разработки ПО на Java для ПК.

Цель. Целью работы является построение нелинейных регрессионных моделей для оценки продолжительности разработки ПО на Java для ПК с использованием нормализующего преобразования в виде десятичного логарифма и удаления выбросов в данных для повышения достоверности оценивания по сравнению с моделью ISBSG для платформы ПК.

Метод. С помощью соответствующих методов на основе нормализующих преобразований для негауссовых данных построена модель, доверительные интервалы и интервалы прогнозирования нелинейной регрессии для оценки продолжительности разработки ПО на Java для ПК. Методы построения моделей, доверительных интервалов и интервалов прогнозирования нелинейных регрессий базируются на нормализирующих преобразованиях. Также мы используем удаление выбросов для построения модели. В целом вышеупомянутое приводит к уменьшению средней величины относительной погрешности, ширины доверительных интервалов и интервалов прогнозирования по сравнению с нелинейными моделями, построенными без применения удаления выбросов в процессе построения модели.

Результаты. Произведено сравнение построенной на основі десятичного логарифма модели с моделями нелинейной регрессии на основе преобразований Джонсона (для семейства S_B) и Бокса-Кокса как одномерных, так и двумерных.

Выводы. Модель нелинейной регрессии для оценивания продолжительности разработки ПО на Java для ПК построена на основе преобразования десятичного логарифма. Эта модель, по сравнению с другими моделями нелинейной регрессии, имеет меньшие значения ширины доверительных интервалов и интервалов прогнозирования для трудозатрат, превышающих 900 человеко-часов. Перспективы дальнейших исследований могут предусматривать применение двумерных нормали-

DOI 10.15588/1607-3274-2022-3-14

зирующих преобразований и наборов данных для построения нелинейных регрессионных моделей для оценивания продолжительности разработки ПО на других языках для ПК и других платформ, например мейнфреймов.

КЛЮЧЕВЫЕ СЛОВА: длительность, разработка программного обеспечения, Java, персональный компьютер, нелинейная регрессионная модель, нормализующее преобразование, негауссовы данные, ISBSG.

ЛІТЕРАТУРА / ЛИТЕРАТУРА

1. Boehm B. W. Software engineering economics / B. W. Boehm. – Englewood Cliffs, NJ : Prentice Hall, 1981. – 768 p.
2. Boehm B.W. Software cost estimation with COCOMO II / [B. W. Boehm, C. Abts, A. W. Brown, et al.]. – Upper Saddle River, NJ: Prentice Hall PTR, 2000. – 506 p.
3. Owais M. Effort, duration and cost estimation in agile software development / M. Owais, R. Ramakishore // 2016 Ninth International Conference on Contemporary Computing (IC3), 2016. – P. 1–5, DOI: 10.1109/IC3.2016.7880216.
4. Abran A. Data collection and industry standards: the ISBSG repository / A. Abran // Software Project Estimation: The Fundamentals for Providing High Quality Information to Decision Makers, IEEE, 2015. – P. 161–184, DOI: 10.1002/9781118959312.ch8.
5. Putnam L. H. A general empirical solution to the macrosoftware sizing and estimating problem / L. H. Putnam // IEEE Transactions on Software Engineering. –1978 July. – Vol. 4, No. 2. – P. 345–361.
6. Exploring the relation between effort and duration in software engineering projects / [S. Oligny, P. Bourque, A. Abran, B. Fournier] // Proceedings of the World Computer Congress, Aug. 2000. – P. 175–178.
7. Prykhodko S. B. Developing PC Software Project Duration Model based on Johnson transformation / S. B. Prykhodko, A. V. Pukhalevich // Proceedings of the 12th International Conference Modern Problems of Radio Engineering, Telecommunications and Computer Science TCSET'2014, Lviv-Slavske, Ukraine. – Lviv : Polytechnic National University, 2014. – P. 114–116.
8. Prykhodko S.B. Confidence interval estimation of PC software project duration regression based on Johnson transformation / S. B. Prykhodko, A. V. Pukhalevich // Радіоелектронні і комп'ютерні системи. – Харків, 2014. – № 2 (66) – С. 104–107. – ISSN: 1814-4225
9. López-Martín C. Neural networks for predicting the duration of new software projects / C. López-Martín, A. Abran // Journal of Systems and Software. – 2015, Vol. 101. – P. 127–135. DOI: 10.1016/J.JSS.2014.12.002
10. Pospieszny P. An effective approach for software project effort and duration estimation with machine learning algorithms / P. Pospieszny, B. Czarnacka-Chrobot, A. Kobylinski // Journal of Systems and Software. – 2018. – P. 184–196. DOI: 10.1016/J.JSS.2017.11.066
11. Prykhodko S. Mathematical modeling of non-Gaussian dependent random variables by nonlinear regression models based on the multivariate normalizing transformations / S. Prykhodko, N. Prykhodko // Mathematical Modeling and Simulation of Systems : 15th International Scientific-practical Conference MODS'2020, Chernihiv, Ukraine, June 29 – July 01, 2020 : selected papers. – Springer, Cham., 2021. – P. 166–174. – (Advances in Intelligent Systems and Computing, Vol. 1265). DOI: 10.1007/978-3-030-58124-4_16
12. Bates D. M. Nonlinear regression analysis and its applications / D. M. Bates, D. G. Watts. – New York : John Wiley & Sons, 1988. – 384 p. DOI:10.1002/9780470316757
13. Seber G.A.F. Nonlinear regression / G.A.F. Seber, C. J. Wild. – New York : John Wiley & Sons, 1989. – 768 p. DOI: 10.1002/0471725315
14. Ryan T.P. Modern regression methods / T. P. Ryan. – New York : John Wiley & Sons, 1997. – 529 p. DOI: 10.1002/9780470382806
15. Drapper N. R. Applied regression analysis / N. R. Drapper, H. Smith. – New York : John Wiley & Sons, 1998. – 736 p.
16. Johnson R. A. Applied multivariate statistical analysis / R. A. Johnson, D. W. Wichern. – Pearson Prentice Hall, 2007. – 800 p.
17. Chatterjee S. Handbook of regression analysis / S. Chatterjee, J. S. Simonoff. – New York : John Wiley & Sons, 2013. – 236 p. DOI: 10.1002/9781118532843
18. Detecting Outliers in Multivariate Non-Gaussian Data on the basis of Normalizing Transformations / [S. Prykhodko, N. Prykhodko, L. Makarova, et al.] // Electrical and Computer Engineering : the 2017 IEEE First Ukraine Conference (UKRCON) «Celebrating 25 Years of IEEE Ukraine Section», Kyiv, Ukraine, May 29 – June 2, 2017 : proceedings. – Kyiv : IEEE, 2017. – P. 846–849. DOI: 10.1109/UKRCON.2017.8100366
19. Application of the Squared Mahalanobis Distance for Detecting Outliers in Multivariate Non-Gaussian Data / [S. Prykhodko, N. Prykhodko, L. Makarova, et al.] // Radioelectronics, Telecommunications and Computer Engineering : 14th International Conference on Advanced Trends (TCSET), Lviv-Slavske, Ukraine, February 20–24, 2018 : proceedings. – P. 962–965. DOI: 10.1109/TCSET.2018.8336353
20. Prykhodko S. Estimating the efforts of mobile application development in the planning phase using nonlinear regression analysis / S. Prykhodko, N. Prykhodko, K. Knyrik // Applied Computer Systems. – 2020. – Vol. 25, No. 2. – P. 172–179. DOI: 10.2478/acss-2020-0019
21. Mardia K. V. Measures of multivariate skewness and kurtosis with applications / K. V. Mardia // Biometrika. – 1970. – Vol. 57. – P. 519–530. DOI: 10.1093/biomet/57.3.519
22. Mardia K. V. Applications of some measures of multivariate skewness and kurtosis in testing normality and robustness studies / K. V. Mardia // Sankhya: The Indian Journal of Statistics, Series B (1960–2002). – 1974. – Vol. 36, Issue 2. – P. 115–128.
23. Campbell S. L. Modeling and simulation in Scilab/Scicos / S. L. Campbell, J.-P. Chancelier, R. Nikoukhah. – Springer, 2005. – 313 p.
24. Foss T. A simulation study of the model evaluation criterion MMRE / T. Foss, E. Stensrud, B. Kitchenham, I. Myrtveit // IEEE Transactions on software engineering. – 2003. – Vol. 29, Issue 11. – P. 985–995. DOI: 10.1109/TSE.2003.1245300
25. Port D. Comparative studies of the model evaluation criterions MMRE and PRED in software cost estimation research / D. Port, M. Korte // Empirical Software Engineering and Measurement : the 2nd ACM-IEEE International Symposium ESEM, Kaiserslautern, Germany, October, 2008 : proceedings. – New York : ACM, 2008. – P. 51–60.

DOI 10.15588/1607-3274-2022-3-14

УПРАВЛІННЯ У ТЕХНІЧНИХ СИСТЕМАХ

CONTROL IN TECHNICAL SYSTEMS

УПРАВЛЕНИЕ В ТЕХНИЧЕСКИХ СИСТЕМАХ

UDC 519.816, 681.518.2

METHOD FOR WEIGHTS CALCULATION BASED ON INTERVAL MULTIPLICATIVE PAIRWISE COMPARISON MATRIX IN DECISION-MAKING MODELS

Nedashkovskaya N. I. – Dr. Sc., Associate Professor at the Department of Mathematical Methods of System Analysis, Institute for Applied Systems Analysis at National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, Ukraine.

ABSTRACT

Context. The pairwise comparison method is a component of several decision support methodologies such as the analytic hierarchy and network processes (AHP, ANP), PROMETHEE, TOPSIS and other. This method results in the weight vector of elements of decision-making model and is based on inversely symmetrical pairwise comparison matrices. The evaluation of the elements is carried out mainly by experts under conditions of uncertainty. Therefore, modifications of this method have been explored in recent years, which are based on fuzzy and interval pairwise comparison matrices (IPCMs).

Objective. The purpose of the work is to develop a modified method for calculation of crisp weights based on consistent and inconsistent multiplicative IPCMs of elements of decision-making model.

Method. The proposed modified method is based on consistent and inconsistent multiplicative IPCMs, fuzzy preference programming and results in more reliable weights for the elements of decision-making model in comparison with other known methods. The differences between the proposed method and the known ones are as follows: coefficients that characterize extended intervals for ratios of weights are introduced; membership functions of fuzzy preference relations are proposed, which depend on values of IPCM elements. The introduction of these coefficients and membership functions made it possible to prove the statement about the required coincidence of the calculated weights based on the "upper" and "lower" models. The introduced coefficients can be further used to find the most inconsistent IPCM elements.

Results. Experiments were performed with several IPCMs of different consistency level. The weights on the basis of the considered consistent and weakly consistent IPCMs obtained using the proposed and other known methods have determined the same rankings of the compared objects. Therefore, the results using the proposed method on the basis of such IPCMs do not contradict the results obtained for these types of IPCMs using other known methods. Rankings by the proposed method based on the considered highly inconsistent IPCMs are much closer to rankings based on the corresponding initial undisturbed IPCMs in comparison with rankings obtained using the known FPP method. The most inconsistent elements in the considered IPCMs are found.

Conclusions. The developed method has shown its efficiency, results in more reliable weights and can be used for a wide range of decision support problems, scenario analysis, priority calculation, resource allocation, evaluation of decision alternatives and criteria in various application areas.

KEYWORDS: interval multiplicative pairwise comparison matrix, consistency, expert judgements, fuzzy preference programming, decision support systems.

ABBREVIATIONS

DM is a decision-making;
PCM is a pairwise comparison matrix;
IPCM is an interval pairwise comparison matrix;
AHP is an analytic hierarchy process;
FPP is a fuzzy preference programming.

NOMENCLATURE

A is an interval pairwise comparison matrix;
n is a dimension of matrix A, the number of pairwise compared objects;
a_{ij} is an interval number, an element of matrix A;
l_{ij} is a left end of interval a_{ij};
u_{ij} is a right end of interval a_{ij};

DOI 10.15588/1607-3274-2022-3-15

w is a weight vector;

w_i is an element of weight vector, $i = 1,...,n$;

P is a matrix describing a system of inequalities, $P \in R^{q \times n}$;

q is a number of rows in matrix P, $q = n(n-1)$;

P_k is the k-th row of matrix P;

$\mu_k(P_k w)$ is a membership function of a fuzzy preference relation;

d_k is a parameter in function $\mu_k(P_k w)$;

$\mu_{ij}(w)$ is a convex membership function of a fuzzy relation;

d_{ij} is a parameter in function $\mu_{ij}(w)$;

m_{ij} is the most probable predominance value from the interval $[l_{ij}, u_{ij}]$;

$\tilde{A}$ is a fuzzy feasible area;

T^{n-1} is a $(n-1)$-dimensional simplex;

w^* is a maximizing solution, the resulting vector of weights;

λ is an auxiliary variable;

$\lambda^* = \mu_{\tilde{A}}(w^*)$ is the degree of overall satisfaction of the decision maker with the optimal decision w^*;

λ_{ij}^* are coefficients, which characterize extended intervals for weights ratios.

INTRODUCTION

The problems of evaluating decision alternatives in semi-structured and unstructured subject areas have their own characteristics [1, 2]: the uniqueness, complexity, lack of optimality in the classical sense, multicriteria, data uncertainty, incompleteness of quantitative input information and the need to take into account qualitative judgements of a decision maker. Semi-structured and unstructured decision support problems are solved using the expert estimates and the principle of decomposition of a complex problem into subproblems [2–4]. Thus, a typical hierarchical or network model for a practical decision-making (DM) problem contains decision criteria, sub-criteria, goals, sub-goals and decision alternatives [4]. By performing pairwise comparisons of the criteria and alternatives, preference relations on the set of compared elements are built, and the pairwise comparison matrices (PCMs) are formed. Moreover, the pairwise comparison method represents a natural, general way of thinking of a person when making decisions [5, 6]. Based on PCMs, the weights of criteria and the priorities of alternatives in terms of each criterion are calculated. These priorities form partial solutions to the initial problem and are further aggregated into the resulting vector of global priorities or weights of the alternatives. In recent decades, considerable attention is given to development of the theory and applications of the Analytic Hierarchy Process (AHP) [7 – 10]. The rating, prioritization and resource allocation problems, problems of choosing the best decision alternatives, planning, scenario analysis and sustainable development problems are solved using the AHP.

One of the main elements of both the basic AHP and its modifications is the calculation of local priorities or weights using the pairwise comparison method and expert judgements. The complex psychological process of a comparative analysis made by a person in the presence of multiple criteria and decision alternatives is significantly influenced by various kinds of uncertainties [11]. The input information in the majority of multi-criteria DM problems is uncertaint and imprecise [12]. The uncertainty level of pairwise comparison results can increase with an increase of the number of compared elements [5, 13]. Therefore, many modifications of the AHP have been proposed for calculating weights based on uncertain estimates of pairwise comparisons: stochastic [14], fuzzy [15–35], or their combinations. Fuzzy multiplicative PCMs [15–18], interval multiplicative PCMs [19–29] and different types of fuzzy preference relations [30–35] on a set of pairwise compared alternatives are investigated. The weights are calculated on the basis of interval fuzzy preference relations [30–32] and interval multiplicative preference relations [18, 19, 34, 35].

The resulting local priorities (weights) of the decision criteria and alternatives on the basis of PCMs are significantly influenced by the consistency level of the initial expert pairwise comparison judgements. Methods for assessing the consistency level of PCMs are developed in [23, 26, 29, 31–35], methods for increasing the consistency level of PCMs are proposed and discussed in [27, 36], but this issue requires further research.

The object of study is an interval multiplicative PCM of elements of DM model.

The subject of study is a method for calculation of priorities (weights) of elements of DM model on the basis of interval multiplicative PCM and fuzzy preference programming.

The purpose of this work is to develop a modified method for calculation of crisp weights (priorities) based on consistent and inconsistent interval multiplicative PCM of elements of DM model.

1 PROBLEM STATEMENT

Let $A = \{a_{ij} = [l_{ij}, u_{ij}] \mid i, j = 1,...,n\}$ be an interval multiplicative pairwise comparison matrix (IPCM) of objects, for example, decision alternatives regarding their common characteristic (decision criterion), $0 < l_{ij} \le u_{ij}$, $l_{ij} = 1/u_{ji}$, $l_{ii} = u_{ii} = 1$. In is necessary to find the vector $w = \{w_i \mid i = 1,...,n\}$ of weights of the objects, such that $w_i \in R^+$, $\sum_{i=1}^{n} w_i = 1$.

DOI 10.15588/1607-3274-2022-3-15

2 REVIEW OF THE LITERATURE

The first modifications of the pairwise comparison method using fuzzy set theory appeared in the early 1980s [37, 38]. Using extended binary arithmetic operations, a method for calculating fuzzy weights was proposed based on the PCMs with triangular fuzzy numbers [37]. This approach was further developed in [15–17], where the approximation of PCM elements a_{ij} by the (L–R)-type fuzzy numbers $\tilde{a}_{ij}$ was performed. Fuzzy numbers $\tilde{a}_{ij}$ represent the approximate value of preference of one alternative over the other and form a fuzzy PCM $\tilde{A}$. On the basis of the fuzzy matrix $\tilde{A}$, fuzzy weights are calculated, which approximate $\tilde{a}_{ij}$: $\tilde{a}_{ij} \approx \tilde{w}_i / \tilde{w}_j$.

Another approach, presented in [18, 20–22, 39], is based on the discretization of fuzzy numbers – elements of fuzzy PCM. Decomposition representation of fuzzy PCM using level sets results in interval PCMs. Models of least logarithmic squares [39], FPP [18, 19], goal programming [22], lower and upper approximation models [20], two-stage models [21, 40] and other have been proposed, which operate with IPCMs.

The first stage of the two-stage model [21] consists in finding the minimum deviations of the expert IPCM from the unknown consistent PCM. In the second stage, the weight vector is directly calculated based on the founded deviations. Another two-stage method for weight vector calculation is based on the interval additive PCM [40]. At the first stage of this method, programming models are built to obtain crisp consistent PCMs based on inconsistent interval PCMs. At the second stage, the resulting weights are calculated with different degrees of confidence.

The FPP method for calculation of crisp weights is based on the IPCM and fuzzy mathematical programming [18]. The FPP method requires the setting of additional parameters, which characterize the level of satisfaction of an expert or a decision maker with the calculated vector of weights. The problem is transformed to a classical fuzzy programming problem using the Bellman-Zade principle. The FPP method [18, 19] has significant limitation, which casts doubt on the validity of obtained results. It is the sensitivity of results by the FPP method to the renumbering of the compared objects.

The analysis of modifications of the pairwise comparison method, analytical hierarchy and network methods, which use the fuzzy sets, has shown that in these methods little attention is given to methods for increasing the consistency level of expert pairwise comparison judgements represented by interval and fuzzy PCMs.

3 MATERIALS AND METHODS

IPCM A is inversely symmetric: $a_{ji} = 1/a_{ij}$, $\forall i, j = 1,...,n$. This is equivalent to the fulfillment of the condition $l_{ij} = 1/u_{ji}$ for $\forall i, j = 1,...,n$. Thus, the elements of the upper triangular and lower triangular parts of the IPCM carry the same information about the values of preference on a set of compared objects.

Definition. IPCM A is called consistent, if a weight vector w exists, which satisfies the conditions $w_i \in R^+$, $\sum_{i=1}^{n} w_i = 1$ [20, 21]:

$$l_{ij} \le w_i / w_j \le u_{ij}, \quad i = 1,2,...,n-1, \; j = 2,3,...,n, \; i < j. \tag{1}$$

A weaker concept of a fuzzy consistent IPCM is also used, such that violation of inequalities (1) is allowed to some extent. According to the FPP method, a vector of weights w is calculated that satisfies inequalities (1) approximately.

Definition. IPCM A is called fuzzy consistent if a weight vector w exists, such that $w_i \in R^+$, $\sum_{i=1}^{n} w_i = 1$ [18, 19]:

$$l_{ij} \lessapprox w_i / w_j \lessapprox u_{ij}, \quad i = 1,2,...,n-1, \; j = 2,3,...,n, \; i < j, \tag{2}$$

where $\lessapprox$ is a fuzzy preference relation.

Inequalities (2) are transformed in order to formulate a linear optimization problem for calculating the weight vector w:

$$w_i - u_{ij} w_j \lessapprox 0, \quad -w_i + l_{ij} w_j \lessapprox 0, \; i < j. \tag{3}$$

System (3), which contains $n(n-1)$ inequalities, can be written in the equivalent matrix form:

$$Pw \lessapprox 0, \tag{4}$$

where $P \in R^{q \times n}$, $q = n(n-1)$.

In the initial FPP method [18], the following piecewise continuous membership function was used, representing the kth row of the inequality (4), for which $P_k w \lessapprox 0$, $k = 1,2,...,q$:

$$\mu_k(P_k w) = \begin{cases} 1, & P_k w \le 0, \\ 1 - \dfrac{P_k w}{d_k}, & 0 < P_k w \le d_k, \\ 0, & P_k w > d_k; \end{cases} \tag{5}$$

where d_k is a parameter specifying the interval of approximate fulfillment of a crisp inequality $P_k w \le 0$, subscript k corresponds to the number of one-sided inequality in the constraint system (3).

DOI 10.15588/1607-3274-2022-3-15

The membership function $\mu_k(P_k w)$ (5) describes the degree of satisfaction of the decision maker with some weight vector, according to the *k*th one-sided inequality (3). According to (5), the $\mu_k(P_k w)$ value:

– is zero, if the corresponding crisp constraint $P_k w \le 0$ is strongly violated;

– grows linearly, is positive and less than one, if the constraint $P_k w \le 0$ is approximately satisfied;

– is equal to one, if the constraint $P_k w \le 0$ is fully satisfied.

Since both constraints in (3) correspond to the same interval for a_{ij} given by the pair (i, j), a modified FPP model is proposed [19], such that functions (5) are represented in the form of the following convex membership functions:

$$\mu_{ij}(w_i, w_j) = \begin{cases} 1 - \dfrac{(-w_i + l_{ij} w_j)}{d_{ij}}, \dfrac{w_i}{w_j} \le m_{ij}, \\ 1 - \dfrac{(w_i - u_{ij} w_j)}{d_{ij}}, \dfrac{w_i}{w_j} \ge m_{ij}, \end{cases} \qquad (5')$$

where d_{ij} is a parameter for interval $[l_{ij}, u_{ij}]$, m_{ij} is the most probable value of the preference from the interval $[l_{ij}, u_{ij}]$, namely the middle of this interval. The functions (5′) are as follows: $\mu_{ij}(w): R^{n+} \to [-\infty, \mu^{\max}]$.

The Bellman-Zadeh mathematical programming approach is uses in [18] for calculating the weight vector on the basis of IPCM $A = \{a_{ij} = [l_{ij}, u_{ij}] \mid i, j = 1,...,n\}$. Let $\mu_k(P_k w)$, $k = 1,2,...,q$ be the membership functions (5) of fuzzy constraints $P_k w \lesssim 0$ on the $(n-1)$-measuring simplex $T^{n-1} = \left\{(w_1,...,w_n) \mid w_i \in R^+, \sum_{i=1}^{n} w_i = 1\right\}$.

Definition. A fuzzy feasible region $\tilde{A}$ on T^{n-1} is a fuzzy set, which is an intersection of fuzzy constraints (4) [18]:

$$\mu_{\tilde{A}}(w) = \left\{\min_{k=1,...,q} \{\mu_k(P_k w)\} \mid w_i \in R^+, \sum_{i=1}^{n} w_i = 1\right\}. \qquad (6)$$

The parameters d_k values in (5) should be chosen large enough to obtain a non-empty area $\tilde{A}$ (6). In this case, fuzzy set $\tilde{A}$ (6) on T^{n-1} is convex. If fuzzy constraints (3) are defined using membership functions (5′), the requirement of non-emptiness of the set $\tilde{A}$ (6) can be weakened, and $\tilde{A}$ is defined as follows:

$$\mu_{\tilde{A}}(w) = \min_{\substack{i=1,2,...,n-1, \\ j=2,3,...,n}} \{\mu_{ij}(w)\}. \qquad (6')$$

Obviously, the fuzzy set $\tilde{A}$ (6′) is convex. In contrast to the function (6), the function (6′) can take negative values for a strongly inconsistent IPCM in a case of too small values of the parameter d_k.

The area $\tilde{A}$ defined by (6) or (6′) indicates the overall satisfaction for the decision maker with a certain crisp weight vector. The maximizing solution is the resulting weight vector.

Definition. The maximizing solution of the problem is the vector [18]:

$$w^* = \arg\max_{w \in T^{n-1}} \min_k \mu_{\lesssim_k}(w), \qquad (7)$$

where $\lesssim_k$ presents the *k*-th fuzzy inequality in (4), namely $P_k w \lesssim 0$.

Depending on the choice of the membership function $\mu_{\lesssim_k}(w)$ in (7), the maximizing solution can be calculated in one of the following ways:

$$w^* = \arg\max_w \left\{\min_{k=1,...,q} \{\mu_k(P_k w)\} \mid w_i \in R^+, \sum_{i=1}^{n} w_i = 1\right\}, \qquad (8)$$

where values $\mu_k(P_k w)$ are defined as in (5), or

$$w^* = \arg\max_{w \in T^{n-1}} \min_{i,j} \{\mu_{ij}(w)\}, \qquad (8')$$

where $\mu_{ij}(w)$ are defined as in (5′).

In case when all fuzzy constraints are determined using the membership functions (5), at least one point w^* is present in T^{n-1}, which has the maximum degree of membership in the set $\tilde{A}$. However, the solution to problem (8) will not necessarily be unique. The optimization problem (8′), in turn, has a unique solution.

By introducing a variable λ, the problem (7) for resulting weight vector calculation is presented as a following fuzzy mathematical programming problem:

$$\max \lambda \qquad (9)$$

under constraints

$\lambda \le \mu_k(P_k w)$, $k = 1,2,...,q$,

where values $\mu_k(P_k w)$ are defined as in (5).

Problem (9) can also be written as

$$\max \lambda \qquad (9')$$

under constraints

$\lambda \le \mu_{ij}(w)$, $i = 1,2,...,n-1$, $j = 2,3,...,n$, $i < j$,

where $\mu_{ij}(w)$ are defined as in (5′).

The membership functions (5′) are linear with respect to variables $w_1,...,w_n$, so (9′) can be written as a following linear programming problem [18]:

DOI 10.15588/1607-3274-2022-3-15

$$\max \lambda \quad (10)$$

under constraints

$d_{ij}\lambda + w_i - u_{ij} w_j \le d_{ij}$,

$d_{ij}\lambda - w_i + l_{ij} w_j \le d_{ij}$,

$i = 1,2,...,n-1$, $j = 2,3,...,n$, $i < j$,

$\sum_{k=1}^{n} w_k = 1$, $w_k > 0$, $k = 1,2,...,n$.

The pair (w^*, λ^*) forms the solution to problem (10), where w^* is the resulting weight vector, and $\lambda^* = \mu_{\tilde{A}}(w^*)$ is the value of maximum membership in the aggregated set $\tilde{A}$. The value λ^* measures the degree of expert's (decision maker's) overall satisfaction with the optimal solution w^*. Therefore $\lambda^* = \mu_{\tilde{A}}(w^*)$ is defined as an indicator of consistency level of expert (decision maker) judgments [18].

If IPCM is consistent, then $\lambda^* \ge 1$. Indeed, according to the definition of consistent IPCM, there is a vector w, $w_i \in R^+$, $\sum_{i=1}^{n} w_i = 1$, satisfying $l_{ij} \le w_i / w_j \le u_{ij}$, $i < j$. Therefore, $P_k w \le 0$ and $\mu_k(P_k w) \ge 1$ are carried out for every $k = 1,2,...,q$. Consequently, $\mu_{\tilde{A}}(w) = \left\{ \min_{k=1,...,m} (\mu_k(P_k w)) \mid w_i \in R^+, \sum_{i=1}^{n} w_i = 1 \right\} \ge 1$, and $\lambda^* \ge 1$, where $w^* = w$.

Next, let us consider an inconsistent IPCM and a case when a weight vector w exists, such that the system of inequalities (2) is satisfied. Then, there is such $k = 1,2,...,q$ that $P_k w > 0$ and $\mu_k(P_k w) < 1$. Consequently, $\mu_{\tilde{A}}(w) < 1$ and $\lambda^* < 1$. By choosing large enough values of d_k parameter, a positive value λ^* can be achieved. It can be shown that $\lambda^* > 0$ if $d_k \ge 1$. Thus, for an inconsistent IPCM we have $\lambda^* \in (0,1)$, and λ^* depends on the inconsistency level of IPCM and d_k values.

Let us consider problems (9), (9′) and (10) for calculating the resulting weight vector. In these problems, the weights are defined based on expert's estimates presented in the upper triangular part of the IPCM. As noted above, the IPCM has the property of inverse symmetry. So, the sets of elements of the IPCM, which form the upper and lower triangular parts of the IPCM, carry the same information about the unknown weights. Therefore, the solution based on problem (10) and judgments $[l_{ij}, u_{ij}]$, $i < j$ must coincide with the solution of the same problem based on judgments $[l_{ij}, u_{ij}]$, $i > j$. However, as shown in examples below, the weight vectors according to the "upper" and "lower" FPP models [18, 19] do not coincide with each other.

"Upper" FPP model: $\max \lambda$

under constraints

$\lambda \le \mu_{ij}(w)$, $i = 1,2,...,n-1$, $j = 2,3,...,n$, $i < j$.

"Lower" FPP model: $\max \lambda$

under constraints

$\lambda \le \mu_{ji}(w)$, $j = 1,2,...,n-1$, $i = 2,3,...,n$, $j < i$.

The above "upper" and "lower" FPP models have additional constraints $\sum_{k=1}^{n} w_k = 1$, $w_k > 0$, $k = 1,2,...,n$.

The difference in the results based on the "upper" and "lower" FPP models means that the FPP solution [18, 19] depends on how the compared elements are numbered. In this paper, a new method is proposed that does not have such drawback.

Consider membership functions (5′), where the middle of the interval $[l_{ij}, u_{ij}]$ is chosen as the most probable preference value m_{ij}. Indeed, an expert or a decision maker is not equally satisfied with all the ratios of the resulting weights inside or outside the intervals (2) for $a_{ij}, i < j$. Obviously, he/she would prefer a solution around the middle of each interval than a solution on the given boundaries of the intervals for a_{ij}. On the other hand, in case of inconsistent IPCM, any ratio of resulting weights w_i / w_j that is outside the interval a_{ij} close to the boundaries of this interval is more preferable than the solution far from these boundaries. Therefore, the degree of satisfaction of an expert or a decision maker with the ratio of resulting weights should be presented as a monotone continuous function, gradually increasing towards the middle of the interval.

However, the use of the middle of interval $m_{ij} = \frac{l_{ij} + u_{ij}}{2}$ seems to be justified only for the case $1 \le l_{ij} \le u_{ij}$. The middle of interval as the most probable preference value is not acceptable for all intervals in IPCMs. For example, if $l_{ij} \le u_{ij} \le 1$ then the value $m_{ij} = (\frac{(l_{ij})^{-1} + (u_{ij})^{-1}}{2})^{-1} = \frac{2}{(l_{ij})^{-1} + (u_{ij})^{-1}}$ should be considered as the most probable value of preference. In the third possible case $l_{ij} \le 1 \le u_{ij}$, it is reasonable to choose the value $m_{ij} = \frac{u_{ij} + 1}{(l_{ij})^{-1} + 1}$ as the most probable preference value.

Thus, it is proposed to determine the value m_{ij} depending on $[l_{ij}, u_{ij}]$ as follows:

– if $1 \le l_{ij} < u_{ij}$, then $m_{ij} = \frac{l_{ij} + u_{ij}}{2}$, (11)

– if $l_{ij} < u_{ij} \le 1$, then $m_{ij} = \frac{2}{(l_{ij})^{-1} + (u_{ij})^{-1}}$, (12)

– if $l_{ij} < 1 < u_{ij}$, then $m_{ij} = \frac{u_{ij} + 1}{(l_{ij})^{-1} + 1}$. (13)

DOI 10.15588/1607-3274-2022-3-15

According to the new values m_{ij} (11) – (13), the membership functions (5′) are changed as follows:

1) if $1 \le l_{ij} < u_{ij}$, then

$$\mu_{ij}(w_i, w_j) = \begin{cases} 1 - \dfrac{(-w_i + l_{ij} w_j)}{d_{ij}}, \dfrac{w_i}{w_j} \le m_{ij}, \\ 1 - \dfrac{(w_i - u_{ij} w_j)}{d_{ij}}, \dfrac{w_i}{w_j} \ge m_{ij}, \end{cases} \quad (14)$$

where m_{ij} is calculated as in (11),

2) if $l_{ij} < u_{ij} \le 1$, then

$$\mu_{ij}(w_i, w_j) = \begin{cases} 1 - \dfrac{u_{ij}(-w_i + l_{ij} w_j)}{d_{ij}}, \dfrac{w_i}{w_j} \le m_{ij}, \\ 1 - \dfrac{l_{ij}(w_i - u_{ij} w_j)}{d_{ij}}, \dfrac{w_i}{w_j} \ge m_{ij}, \end{cases} \quad (15)$$

where m_{ij} is calculated as in (12),

3) if $l_{ij} < 1 < u_{ij}$, then

$$\mu_{ij}(w_i, w_j) = \begin{cases} 1 - \dfrac{(-w_i + l_{ij} w_j)}{(l_{ij}-1)(u_{ij}-1) + d_{ij}}, \dfrac{w_i}{w_j} \le m_{ij}, \\ 1 - \dfrac{(w_i - u_{ij} w_j)}{(l_{ij}-1)(u_{ij}-1) + d_{ij}}, \dfrac{w_i}{w_j} \ge m_{ij}, \end{cases} \quad (16)$$

where m_{ij} is calculated as in (13), and the value of d_{ij} parameter is proposed to be equal to $d_{ij} = u_{ij} - l_{ij}$ for all the above cases 1) – 3).

In order to preserve the structure of expert (decision-maker) preferences, to further assess and improve the consistency level of interval expert pairwise comparison judgments, a modified method is proposed for calculating a crisp vector of priorities or weights for elements of DM model. The method includes the following two models:

Proposed "upper" model:

$$\max \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \lambda_{ij}, \quad (17)$$

– under constraints

$$\lambda_{ij} \le \mu_{ij}(w),\ i = 1,2,...,n-1,\ j = 2,3,...,n,\ i < j,$$

where $\mu_{ij}(w)$ values are calculated as in (14)–(16).

Proposed "lower" model:

$$\max \sum_{j=1}^{n-1} \sum_{i=j+1}^{n} \lambda_{ij} \quad (18)$$

– under constraints

$$\lambda_{ji} \le \mu_{ji}(w),\ j = 1,2,...,n-1,\ i = 2,3,...,n,\ j < i,$$

where $\mu_{ij}(w)$ are calculated as in (14) – (16).

The "upper" and "lower" models (17) and (18) have additional constraints $\sum_{k=1}^{n} w_k = 1$, $w_k > 0$, $k = 1,2,...,n$. A similar approach to constructing the objective function as a sum $\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} (p_{ij} + q_{ij})$ is used in another TLGP method [21], where p_{ij}, q_{ij} are variables that form unknown extended intervals. TLGP is a two-stage method and results in optimization problems with non-linear constraints.

Statement. The weight vector based on the "lower" model (18) is equal to the weight vector based on the "upper" model (17).

Proof. Suppose that the constraints $\lambda_{ij} \le \mu_{ij}(w)$, $\forall i < j$ of the "upper" model are satisfied. Let us show that for each of the conditions 1–3 for the ends of the interval $[l_{ij}, u_{ij}]$ in (14)–(16), the objective functions of the "upper" and "lower" models coincide and the constraints of the "upper" and "lower" models are also coincide.

Consider $l_{ij} \lessapprox a_{ij} \lessapprox u_{ij}$ and the coresponding condition $\lambda_{ij} \le \mu_{ij}(w)$ for $i < j$. Let $1 \le l_{ij} < u_{ij}$ be satisfied for $i < j$. Then the constraint $\lambda_{ij} \le \mu_{ij}(w)$, $i < j$ in (17) is equivalent to the fulfillment of two inequalities:

$$d_{ij}\lambda_{ij} + w_i - u_{ij} w_j \le d_{ij},\ i < j, \quad (19)$$

$$d_{ij}\lambda_{ij} - w_i + l_{ij} w_j \le d_{ij},\ j < i. \quad (20)$$

Consider $\frac{1}{u_{ij}} \lessapprox a_{ji} \lessapprox \frac{1}{l_{ij}}$ and the coresponding condition $\lambda_{ji} \le \mu_{ji}(w)$ for $j < i$. In this case, $l_{ji} < u_{ji} \le 1$, $j < i$, and inequality $\lambda_{ji} \le \mu_{ji}(w)$, $j < i$ is equivalent to the following two:

$$d_{ji}\lambda_{ji} + l_{ji} w_j - l_{ji} u_{ji} w_i \le d_{ji},\ j < i, \quad (19')$$

$$d_{ji}\lambda_{ji} - \frac{1}{l_{ij}} w_j + \frac{1}{l_{ij}} \frac{1}{u_{ij}} w_i \le d_{ji},\ j < i. \quad (20')$$

The inequality (19′) is written as

$$d_{ji}\lambda_{ji} + \frac{1}{u_{ij}} w_j - \frac{1}{u_{ij}} \frac{1}{l_{ij}} w_i \le d_{ji} \Leftrightarrow$$

$$d_{ji} u_{ij} l_{ij} \lambda_{ji} + l_{ij} w_j - w_i \le d_{ji} u_{ij} l_{ij}. \quad (19'')$$

The inequality (20′) is written as

$$d_{ji} u_{ij} l_{ij} \lambda_{ji} - u_{ij} w_j + w_i \le d_{ji} u_{ij} l_{ij}. \quad (20'')$$

Note that inequality (19″) is equivalent to (20), and (20″) is equivalent to (19), if $d_{ij} = d_{ji} u_{ij} l_{ij}$ and $\lambda_{ij} = \lambda_{ji}$.

DOI 10.15588/1607-3274-2022-3-15

For example, when choosing $d_{ij} = u_{ij} - l_{ij}$, equality $d_{ij} = d_{ji} u_{ij} l_{ij}$ is satisfied, since $d_{ji} = u_{ji} - l_{ji} = \frac{u_{ij} - l_{ij}}{u_{ij} l_{ij}}$.

The proof is similar in other two cases: $l_{ij} < u_{ij} \le 1$, if $i < j$, and $l_{ij} < 1 < u_{ij}$, if $i < j$.

4 EXPERIMENTS

Let us consider several matrices of different consistency levels (Tables 1 – 4), which other researchers have analyzed using other methods. IPCM *A*1 [19] (Table 1) is not consistent by definition, and indicator $\lambda^* = 0.9583$. However, the IPCM *A*1 is weakly consistent, since $(a_{12} > 1) \wedge (a_{23} > 1) \Rightarrow (a_{13} > 1)$.

IPCMs *A*2 [21, 23, 26, 29] (Table 2) and *A*3 [22] (Table 3) are consistent by definition. Consistency is also confirmed by values $\lambda^* = 1.0435$ and $\lambda^* = 1.0313$ of these IPCMs. Consistency is a stronger concept than weak consistency. So, IPCMs *A*2 and *A*3 are weakly consistent.

PCM *B* (Table 4) has proposed in [10] to solve the sprint planning problem. PCM *B* is not consistent by definition, since the transitivity condition $b_{ij} = b_{ik} b_{kj}$ is not satisfied for all $i, j, k = 1, ..., n$. However, the consistency ratio CR = 0.083 is less than its threshold value, so PCM *B* is admissibly inconsistent and can be used for weights calculation. PCM *B* is weakly consistent, since $(b_{ik} > 1) \wedge (b_{kj} > 1) \Rightarrow (b_{ij} > 1)$ for all $i, j, k = 1, ..., 5$.

IPCM A4 (Table 5) is the result of *B* fuzzification for the application of the proposed method. IPCM *A*4 is consistent, since $\lambda^* = 1.0244$.

Let us disturb individual elements of IPCM *A*4 in order to increase the inconsistency level of this matrix. Inconsistent IPCMs often occur in practice. Therefore, it is interesting to investigate the implementation of the proposed method also on IPCMs of this class. So, only the element $a_{21} := [2, 4]$ of IPCM *A*4 and the symmetric element are changed. Resulting IPCM *A*5 is shown in Table 6. Such IPCM can be, for example, the result of an accidental expert error. The value $\lambda^* = 0.9500$ of IPCM *A*5 indicates an increase in inconsistency compared to *A*4. The same conclusion is based on the consistency ratio CR = 0.650 (after *A*5 defuzzification). In addition, IPCM *A*5 is not weakly consistent, which means it has a cycle and an undesirable violation of ordinal transitivity on the set of its elements.

The next IPCM A6 (Table 7) coincides with *A*4 except for the element $a_{24} := [3, 5]$ and symmetrical to it $a_{42} := [1/5, 1/3]$. Values $\lambda^* = 0.8611$ and CR = 1.002 (after defuzzification of *A*6) for IPCM *A*6 mean that this matrix is the most strongly inconsistent in comparison with *A*4, *A*5 and *A*1 – *A*3.

Table 1 – Weakly consistent IPCM *A*1 [19]

1	[1, 2]	[8, 9]
[1/2, 1]	1	[2, 3]
[1/9, 1/8]	[1/3, 1/2]	1

Table 2 – Consistent IPCM *A*2 [21, 23 , 26, 29]

1	[2, 5]	[2, 4]	[1, 3]
[1/5, 1/2]	1	[1, 3]	[1, 2]
[1/4, 1/2]	[1/3, 1]	1	[1/2, 1]
[1/3, 1]	[1/2, 1]	[1, 2]	1

Table 3 – Consistent IPCM *A*3 [22]

1	[1, 3]	[3, 5]	[5, 7]	[5, 9]
[1/3, 1]	1	[1, 4]	[1, 5]	[1, 4]
[1/5, 1/3]	[1/4, 1]	1	[1/5, 5]	[2, 4]
[1/7, 1/5]	[1/5, 1]	[1/5, 5]	1	[1, 2]
[1/9, 1/5]	[1/4, 1]	[1/4, 1/2]	[1/2, 1]	1

Table 4 – PCM *B* [10]

1	3	3	1/2	2
1/3	1	1/2	1/4	1/2
1/3	2	1	1/3	1/2
2	4	3	1	2
1/2	2	2	1/2	1

Table 5 – Consistent IPCM *A*4

[1, 1]	[2, 4]	[2, 4]	[1/3, 1]	[1, 3]
[1/4, 1/2]	[1, 1]	[1/3, 1]	[1/5, 1/3]	[1/3, 1]
[1/4, 1/2]	[1, 3]	[1, 1]	[1/4, 1/2]	[1/3, 1]
[1, 3]	[3, 5]	[2, 4]	[1, 1]	[1, 3]
[1/3, 1]	[1, 3]	[1, 3]	[1/3, 1]	[1, 1]

Table 6 – Weakly inconsistent IPCM *A*5

[1, 1]	**[1/4, 1/2]**	[2, 4]	[1/3, 1]	[1, 3]
[2, 4]	[1, 1]	[1/3, 1]	[1/5, 1/3]	[1/3, 1]
[1/4, 1/2]	[1, 3]	[1, 1]	[1/4, 1/2]	[1/3, 1]
[1, 3]	[3, 5]	[2, 4]	[1, 1]	[1, 3]
[1/3, 1]	[1, 3]	[1, 3]	[1/3, 1]	[1, 1]

Table 7 – The most strongly inconsistent IPCM *A*6

[1, 1]	[2, 4]	[2, 4]	[1/3, 1]	[1, 3]
[1/4, 1/2]	[1, 1]	[1/3, 1]	**[3, 5]**	[1/3, 1]
[1/4, 1/2]	[1, 3]	[1, 1]	[1/4, 1/2]	[1/3, 1]
[1, 3]	**[1/5, 1/3]**	[2, 4]	[1, 1]	[1, 3]
[1/3, 1]	[1, 3]	[1, 3]	[1/3, 1]	[1, 1]

5 RESULTS

In the following Tables 8–13, weights are shown, which are calculated using different methods based on the above-considered IPCMs. The λ^*_{ij} values for these IPCMs (Table 14) are further used to find the most inconsistent elements in the IPCMs. Calculated weights are also compared with weights by the Saaty's eigenvector method (EM) on the basis of defuzzified IPCMs *A*5 and *A*6 (Tables 12 and 13).

DOI 10.15588/1607-3274-2022-3-15

6 DISCUSSION

As follows from Tables 8–13, the weights based on the "lower" FPP model [18, 19] differ from the weights based on the "upper" FPP model [18, 19] for all the considered IPCMs. The "upper" and "lower" models proposed in this paper lead to the same resulting weight vectors.

Table 8 – Weights obtained using different methods based on IPCM $A1$ from Table 1

	Weights			
	FPP model		Proposed model	
	"upper"	"lower"	"upper"	"lower"
w_1	0.6250	0.6154	0.5604	0.5604
w_2	0.2917	0.2967	0.3736	0.3736
w_3	0.0833	0.0879	0.0659	0.0659

Table 9 – Weights obtained using different methods based on consistent IPCM $A2$ from Table 2

	Interval weights			
	Li&Tong's method [26]	TLGP [21]	Liu's method [23]	Kuo's method [29]
w_1	[1.5540, 2.5329]	[1.6818 2.4495]	[1.4142, 2.7832]	0.4499
w_2	[0.7348, 1.1977]	[0.7598 1.1067]	[0.8409, 1.0466]	0.2127
w_3	[0.5105, 0.7442]	[0.5000 0.8409]	[0.5373, 0.7071]	0.1398
w_4	[0.7219, 1.0525]	[0.6866 1.0000]	[0.6389, 1.1892]	0.1977

Table 9 continuation – Weights obtained using different methods based on consistent IPCM $A2$ from Table 2

	Weights			
	FPP model		Proposed model	
	"upper"	"lower"	"upper"	"lower"
w_1	0.4783	0.4675	0.4000	0.4000
w_2	0.2174	0.2078	0.2667	0.2667
w_3	0.1304	0.1429	0.1333	0.1333
w_4	0.1739	0.1818	0.2000	0.2000

Table 10 – Weights obtained using different methods based on consistent IPCM $A3$ from Table 3

	Interval weights		
	LUAM-1	LUAM-2	GPM
w_1	[0.2909, 0.4091]	[0.4225, 0.5343]	0.4527
w_2	[0.1364, 0.2909]	[0.1781, 0.2817]	[0.1396, 0.3320]
w_3	[0.0273, 0.1818]	0.1409	[0.0818, 0.2097]
w_4	[0.0364, 0.1364]	[0.0763, 0.0845]	[0.0591, 0.1347]
w_5	[0.0455, 0.1364]	0.0704	0.0633

Table 10 continuation – Weights obtained using different methods based on consistent IPCM $A3$ from Table 3

	Weights			
	FPP model		Proposed model	
	"upper"	"lower"	"upper"	"lower"
w_1	0.5000	0.5089	0.4855	0.4855
w_2	0.1875	0.1809	0.2428	0.2428
w_3	0.1563	0.1583	0.1214	0.1214
w_4	0.0937	0.0840	0.0809	0.0809
w_5	0.0625	0.0679	0.0694	0.0694

Table 11 – Weights obtained using different methods based on consistent IPCM $A4$ from Table 5

	Weights					
			FPP model		Proposed model	
	g_i [10]	Normalised g_i	"upper"	"lower"	"upper"	"lower"
w_1	1.55	0.2660	0.2683	0.3000	0.2647	0.2647
w_2	0.46	0.0790	0.0976	0.1000	0.0882	0.0882
w_3	0.64	0.1110	0.1220	0.1250	0.1176	0.1176
w_4	2.17	0.3720	0.3659	0.3250	0.3529	0.3529
w_5	1.00	0.1720	0.1463	0.1500	0.1765	0.1765

Table 12 – Weights obtained using different methods based on weakly inconsistent IPCM $A5$ from Table 6

	Weights				
		FPP model		Proposed model	
	EM	"upper"	"lower"	"upper"	"lower"
w_1	0.1810	0.1167	0.1149	0.1935	0.1935
w_2	0.1290	0.1333	0.1609	0.0968	0.0968
w_3	0.1170	0.0833	0.1264	0.1290	0.1290
w_4	0.3920	0.5000	0.4138	0.3871	0.3871
w_5	0.1810	0.1667	0.1839	0.1935	0.1935

Table 13 – Weights obtained using different methods based on the most strongly inconsistent IPCM $A6$ from Table 7

	Weights				
		FPP model		Proposed model	
	EM	"upper"	"lower"	"upper"	"lower"
w_1	0.1810	0.2500	0.2500	0.3000	0.3000
w_2	0.1290	0.1944	0.2143	0.1000	0.1000
w_3	0.1170	0.1944	0.1250	0.1000	0.1000
w_4	0.3920	0.1111	0.1607	0.3000	0.3000
w_5	0.1810	0.2500	0.2500	0.2000	0.2000

Table 14 – Values λ_{ij}^* for IPCMs of different consistency

IPCM	Values λ_{ij}^*
Weakly consistent $A1$ from Table 1	[1.1868, 1.0330, 0.8242]
Consistent $A2$ from Table 2	[0.9556, 1.0667, 1.1000, 1.0667, 1.0667, 1.0667]
Consistent $A3$ from Table 3	[1.1214, 1.0607, 1.0405, 1.0347, 1.0405, 1.0405, 1.0116, 1.0354, 0.9913, 1.0116]
Consistent $A4$ from Table 5	[1.0441, 1.0147, 1.0441, 1.0441, 1.0147, 1.0441, 1.0441, 1.0588, 1.0294, 1.0882]
Weakly inconsistent $A5$ from Table 6	[**0.8548**, 0.9677, 1.0968, 1.0000, 1.0161, 1.0484, 1.0484, 1.0645, 1.0323, 1.0968]
Weakly inconsistent $A6$ from Table 7	[1.05, 1.05, 1.00, 1.05, 1.00, **0.6**, 1.05, 1.05, 1.05, 1.05]

Analysis of Tables 8–11 shows that the weights obtained by different methods on the basis of the consistent IPCMs $A2$–$A4$ (Tables 9–11) and weakly consistent IPCM $A1$ (Table 8) provide the same rankings of the compared objects. Therefore, the results obtained by the proposed models do not contradict the results for such IPCMs obtained by other known methods [10, 19, 21, 22, 23, 26, 29]. The modeling shows that solutions to the problem of choosing one "best" object, obtained by different methods on the basis of a consistent IPCMs, are generally coincide. The same is true for solutions to the

DOI 10.15588/1607-3274-2022-3-15

problems of ranking and rating of objects based on weakly consistent IPCMs.

When highly inconsistent expert estimates of paired comparisons (IPCMs *A*5 and *A*6, see Tables 6 and 7) are the input data, then different considered methods lead not only to different weights, but also to different rankings of compared objects (Tables 12 and 13). For example, for IPCM *A*5, the "upper" FPP model specifies the ranking $w_4 > w_5 > w_2 > w_1 > w_3$, the "lower" FPP model specifies the ranking $w_4 > w_5 > w_2 > w_3 > w_1$, and the proposed method specifies another ranking $w_4 > w_5 = w_1 > w_3 > w_2$ (Table 12). For IPCM *A*6, the "upper" FPP model determines $w_1 = w_5 > w_2 = w_3 > w_4$, the "lower" FPP model determines $w_1 = w_5 > w_2 > w_4 > w_3$, and the proposed method determines the ranking $w_1 = w_4 > w_5 > w_2 = w_3$ (Table 13).

The result $w_1 = w_4 > w_5 > w_2 = w_3$ (Table 13) by the proposed method based on the inconsistent IPCM *A*6 is significantly closer to the ranking $w_4 > w_1 > w_5 > w_3 > w_2$ (Table 11) by the proposed method based on the initial unperturbed IPCM *A*4 in comparison with the FPP rankings. Therefore, the resulting vector of weights by the proposed method based on the inconsistent IPCMs with outliers, like IPCM *A*6, is not significantly sensitive to individual strongly perturbed element in this matrix.

Another advantage of the proposed method in comparison with FPP is that it becomes possible to find the most inconsistent elements in highly inconsistent IPCMs based on values λ^*_{ij}, in order to their further correction to reduce the inconsistency level of the entire IPCM. So, the element a_{12} of IPCM *A*5 (Table 6) is the most inconsistent. This element corresponds to the element λ^*_{12}, which is equal to 0.8548 and is the smallest element of corresponding λ^*_{ij} (Table 14). In IPCM *A*6 (Table 7), the most inconsistent element is a_{24}. It corresponds to the element λ^*_{24}, which is equal to 0.6000 and is the smallest element of these λ^*_{ij} (Table 14). In the consistent IPCMs *A*2 – *A*4, the elements λ^*_{ij} generally take on values greater than one.

Solutions to problems (9) and (9′) [18, 19] depend on the choice of values of parameters d_k and d_{ij} of membership functions $\mu_k(P_k w)$ (5) and $\mu_{ij}(w)$ (5′), respectively. In the general case, this choice has to be made by a decision maker. In this paper, the values d_{ij} are calculated without the role of a decision maker: $d_{ij} = u_{ij} - l_{ij}$.

The proposed method consists in solving a linear programming problem with $n + n(n-1)/2$ variables. For comparison, the linear programming problem of the FPP method has $(n + 1)$ variables. The linear programming problems of the GPM [22] and LUAM [20] have $6n$ and $4n$ variables, respectively. Additional $n(n-1)/2$ variables in proposed method are needed to preserve the prevalence values determined by an expert, and are also used to find the most inconsistent expert pairwise comparison judgments.

The TLGP [21], Liu's [23], Li&Tong's [26], Nedashkovskaya's [13] and Kuo's [29] methods result in interval and non-normalized weights on the basis of IPCM. So, the question of choosing a method for normalizing interval weights arises. The weights obtained by the proposed method are crisp and therefore do not require special methods for their normalization and ranking. However, interval resulting weights provide more information to the decision maker and are more flexible when applied in the DM process.

The method can be used for a wide range of decision support problems, planning, prioritization and rating, resource allocation problems, evaluating decision alternatives and criteria in various applied areas.

CONCLUSIONS

The scientific novelty of the obtained results consists in suggestion of a modified method for calculating a crisp weight vector for elements of DM model based on consistent and inconsistent multiplicative IPCMs. The differences between the proposed method and the known ones are as follows: coefficients that characterize extended intervals for ratios of weights are introduced; membership functions of fuzzy preference relations are proposed, which depend on values of IPCM elements. The introduction of these coefficients and membership functions made it possible to prove the statement about the required coincidence of the calculated weights based on the "upper" and "lower" models. The introduced coefficients can be further used to find the most inconsistent IPCM elements. Therefore, the proposed method results in more reliable weight vectors on the basis of inconsistent multiplicative IPCMs compared to another known methods.

The results show **the practical significance** of the method for solving prioritization problems. Resulting weight vectors, priorities and rankings based on them are not sensitive to the renumbering of the compared elements in contrast to the known FPP method. The most inconsistent elements of IPCM, which are a by-product of the proposed method, are further used for adjusting the elements of IPCM to improve the quality of decisions based on these matrices. In the proposed method, experts or a decision-maker are not required to set the parameters of the membership functions, these parameters are calculated without the participation of experts. Modeling has shown that the weights and rankings obtained by the proposed method on the basis of consistent and weakly consistent IPCMs do not contradict the weights and rankings for such IPCMs calculated using other known methods. For strongly inconsistent studied IPCMs, the proposed method gave more reliable weights compared to the other known methods, since the obtained weights were practically insensitive to individual strongly perturbed during the simulation elements in these matrices.

Prospects for further research consists in investigation of other types of fuzzy preference relations on a set of compared decision alternatives, other types of interval

DOI 10.15588/1607-3274-2022-3-15

and fuzzy PCMs, in particular, type-2 fuzzy PCMs; investigation of aggregation of fuzzy local weights of decision alternatives according to multiple criteria; application of the developed method for solving practical problems of decision support in various applied areas.

ACKNOWLEDGEMENTS

This study was funded and supported by National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute" (NTUU KPI) in Kyiv (Ukraine), and also financed in part of the NTUU KPI Science-Research Work by the Ministry of Education and Science of Ukraine "Development of the theoretical foundations of scenario analysis based on large volumes of semi-structured information" (State Reg. No. 0117U002150).

REFERENCES

1. Larichev O. I. Science and Art of Decision Making. Moscow, Nauka Publisher, 1979, 200 p. (in Russian).
2. Larichev O. I., Moshkovich H. M., Rebrik S. B. Systematic research into human behavior in multiattribute object classification problems, *Acta Psychologica,* 1988, Vol. 68, Issue 1–3, pp. 171–182. DOI: 10.1016/0001-6918(88)90053-4
3. Simon H. A. The architecture of complexity, *Proceedings of the American Philosophical Society,* 1962, Vol. 106, Issue 6. pp. 467–482. https://www.jstor.org/stable/985254
4. Saaty T. L. The Analytic Hierarchy Process. New York, McGraw-Hill, 1980.
5. Saaty T. L. The modern science of multicriteria decision making and its practical applications: The AHP/ANP approach, *Operations Research,* 2013, Vol. 61, Issue 5, pp. 1101–1118. DOI: 10.1287/opre.2013.1197
6. Bernasconi M.б Choirat C., Seri R. The analytic hierarchy process and the theory of measurement, *Management Science*, 2010, Vol. 56, Issue 4, pp. 699–711. DOI: 10.1287/mnsc.1090.1123
7. Vaidya O. S., Kumar S. Analytic hierarchy process: An overview of applications, *European Journal of Operational Research,* 2006, Vol. 169, Issue 1, pp. 1–29. DOI: 10.1016/j.ejor.2004.04.028
8. Brunelli M. Introduction to the Analytic Hierarchy Process. New York, Springer, 2015, 83 p. DOI: 10.1007/978-3-319-12502-2
9. Titenko E. A., Frolov N. S., Khanis A. L. et al. Models for calculation weights for estimation innovative technical objects, *Radio Electronics, Computer Science, Control*, 2020, No. 3, pp. 181 – 193. DOI: 10.15588/1607-3274-2020-3-17
10. Melnyk K. V., Hlushko V. N., Borysova N. V. Decision support technology for sprint planning, *Radio Electronics, Computer Science, Control,* 2020, No. 1, pp. 135–145. DOI: 10.15588/1607-3274-2020-1-14
11. Rezaei J. Best-worst multi-criteria decision-making method: Some properties and a linear model, *Omega,* 2016, Vol. 64. pp. 126–130. DOI: 10.1016/j.omega.2015.12.001
12. Gwo-Hshiung T., Huang J.-J. Multiple Attribute Decision Making: Methods and Applications. New York, Chapman and Hall, 2011, 352 p. DOI : 10.1201/b11032
13. Nedashkovskaya N. I. Method for Evaluation of the Uncertainty of the Paired Comparisons Expert Judgements when Calculating the Decision Alternatives Weights, *Journal of Automation and Information Sciences*, 2015, Vol. 47, Issue 10, pp. 69–82. DOI: 10.1615/JAutomatInfScien.v47.i10.70
14. Durbach I., Lahdelma R., Salminen P. The analytic hierarchy process with stochastic judgements, *European Journal of Operational Research*, 2014, Vol. 238, No. 2, pp. 552–559. DOI: 10.1016/j.ejor.2014.03.045
15. Wang Y.-M., Chin K.-S. An eigenvector method for generating normalized interval and fuzzy weights, *Applied Mathematics and Computation*, 2006, Vol. 181, pp. 1257–1275. DOI: 10.1016/j.amc.2006.02.026
16. Krejčí J. Fuzzy eigenvector method for obtaining normalized fuzzy weights from fuzzy pairwise comparison matrices, *Fuzzy Sets and Systems,* 2017, Vol. 315, pp. 26–43. DOI: 10.1016/j.fss.2016.03.006
17. Ramík J. Deriving priority vector from pairwise comparisons matrix with fuzzy elements, *Fuzzy Sets and Systems,* 2021, Vol. 422, pp. 68–82. DOI: 10.1016/j.fss.2020.11.022
18. Mikhailov L. Deriving priorities from fuzzy pairwise comparison judgements, *Fuzzy Sets and Systems*, 2003, Vol. 134, Issue 3, pp. 365–385. DOI: 10.1016/S0165-0114(02)00383-4
19. Mikhailov L. A fuzzy approach to deriving priorities from interval pairwise comparison judgements, *European Journal of Operational Research*, 2004, Vol.159, Issue 3, pp. 687–704. DOI: 10.1016/S0377-2217(03)00432-6
20. Sugihara K., Ishii H., Tanaka H. Interval priorities in AHP by interval regression analysis, *European Journal of Operational Research,* 2004, Vol. 158, pp. 745–754. DOI: 10.1016/S0377-2217(03)00418-1
21. Wang Y.-M., Yang J.-B., Xu D.-L. A two-stage logarithmic goal programming method for generating weights from interval comparison matrices, *Fuzzy Sets and Systems,* 2005, Vol. 152, Issue 3, pp. 475–498. DOI: 10.1016/j.fss.2004.10.020
22. Wang Y.-M., Elhag T. M. S. A goal programming method for obtaining interval weights from an interval comparison matrix, *European Journal of Operational Research*, 2007, Vol. 177, Issue 1, pp. 458–471. DOI: 10.1016/j.ejor.2005.10.066
23. Liu F. Acceptable consistency analysis of interval reciprocal comparison matrices, *Fuzzy Sets and Systems,* 2009, Vol. 160, Issue 18, pp. 2686–2700. DOI: 10.1016/j.fss.2009.01.010
24. Entani T., Inuiguchi M. Pairwise comparison based interval analysis for group decision aiding with multiple criteria, *Fuzzy Sets and Systems,* 2015, Vol. 274, pp. 79–96. DOI: 10.1016/j.fss.2015.03.001
25. Pankratova N. D., Nedashkovskaya N. I. Estimation of decision alternatives on the basis of interval pairwise comparison matrices, *Intelligent Control and Automation,* 2016, Vol. 7, Issue 2, pp. 39–54. DOI: 10.4236/ica.2016.72005
26. Li K. W., Wang Z.-J., Tong X. Acceptability analysis and priority weight elicitation for interval multiplicative comparison matrices, *European Journal of Operational Research,* 2016, Vol. 250, Issue 2, pp. 628–638. DOI: 10.1016/j.ejor.2015.09.010
27. Nedashkovskaya N. I. Investigation of methods for improving consistency of a pairwise comparison matrix, *Journal of the Operational Research Society,* 2018, Vol. 69, Issue 12, pp. 1947–1956. DOI: 10.1080/01605682.2017.1415640
28. Pankratova N. D., Nedashkovskaya N. I. Method for processing fuzzy expert information in prediction problems. Part I, *Journal of Automation and Information Sciences,* 2007, Vol. 39, Issue 4, pp. 22–36. DOI: 10.1615/jautomatinfscien.v39.i4.30
29. Kuo T. Interval multiplicative pairwise comparison matrix: Consistency, indeterminacy and normality, *Information Sci-*

DOI 10.15588/1607-3274-2022-3-15

ences, 2020, Vol. 517, pp. 244–253. DOI: 10.1016/j.ins.2019.12.066

30. Liu F., Zhang W. G., Fu J. H. A new method of obtaining the priority weights from an interval fuzzy preference relation, *Information Sciences,* 2012, Vol. 185, Issue 1, pp. 32–42. DOI: 10.1016/j.ins.2011.09.019
31. Xu Y., Li K. W., Wang H. Consistency test and weight generation for additive interval fuzzy preference relations, *Soft Computing*, 2014, Vol. 18, Issue 8, pp. 1499–1513. DOI: 10.1007/s00500-013-1156-x
32. Wang Z-J., Yang X., Jin X.-T. And-like-uninorm-based transitivity and analytic hierarchy process with interval-valued fuzzy preference relations, *Information Sciences,* 2020, Vol. 539, pp. 375–396. DOI: 10.1016/j.ins.2020.05.052
33. Wang Z.-J. A goal programming approach to deriving interval weights in analytic form from interval Fuzzy preference relations based on multiplicative consistency, *Information Sciences*, 2018, Vol. 462, pp. 160–181. DOI: 10.1016/j.ins.2018.06.006
34. López-Morales V. A reliable method for consistency improving of interval multiplicative preference relations expressed under uncertainty, *International Journal of Information Technology & Decision Making,* 2018, Vol. 17, Issue 5, pp. 1561–1585. DOI: 10.1142/S0219622018500359
35. Wang Z.-J., Lin J. Consistency and optimized priority weight analytical solutions of interval multiplicative preference relations, *Information Sciences*, 2019, Vol. 482, pp. 105–122. DOI: 10.1016/j.ins.2019.01.007.
36. Nedashkovskaya N. I. The M_Outflow Method for Finding the Most Inconsistent Elements of a Pairwise Comparison Matrix, *System Analysis and Information Technologies (SAIT), 17th International Conference, Kyiv, 2015, proceeding*. Kyiv, NTUU KPI, 2015, P. 90. http://sait.kpi.ua/media/filer_public/f8/7e/f87e3b7b-b254-407f-8a58-2d810d23a2e5/sait2015ebook.pdf
37. Buckley J. J. Fuzzy hierarchical analysis, *Fuzzy Sets and Systems,* 1985, Vol. 17, Issue 3, pp. 233–247. DOI: 10.1016/0165-0114(85)90090-9
38. Van Laarhoven P.J.M., Pedrycz W. A fuzzy extension of Saaty's priority theory, *Fuzzy Sets and Systems,* 1983, Vol. 11, Issues 1–3, pp. 229–241. DOI: 10.1016/S0165-0114(83)80082-7
39. Wang Y.-M., Elhag T. M. S., Hua Z. A modified fuzzy logarithmic least squares method for fuzzy analytic hierarchy process, *Fuzzy Sets and Systems,* 2006, Vol. 157, Issue 23, pp. 3055–3071. DOI: 10.1016/j.fss.2006.08.010
40. Zhang F., Ignatius J.,. Lim C. P et al. A new method for deriving priority weights by extracting consistent numerical-valued matrices from interval-valued fuzzy judgement matrix, *Information Sciences,* 2014, Vol. 279, pp. 280–300. DOI: 10.1016/j.ins.2014.03.120

Received 15.06.2022.
Accepted 29.07.2022.

УДК 519.816, 681.518.2

МЕТОД РОЗРАХУНКУ ВАГ ЕЛЕМЕНТІВ МОДЕЛІ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ НА ОСНОВІ ІНТЕРВАЛЬНИХ МУЛЬТИПЛІКАТИВНИХ МАТРИЦЬ ПАРНИХ ПОРІВНЯНЬ

Недашківська Н. І. – д-р техн. наук, доцент кафедри математичних методів системного аналізу, Інститут прикладного системного аналізу, НТУУ «Київський політехнічний інститут ім. Ігоря Сікорського», Київ, Україна.

АНОТАЦІЯ

Актуальність. Метод парних порівнянь – складова кількох методологій підтримки прийняття рішень, таких як PROMETHEE, TOPSIS, аналізу ієрархій і мереж. Його суть полягає в розрахунку вектора пріоритетів (ваг) елементів моделі прийняття рішень на основі обернено симетричних матриць парних порівнянь. Оцінювання елементів моделі здійснюється здебільшого експертами в умовах невизначеності. Тому в останні роки досліджуються модифіковані методи розрахунку ваг з використанням нечітких та інтервальних матриць парних порівнянь (ІМПП).

Мета. Розробка модифікованого методу розрахунку ваг на основі узгоджених і неузгоджених мультиплікативних ІМПП елементів моделі прийняття рішень.

Метод. Запропоновано модифікований метод на основі узгоджених і неузгоджених мультиплікативних ІМПП та нечіткого програмування переваг, який призводить до більш достовірних ваг елементів моделі прийняття рішень порівняно з іншими відомими методами. Розроблений метод відрізняється від інших наступними особливостями: введено коефіцієнти, які характеризують розширені інтервали для відношень невідомих ваг; запропоновано функції належності нечітких відношень нестрогої переваги залежно від значень елементів ІМПП. Введення вказаних коефіцієнтів і функцій належності дозволило довести твердження про несуперечливість результуючих ваг на основі «верхньої» та «нижньої» моделей. Пропоновані коефіцієнти в подальшому використовуються для пошуку найбільш неузгоджених елементів ІМПП.

Результати. Виконано експерименти з кількома ІМПП різного рівня узгодженості. Ваги, отримані пропонованим та іншими відомими методами на основі розглянутих узгоджених та слабко узгоджених ІМПП, визначили однакові ранжування порівнюваних об'єктів. Результати, отримані пропонованим методом, не суперечать результатам для таких ІМПП за іншими відомими методами. Ранжування пропонованим методом на основі розглянутих сильно збурених ІМПП суттєво ближчі до ранжувань на основі відповідних початкових незбурених ІМПП порівняно з ранжуваннями відомим методом FPP. Знайдено найбільш неузгоджені елементи в розглянутих ІМПП.

Висновки. Розроблений метод показав свою ефективність і може використовуватися для широкого кола задач підтримки прийняття рішень, сценарного аналізу, розрахунку пріоритетів, розподілу ресурсів, оцінювання варіантів та критеріїв рішень у різних прикладних областях.

КЛЮЧОВІ СЛОВА: інтервальна мультиплікативна матриця парних порівнянь, узгодженість, експертні оцінки, нечітке програмування переваг, системи підтримки прийняття рішень.

DOI 10.15588/1607-3274-2022-3-15

УДК 519.816, 681.518.2

МЕТОД РАСЧЕТА ВЕСОВ ЭЛЕМЕНТОВ МОДЕЛИ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ НА ОСНОВЕ ИНТЕРВАЛЬНЫХ МУЛЬТИПЛИКАТИВНЫХ МАТРИЦ ПАРНЫХ СРАВНЕНИЙ

Недашковская Н. И. – д-р техн. наук, доцент кафедры математических методов системного анализа, Институт прикладного системного анализа НТУУ «Киевский политехнический институт им. Игоря Сикорского», Киев, Украина.

АННОТАЦИЯ

Актуальность. Метод парных сравнений – составляющая нескольких методологий поддержки принятия решений, таких как PROMETHEE, TOPSIS, анализа иерархий и сетей. Его суть заключается в расчете вектора приоритетов или весов элементов модели принятия решений на основе обратно симметричных матриц парных сравнений. Оценка элементов модели осуществляется в основном экспертами в условиях неопределенности. Поэтому в последние годы исследуются модификации методов расчета весов с использованием нечетких и интервальных матриц парных сравнений (ИМПС).

Цель. Разработка модифицированного метода расчета приоритетов на основе согласованных и несогласованных мультипликативных ИМПС элементов модели принятия решений.

Метод. Предложен модифицированный метод на основе согласованных и несогласованных мультипликативных ИМПС и нечеткого программирования предпочтений, который приводит к более достоверным весам элементов модели принятия решений по сравнению с другими известными методами. Разработанный метод отличается от других следующими особенностями: введены коэффициенты, характеризующие расширенные интервалы для отношений неизвестных весов; предложены функции принадлежности нечетких отношений нестрогого предпочтения в зависимости от значений элементов ИМПС. Введение указанных коэффициентов и функций принадлежности позволило доказать утверждение о непротиворечивости результирующих весов на основе «верхней» и «нижней» моделей. Предлагаемые коэффициенты в дальнейшем используются для поиска наиболее несогласованных элементов ИМПС.

Результаты. Выполнены эксперименты с несколькими ИМПС разного уровня согласованности. Веса, полученные предлагаемым и другими известными методами на основе рассмотренных согласованных и слабо согласованных ИМПС, определили одинаковые ранжирования сравниваемых объектов. Результаты, полученные предлагаемым методом, не противоречат результатам для таких ИМПС по другим известным методам. Ранжирования предлагаемым методом на основе рассмотренных сильно несогласованных ИМПС существенно ближе к ранжированиям на основе соответствующих начальных невозмущенных ИМПС по сравнению с ранжированиями известным методом FPP. Найдены наиболее несогласованные элементы в рассмотренных ИМПС.

Выводы. Разработанный метод показал свою эффективность и может использоваться для широкого круга задач поддержки принятия решений, сценарного анализа, расчета приоритетов, распределения ресурсов, оценки вариантов и критериев решений в разных прикладных областях.

КЛЮЧЕВЫЕ СЛОВА: интервальная мультипликативная матрица парных сравнений, согласованность, экспертные оценки, нечеткое программирование предпочтений, системы поддержки принятия решений.

ЛІТЕРАТУРА / ЛИТЕРАТУРА

1. Larichev O. I. Science and Art of Decision Making / O. I. Larichev. – M. : Nauka Publisher, 1979. – 200 p. (in Russian).
2. Larichev O. I. Systematic research into human behavior in multiattribute object classification problems / O. I. Larichev, H. M. Moshkovich, S. B. Rebrik // Acta Psychologica. – 1988. – Vol. 68, Issue 1–3. – P. 171–182. DOI: 10.1016/0001-6918(88)90053-4
3. Simon H. A. The architecture of complexity / H. A. Simon // Proceedings of the American Philosophical Society. – 1962. – Vol. 106, Issue 6. – P. 467–482. https://www.jstor.org/stable/985254
4. Saaty T. L. The Analytic Hierarchy Process / T. L. Saaty. – New York : McGraw-Hill, 1980.
5. Saaty T. L. The modern science of multicriteria decision making and its practical applications: The AHP/ANP approach / T. L. Saaty // Operations Research. – 2013. – Vol. 61, Issue 5. – P. 1101–1118. DOI: 10.1287/opre.2013.1197
6. Bernasconi M. The analytic hierarchy process and the theory of measurement / M. Bernasconi, C. Choirat, R. Seri // Management Science. – 2010. – Vol. 56, Issue 4. – P. 699–711. DOI: 10.1287/mnsc.1090.1123
7. Vaidya O. S. Analytic hierarchy process: An overview of applications / O. S. Vaidya, S. Kumar // European Journal of Operational Research. – 2006. – Vol. 169, Issue 1. – P. 1–29. DOI: 10.1016/j.ejor.2004.04.028
8. Brunelli M. Introduction to the Analytic Hierarchy Process / M. Brunelli. – New York : Springer, 2015. – 83 p. DOI: 10.1007/978-3-319-12502-2
9. Models for calculation weights for estimation innovative technical objects / [E. A. Titenko, N. S. Frolov, A. L. Khanis et al.] // Radio Electronics, Computer Science, Control. – 2020. – No. 3. – P. 181–193. DOI: 10.15588/1607-3274-2020-3-17
10. Melnyk K. V. Decision support technology for sprint planning / K. V. Melnyk, V. N. Hlushko, N. V. Borysova // Radio Electronics, Computer Science, Control. – 2020. – No. 1. – P. 135–145. DOI: 10.15588/1607-3274-2020-1-14
11. Rezaei J. Best-worst multi-criteria decision-making method: Some properties and a linear model / J. Rezaei // Omega. – 2016. – Vol. 64. – P. 126–130. DOI: 10.1016/j.omega.2015.12.001
12. Gwo-Hshiung T. Multiple Attribute Decision Making: Methods and Applications / T. Gwo-Hshiung, J.-J. Huang. – New York : Chapman and Hall, 2011. – 352 p. DOI: 10.1201/b11032
13. Nedashkovskaya N. I. Method for Evaluation of the Uncertainty of the Paired Comparisons Expert Judgements when Calculating the Decision Alternatives Weights / N. I. Nedashkovskaya // Journal of Automation and Information Sciences. – 2015. – Vol. 47, Issue 10. – P. 69–82. DOI: 10.1615/JAutomatInfScien.v47.i10.70
14. Durbach I. The analytic hierarchy process with stochastic judgements / I. Durbach, R. Lahdelma, P. Salminen // Euro-

DOI 10.15588/1607-3274-2022-3-15

pean Journal of Operational Research. – 2014. – Vol. 238, No. 2. – P. 552–559. DOI: 10.1016/j.ejor.2014.03.045
15. Wang Y.-M. An eigenvector method for generating normalized interval and fuzzy weights / Y.-M. Wang, K.-S. Chin // Applied Mathematics and Computation. – 2006. – Vol. 181. – P. 1257–1275. DOI: 10.1016/j.amc.2006.02.026
16. Krejčí J. Fuzzy eigenvector method for obtaining normalized fuzzy weights from fuzzy pairwise comparison matrices / J. Krejčí // Fuzzy Sets and Systems. – 2017. – Vol. 315. – P. 26–43. DOI: 10.1016/j.fss.2016.03.006
17. Ramík J. Deriving priority vector from pairwise comparisons matrix with fuzzy elements / J. Ramík // Fuzzy Sets and Systems. – 2021. – Vol. 422. – P. 68–82. DOI: 10.1016/j.fss.2020.11.022
18. Mikhailov L. Deriving priorities from fuzzy pairwise comparison judgements / L. Mikhailov // Fuzzy Sets and Systems. – 2003. – Vol. 134, Issue 3. – P. 365–385. DOI: 10.1016/S0165-0114(02)00383-4
19. Mikhailov L. A fuzzy approach to deriving priorities from interval pairwise comparison judgements / L. Mikhailov // European Journal of Operational Research. – 2004. – Vol.159, Issue 3. – P. 687–704. DOI: 10.1016/S0377-2217(03)00432-6
20. Sugihara K. Interval priorities in AHP by interval regression analysis / K. Sugihara, H. Ishii, H. Tanaka // European Journal of Operational Research. – 2004. – Vol. 158. – P. 745–754. DOI: 10.1016/S0377-2217(03)00418-1
21. Wang Y.-M. A two-stage logarithmic goal programming method for generating weights from interval comparison matrices / Y.-M. Wang, J.-B. Yang, D.-L. Xu // Fuzzy Sets and Systems. – 2005. – Vol. 152, Issue 3. – P. 475–498. DOI: 10.1016/j.fss.2004.10.020
22. Wang Y.-M. A goal programming method for obtaining interval weights from an interval comparison matrix / Y.-M. Wang, T. M. S. Elhag // European Journal of Operational Research. – 2007. – Vol. 177, Issue 1. – P. 458–471. DOI: 10.1016/j.ejor.2005.10.066
23. Liu F. Acceptable consistency analysis of interval reciprocal comparison matrices / F. Liu // Fuzzy Sets and Systems. – 2009. – Vol. 160, Issue 18. – P. 2686–2700. DOI: 10.1016/j.fss.2009.01.010
24. Entani T. Pairwise comparison based interval analysis for group decision aiding with multiple criteria / T. Entani, M. Inuiguchi // Fuzzy Sets and Systems. – 2015. – Vol. 274. – P. 79–96. DOI: 10.1016/j.fss.2015.03.001
25. Pankratova N. D. Estimation of decision alternatives on the basis of interval pairwise comparison matrices / N. D. Pankratova, N. I. Nedashkovskaya // Intelligent Control and Automation. – 2016. – Vol. 7, Issue 2. – P. 39–54. DOI: 10.4236/ica.2016.72005
26. Li K. W. Acceptability analysis and priority weight elicitation for interval multiplicative comparison matrices / K. W. Li, Z.-J. Wang, X. Tong // European Journal of Operational Research. – 2016. – Vol. 250, Issue 2. – P. 628–638. DOI: 10.1016/j.ejor.2015.09.010
27. Nedashkovskaya N. I. Investigation of methods for improving consistency of a pairwise comparison matrix / N. I. Nedashkovskaya // Journal of the Operational Research Society. – 2018. – Vol. 69, Issue 12. – P. 1947–1956. DOI: 10.1080/01605682.2017.1415640
28. Pankratova N. D. Method for processing fuzzy expert information in prediction problems. Part I / N. D. Pankratova, N. I. Nedashkovskaya // Journal of Automation and Information Sciences. – 2007. – Vol. 39, Issue 4. – P. 22–36. DOI: 10.1615/jautomatinfscien.v39.i4.30
29. Kuo T. Interval multiplicative pairwise comparison matrix: Consistency, indeterminacy and normality / T. Kuo // Information Sciences. – 2020. – Vol. 517. – P. 244–253. DOI: 10.1016/j.ins.2019.12.066
30. Liu F. A new method of obtaining the priority weights from an interval fuzzy preference relation / F. Liu, W. G. Zhang, J. H. Fu // Information Sciences. – 2012. – Vol. 185, Issue 1. – P. 32–42. DOI: 10.1016/j.ins.2011.09.019
31. Xu Y. Consistency test and weight generation for additive interval fuzzy preference relations / Y. Xu, K. W. Li, H. Wang // Soft Computing. – 2014. – Vol. 18, Issue 8. – P. 1499–1513. DOI: 10.1007/s00500-013-1156-x
32. Wang Z-J. And-like-uninorm-based transitivity and analytic hierarchy process with interval-valued fuzzy preference relations / Z.-J. Wang, X. Yang, X.-T. Jin // Information Sciences. – 2020. – Vol. 539. – P. 375–396. DOI: 10.1016/j.ins.2020.05.052
33. Wang Z.-J. A goal programming approach to deriving interval weights in analytic form from interval Fuzzy preference relations based on multiplicative consistency / Z.-J. Wang // Information Sciences. – 2018. – Vol. 462. – P. 160–181. DOI: 10.1016/j.ins.2018.06.006
34. López-Morales V. A reliable method for consistency improving of interval multiplicative preference relations expressed under uncertainty / V. López-Morales // International Journal of Information Technology & Decision Making. – 2018. – Vol. 17, Issue 5. – P. 1561–1585. DOI: 10.1142/S0219622018500359
35. Wang Z.-J. Consistency and optimized priority weight analytical solutions of interval multiplicative preference relations / Z.-J. Wang, J. Lin // Information Sciences. – 2019. – Vol. 482. – P. 105–122. DOI: 10.1016/j.ins.2019.01.007.
36. Nedashkovskaya N. I. The M_Outflow Method for Finding the Most Inconsistent Elements of a Pairwise Comparison Matrix / N. I. Nedashkovskaya // System Analysis and Information Technologies (SAIT) : 17th International Conference, Kyiv, 2015 : proceeding. – Kyiv : NTUU KPI, 2015. – P. 90. http://sait.kpi.ua/media/filer_public/f8/7e/f87e3b7b-b254-407f-8a58-2d810d23a2e5/sait2015ebook.pdf
37. Buckley J. J. Fuzzy hierarchical analysis / J. J. Buckley // Fuzzy Sets and Systems. – 1985. – Vol. 17, Issue 3. – P. 233–247. DOI: 10.1016/0165-0114(85)90090-9
38. Van Laarhoven P.J.M. A fuzzy extension of Saaty’s priority theory / P. J. M. Van Laarhoven, W. Pedrycz // Fuzzy Sets and Systems. – 1983. – Vol. 11, Issues 1–3. – P. 229–241. DOI: 10.1016/S0165-0114(83)80082-7
39. Wang Y.-M. A modified fuzzy logarithmic least squares method for fuzzy analytic hierarchy process / Y.-M. Wang, T. M. S. Elhag, Z. Hua // Fuzzy Sets and Systems. – 2006. – Vol. 157, Issue 23. – P. 3055–3071. DOI: 10.1016/j.fss.2006.08.010
40. A new method for deriving priority weights by extracting consistent numerical-valued matrices from interval-valued fuzzy judgement matrix / [F. Zhang, J. Ignatius, C. P. Lim et al.] // Information Sciences. – 2014. – Vol. 279. – P. 280–300. DOI: 10.1016/j.ins.2014.03.120

DOI 10.15588/1607-3274-2022-3-15

УДК 658.51.012

СИНТЕЗ АЛГОРИТМА УПРАВЛЕНИЯ ТРАНСПОРТНЫМ КОНВЕЙЕРОМ

Пигнастый О. М. – д-р техн. наук, профессор, профессор кафедры распределенных информационных систем и облачных технологий Национального технического университета «Харьковский политехнический институт», г. Харьков, Украина.

Ивановская О. В. – канд. техн. наук, доцент, доцент кафедры композиционных конструкций и авиационного материаловеденья Национального аэрокосмического университета им. Н. Е. Жуковского «Харьковский авиационный институт», г. Харьков, Украина.

АННОТАЦИЯ

Актуальность. Рассмотрена проблема оптимального управления потоковыми параметрами транспортной системы конвейерного типа при наличие ступенчатого регулировании скорости ленты секции конвейера. Объектом исследования являлась аналитическая модель транспортного конвейера, которая использована в качестве фундамента для синтеза алгоритмов оптимального управления потоковыми параметрами транспортного конвейера. Цель работы заключается в развитие методов проектирования систем оптимального управления потоковыми параметрами транспортного конвейера с учетом транспортной задержки при ступенчатом регулировании потоковых параметров транспортной системы.

Метод. Разработана аналитическая модель секции конвейера с учетом ступенчатого регулирования значений потоковых параметров транспортной системы. При построения модели секции конвейера для определения зависимостей между потоковыми параметрами транспортной системы записаны уравнения в частных производных. Для синтеза алгоритмов оптимального управления скоростью движения ленты секции конвейера введен критерий качества управления. Используя принцип максимума Понтрягина, поставлена задача оптимального управления потоковыми параметрами секции конвейера. Для транспортной системы записана функция Гамильтона, учитывающая критерий качества управления, наложенные ограничения и дифференциальные связи между параметрами системы. Продемонстрирована методика синтеза алгоритма оптимального управления скоростью ленты секции конвейера и определены условия переключения скорости движения ленты.

Результаты. Разработанная модель секции конвейера использована для синтеза алгоритма оптимального управления потоковыми параметрами транспортной системы при ступенчатом переключении режимов скорости ленты.

Выводы. Разработана методика синтеза алгоритмов оптимального управления потоковыми параметрами транспортной системы при ступенчатом регулировании скорости движения ленты секции конвейера. Полученные алгоритмы могут быть использованы для снижения удельных энергетических затрат на транспортировку материала на предприятиях горнодобывающей промышленности.

КЛЮЧЕВЫЕ СЛОВА: конвейер, распределенная система, PDE-модель, поточная линия, производственная линия, регулирование скорости ленты.

АББРЕВИАТУРЫ

PDE-model – аналитическая модель непрерывного представления движения материала по технологическому маршруту производственной линии с использованием уравнений в частных производных;

PiKh-model – аналитическая модель конвейерной линии, позволяющая определить плотность материала и расход материала в произвольной точке конвейерной ленты.

НОМЕНКЛАТУРА

$a(t)$ – скорость ленты конвейерной секции;

$a_{\max}$ – максимальное значение скорости ленты конвейерной секции;

a_j – значение *j*-ой ступени регулирования скорости ленты секции конвейера;

a_n – номинальное значение скорости ленты;

C – масса погонного метра движущихся частей;

G^{-1} – функция, обратная к функции $G(\tau)$;

$E(M_{\max})$ – затраты энергии на транспортировку тонны породы на расстояние километра при максимальной загрузки конвейера $M_{\max}$;

$E(M_i)$ – затраты энергии на транспортировку тонны породы на расстояние километра при загрузки конвейера M_i;

$H(S)$ – функция Хевисайда;

$M_{\max}$ – транспортируемая масса груза на ленте секции конвейера, соответствующая максимальной загрузке конвейера по приемной способности;

N_s – количество ступеней регулирования скорости ленты секции конвейера;

N_{xx} – мощность, потребляемая конвейером на холостом ходу;

n_1 – приращение потребляемой мощности при увеличении массы груза на конвейере на одну тонну;

P_j – доля времени, в течение которого секция конвейера функционирует со скоростью ленты a_j;

S_{dk} – длина секции (*k*-ой секции) конвейера;

S – координата, характеризующая местоположение материала вдоль маршрута транспортировки $S \in [0, S_d]$;

S_d – длина транспортного маршрута конвейера;

DOI 10.15588/1607-3274-2022-3-16

$T(t)$ – величина транспортной задержки;

T_d – характерное время транспортировки материала вдоль секции конвейера;

α ($\sec^{-1}$) – величина, обратно пропорциональная времени корреляции;

β – угол наклона секции конвейера;

$\delta(S)$ – функция Дирака;

Θ – максимально допустимое значение линейной плотности материала для секции транспортного конвейера;

$\lambda(t)$ – интенсивность поступления материала на вход поточной линии;

σ_Q^2 – дисперсия входного грузопотока материала;

$\Psi(S)$ – линейная плотность распределения горной породы на конвейерной ленте в начальный момент времени $t = 0$;

$[\chi]_0(t,S)$ – линейная плотность распределения материала вдоль конвейерной линии в момент времени t в точке транспортного маршрута с координатой S;

$[\chi]_{0\max}(t,S)$ – максимальное значение линейной плотности материала вдоль секции конвейера;

$[\chi]_{1\text{in}}(t)$ – грузопоток, поступающий на вход секции конвейера;

$[\chi]_{1\text{t}}(t)$ – экспериментальное значение грузопотока поступающего на вход секции конвейера;

$[\chi]_1(t,S)$ – поток материала в момент времени t в точке транспортного маршрута с координатой S;

$[\chi]_{1\psi}(t,S)$ – нормативная производительность обработки деталей в момент времени t для технологической позиции S;

w – коэффициент сопротивления движения;

$\Delta\tau$ – транспортная задержка для секции конвейера.

ВВЕДЕНИЕ

Конвейер является основным способом транспортировки породы на горнодобывающих предприятиях. При протяженности в несколько десятков километров и пропускной способности ~10^4 t/h потребляемая мощность конвейерных транспортных систем ~10^5 kW [1]. Нормативный расход электроэнергии конвейерным транспортом для перемещения тонны горной породы на расстояние в 1,0 km оценивается $(0{,}1 \div 1{,}0)\dfrac{kW \cdot h}{t \cdot km}$ [1, 2]: Çöllolar Lignite Open Pit Mine (Turkey, 2010) $0{,}39\dfrac{kW \cdot h}{t \cdot km}$, Coarse ore conveyor system Minera Los Pelambres (Chile, 1998) $0{,}79\dfrac{kW \cdot h}{t \cdot km}$, Belt conveyor with gearless drive Solution Prosper Haniel Coal Mine (Germany, 2010) $0{,}24\dfrac{kW \cdot h}{t \cdot km}$ [1], а общие затраты энергии для перемещения тонны породы вдоль всей конвейерной линии соответственно $\sim 6{,}79\ kW \cdot h$, $\sim 10{,}03\ kW \cdot h$, $\sim 3{,}24\ kW \cdot h$. Фактические затраты энергии существенно выше в силу не полной загруженности конвейерной линии из-за неравномерного поступления породы на вход конвейера. Исследованию зависимости затрат энергии для транспортирование груза от величины грузопотока посвящено достаточное количество работ, начиная с [3–6]. Теоретическое и экспериментальное исследования конвейера показали, что уменьшение загруженности конвейера приводит к гиперболическому увеличению затрат энергии на транспортирование. При коэффициенте загруженности конвейера 0,75 затраты энергии на транспортирование увеличиваются на 10%, при 0,5– на 50%, при 0,25–на 160% и при 0,1– на 675% [7]. Такой характер зависимости вызывает существенный практический и теоретический интерес исследователей к указанной проблеме. Предложенные методы, позволяющие обеспечить снижение затрат энергии конвейерной системы до 30%, в большинстве случаях основываются на использовании аккумулирующего буфера для устранения неравномерности загрузки ленточного конвейера [8–11]. Бункер обеспечивает равномерную подачу породы на вход линии, аккумулирует излишек породы в накопителе, если темп поступления выше нормативного, или, если темп ниже нормативного, использует запасы породы в бункере для ликвидации отклонения. При этом скорость ленты, как правило, остается неизменной. Использование бункера в целях стабилизации темпа поступления породы на конвейерную линию требует дополнительного технологического пространства для его размещения [9, 10]. Другая проблема, которую следует учитывать при проектирование конвейерных систем с бункером – ограниченность емкости бункера, что требует наличия системы управления, которая не допустит переполнение [12]. Альтернативный подход повышения эффективности конвейерного транспорта заключается в проектировании систем управления скоростью ленты. При уменьшении потока породы на конвейер скорость ленты уменьшают, обеспечивая нормативную линейную плотность породы на конвейерной ленты, что приводит к аккумулированию горной породы вдоль конвейера. При увеличении потока породы на входе скорость ленты увеличивается, что приводит к снижению значения линейной плотности породы.

Объект исследования – аналитическая модель транспортного конвейера.

Предмет исследования – аналитические методы проектирования системы управления транспортным конвейером.

Цель работы – развитие методов проектирования систем оптимального управления потоковыми параметрами транспортного конвейера.

DOI 10.15588/1607-3274-2022-3-16

1 ПОСТАНОВКА ЗАДАЧИ

Пусть для секции транспортного конвейера с целью синтеза оптимального управления потоковыми параметрами транспортной системы на интервале управления $\tau \in [0, \tau_k]$ задан количественный критерий качества управления

$$K(\tau) = \int_0^{\tau_k} J(\tau, u(\tau)) d\tau ,$$

который является показателем эффективного функционирования секции конвейера. Экстремальное значение критерия качества управления достигается при оптимальном управлении $u(\tau)$ скоростью конвейерной ленты.

Для заданного критерия качества управления сформулируем задачу синтеза оптимальной программы управления скоростью ленты для установившегося режима функционирования секции конвейера $\tau > \tau_d$:

– требуется определить управление скоростью движения конвейерной ленты $u(\tau)$ на временном интервале $\tau \in [0, \tau_k]$, которое приводит к минимуму функционал

$$K(\tau) = \int_0^{\tau_k} J(\tau, u(\tau)) d\tau \to \min ,$$

а) при дифференциальных связях вида

$$\frac{\partial \theta_0(\tau, \xi)}{\partial \tau} + u(\tau) \frac{\partial \theta_0(\tau, \xi)}{\partial \xi} = \delta(\xi) \gamma(\tau) ,$$

которые соответствуют аналитической PDE-model производственной системы;

б) при ограничениях на погонную плотность материала секции конвейера и максимально допустимую нагрузку на привод конвейера

$$\theta_{0\max} \ge \theta_0(\tau, \xi) \ge 0 ,$$

$$m(t) = \int_0^1 \theta_0(\tau, \xi) d\xi \le m_{\max} ;$$

в) при ограничении на управление

$$g_{\max} \ge u(\tau) = g(\tau) ;$$

г) для начального условия старта ленты транспортной секции

$$\theta_0(0, \xi) = H(\xi) \psi(\xi) ,$$

определяемого линейной плотностью начального распределения материала вдоль маршрута транспортировки.

DOI 10.15588/1607-3274-2022-3-16

2 ОБЗОР ЛИТЕРАТУРЫ

Проблема повышения эффективности функционирования конвейерного транспорта за счет регулирования скорости ленты достаточно много обсуждалась специалистами в области проектирования конвейерного транспорта [3]. Однако, несмотря на наличие большого количества работ, данная проблема имеет высокую актуальность и в настоящее время. Конвейер с движущейся горной породой вдоль транспортного маршрута– это распределенная система с рядом ограничений, среди которых не мало важными является ограничение по максимальной удельной линейной нагрузке на конвейерную ленту и ограничение по максимальному объему транспортируемой массы [4]. Конвейерная система является стохастической системой. Неопределенность величины поступающего потока породы на вход конвейера (неопределенность граничных условий) требует при расчете конвейерной линии использовать вероятностные методы [5]. В [13] представлен алгоритм управления скоростью ленты, использующий методы теории математического программирования. В работе записана целевая функция, определены необходимые условия регулирования скорости ленты, представлена имитационная модель конвейерной линии с постоянным значением скорости для j-ой ступени регулирования и инкрементным алгоритмом регулирования скорости. Скорость a_j определяется через величину номинальной погонной нагрузки

$$\psi = \frac{[\chi]_{1n}}{a_n}$$

и значение грузопотока $[\chi]_{1n}$ для *j*-ступени режима регулирования скорости

$$a_j = \frac{[\chi]_{1j}}{\psi} = \frac{[\chi]_{1j}}{[\chi]_{1n}} a_n ,$$

где $[\chi]_{1n}$ – номинальные значения производительности секции конвейера. Расчетное значение средней скорости ленты конвейера за наблюдаемый период определяется выражением

$$\langle a \rangle = \sum_{j=1}^{N_s} a_j P_j , \qquad \sum_{j=1}^{N_s} P_j = 1 .$$

Режимы управления задаются из условия минимума средней скорости движения ленты

$$\frac{\partial \langle a \rangle}{\partial [\chi]_{1j}} = 0 .$$

Статистический анализ эксплуатационных параметров шахтного конвейера выполнен в [14]. При моделировании входного грузопотока использовался нормальный и логарифмический нормальный закон распределения величины шахтного потока материала с корреляционной функцией [4, 14]:

$$R_Q(\tau) = \sigma_Q^2 \exp(-\alpha\tau) \ \ (\text{kg/sec})^2 .$$

Для определения грузопотока материала использованы осциллограммы токов очистных комбайнов. В течение двух суток с шагом одна секунда записывались усилия, действующие на съемное тензоизмерительное устройство, и в течение четырех суток – фактическая мощность привода конвейера при переменном грузопотоке. Гистограмма распределения грузопотока как случайной величины подтвердила гипотезу о логарифмическом нормальном законе распределения грузопотока. Максимальное значение грузопотока составляет 533 (kg/sec). Математическое ожидание и среднеквадратичное отклонение грузопотока, соответственно: 134 kg/sec и 66,4 kg/sec. Для построения зависимости мощности привода и коэффициента сопротивления движению от величины грузопотока определялась масса груза на конвейере в момент времени, соответствующую текущему значению мощности привода:

$$M(t) = \int\limits_{t-4(\text{sec})}^{t+361(\text{sec})} Q(\tau)d\tau, (\text{kg}) .$$

Пределы интегрирования соответствуют моментам времени разгрузки и погрузки взвешиваемой порции груза. Получена зависимость потребляемой мощности и окружного тягового усилия на приводе от величины погонной нагрузки [14]

$$N(t) = N_{xx} + n_1 M(t) .$$

Значения N_{xx} =160 kW, n_1 =1,11 kW/t определены по данным эксперимента. Адекватность математической модели оценена по критерию Фишера. Экспериментальное значение N_{xx} =160 kW в 3 раза больше расчетного значения мощности холостого хода N_{xx} =55 kW [14, 15], что объясняется неудовлетворительным состоянием роликовых опор на шахтном конвейере, трением ленты об элементы конструкции, наличием местных сопротивлений движению в местах перегиба профиля трассы и другими факторами. В работе [16] представлена модель контейнера в виде динамического звена с транспортной задержкой, зависимой от времени. Величина выходного потока породы с конвейера $[\chi]_{1\text{out}}$ определяется через значение входного потока $[\chi]_{1\text{in}}$ и скорость ленты a_{in} :

$$[\chi]_{0\text{out}}(t) = \frac{[\chi]_{1\text{in}}(t-T(t))}{a_{\text{in}}(t-T(t))}, \ [\chi]_{1\text{out}}(t) = [\chi]_{0\text{out}}(t) a_{\text{in}}(t) .$$

Для переменной скорости ленты конвейера величина транспортной задержки определена с использованием численных методов [17]. Масса горной породы, распределенная вдоль маршрута транспортировки, представлена интегральным уравнением

$$M(t) = M(t_n) \int\limits_{t_n}^{t_k} [\chi]_{1\text{in}}(t) \frac{[\chi]_{1\text{in}}(t-T(t))}{a_{\text{in}}(t-T(t))} a_{\text{in}}(t) dt ,$$

дополненным уравнением связи для определения статического тягового усилия F

$$F = a\big(M(t) + b\big),$$

которое необходимо преодолеть приводному электродвигателю. Записанные выше уравнения составляют математическую модель конвейера как элемента системы автоматического регулирования статической нагрузки привода [17]. Коэффициенты a, b учитывают удельную массу ленты, роликов, сцепление между лентой и барабаном, сопротивление движению ленты и другие конструкционные и технологические особенности транспортного конвейера. Для случая, когда входной грузопоток может быть измерен, предложен алгоритм управления скоростью по возмущению величины входного грузопотока [16]. Влияние неравномерной загруженности ленточного конвейера на транспортные затраты изучено в [7]. Произведена оценка энергетических показателей по результатам экспериментальных исследований грузопотока и мощности привода конвейера 2ЛУ120В при длине конвейера L=0,730 km, скорости ленты a=2 m/sec, мощности привода N=500 kW и максимальной производительности $[\chi]_{1\text{imax}}$ =1500 t/h. Масса транспортируемой породы определялся выражением

$$M(t) = M(t_n) \int\limits_{t+t_1}^{t+t_1+t_2} [\chi]_{1\text{t}}(\tau) d\tau ,$$

где t_1 и t_2 – значения времени прохождения груза от места взвешивания до разгрузки и от места загрузки до разгрузки; Регрессионное уравнение для расчета мощности конвейерной установки в зависимости от массы груза на ленте представлено в [14]. Исследована зависимость коэффициента энергопотребления

$$K_e = \frac{E(M_i)}{E(M_{\max})}$$

от коэффициента загрузки и его приемной материалоемкости

DOI 10.15588/1607-3274-2022-3-16

$$K_z = \frac{M_i}{M_{\max}},$$

которая может быть преобразована следующим образом

$$K_e = f\left(K_z\right) \approx \frac{1}{2}\left(\frac{1}{K_z}+1\right).$$

В работе оценено влияние неравномерной загруженности ленты на затраты энергии при транспортировки горной породы по транспортному маршруту. Подчеркнута необходимость построения адекватных математических моделей функционирования конвейера с адаптивной системой управления приводом [7]. Решение данных задач способствует решению проблемы интенсификации угледобычи путем создания для горной промышленности надежных и эффективных конвейеров нового поколения. В работе [18] разработана математическая модель частотно-управляемого электропривода, учитывающая распределенную нагрузку на конвейерной линии. Синтезирован нечеткий регулятор скорости ленты конвейера. Проведен сравнительный анализ переходных процессов в электроприводе конвейера с системой управления, основанной на использовании нечеткого регулятора величины скорости ленты. Распределенная численная модель динамики конвейера на основе конечно-разностного подхода представлена в [19]. Для выполнения численного моделирования лента разбивалась на $10^3 \div 10^4$ элементов, в пределах каждого из которых решается система уравнений, определяющая динамику движения участка ленты, наполненной материалом. В работе [20] введена плотность распределения материала $[\chi]_0(t,S)$, формируемая входным грузопотоком $[\chi]_{1\text{in}}(t)$ и записано уравнение, определяющее значение линейной плотности материала на входе в секцию конвейера

$$[\chi]_0(t,0) = \frac{dm_{in}(t)}{dt}\frac{1}{a(t)} = \frac{[\chi]_{1\text{in}}(t)}{a(t)} = \frac{[\chi]_1(t,0)}{a(t)}.$$

Выходной поток представлен выражением

$$[\chi]_{1\text{out}}(t) = [\chi]_1(t,S_d) = \frac{dm_{out}(t)}{dt} = [\chi]_0(t,S_d)a(t),$$

в котором линейная плотность распределения материала на входе $[\chi]_0(t,0)$ и выходе $[\chi]_0(t,S_d)$ конвейера связана соотношением

$$[\chi]_0(t,S_d) = [\chi]_0(t-T(t),0),$$

$$\int_{t-T(t)}^{t} a(\lambda)d\lambda = S_d.$$

При постоянной скорости $a(t)=a_1$ следует

$$T(t) = \frac{S_d}{a_1} = \text{const}.$$

С учетом последнего допущения, в работе дано выражение для расчета максимально допустимого грузопотока, поступающего на вход конвейерной линии [20]

$$[\chi]_{1\text{out\_max}} = \frac{dm_{out\_\max}}{dt} = [\chi]_{0\max}(t,S)a_{\max}.$$

В работе [21] исследуются методы описания величины случайного грузопотока, поступающего на вход конвейера с использованием эмпирических распределений. В работах [22, 23] с применением моделей теории массового обслуживания выполнен анализ шахтного потока материала. В [22] разработана модель магистрального сборного конвейера для двух очистных забоев, поток материала с каждого из забоев является случайной величиной. В [23] рассмотрено влияние регулируемого привода на динамику потоков материала и эффективность функционирования шахтного конвейера. Представлена модель транспортной системы, состоящей из нескольких последовательно расположенных конвейеров с регулирование скорости движения ленты. Обоснован выбор длины конвейеров в предложенной модели [23]. Практический интерес представляет модель транспортной сети при слиянии потоков с регулируемыми и нерегулируемыми приводами [23]. Исследованы законы распределения потоков, которые формируются конвейерами с регулируемой скоростью. В работе [24] установлена зависимость коэффициента сопротивления движению ленты от величины погонной нагрузки, динамика изменения которой имеет достаточно сложный характер. Эта зависимость играет принципиально важную роль при оценке затрат энергии на транспортирование груза вдоль секции конвейером с постоянной и регулируемой скоростью, нуждается в экспериментальной проверке в шахтных условиях эксплуатации.

3 МАТЕРИАЛЫ И МЕТОДЫ

Уравнения в частных производных для моделирования движения материала в производственных поточных линий применены в [25]. В работах [26, 27] дан обзор моделей поточных линий, где особое внимание уделено моделям с использованием уравнений в частных производных (PDE-модели). Новый класс моделей производственных систем предназначен для описания производственных поточных линий, функционирующих в стационарных и переходных режимах, позволяет моделировать распределение деталей вдоль поточной линии и учитывает стохастический характер взаимодействия предметов труда с оборудованием и между собой в результате технологической

DOI 10.15588/1607-3274-2022-3-16

обработки при переходе от одной технологической операции к другой технологической операции.

Система уравнений, определяющая динамику состояния потоковых параметров производственной линии в одномоментном описании, может быть представлена в виде [28]:

$$\frac{\partial[\chi]_0(t,S)}{\partial t}+\frac{\partial[\chi]_1(t,S)}{\partial S}=\delta(S)\lambda(t),$$

$$\int_{-\infty}^{\infty}\delta(S)dS=1,$$

$$[\chi]_1(t,S)=[\chi]_{1\psi}(t,S).$$

Дополним систему уравнений начальным условием

$$[\chi]_0(t_0,S)=H(S)\Psi(S),$$

$$H(S)=\begin{cases}0, \text{ if } S<0,\\ 1, \text{ if } S\geq 0,\end{cases}$$

которое задает распределение деталей по технологическим позициям в момент времени $t=t_0$. Решение системы уравнений определяется начальным условием $[\chi]_0(t_0,S)$ и известной нормативной производительностью обработки деталей в соответствие с технологическим процессом $[\chi]_{1\psi}(t,S)$ [25; 28].

Конвейерная линия является одной из простейших разновидностей поточной линии. Особенность построения модели конвейерной линии заключается в том, что размещенная на конвейерной ленте в разных местах, определенных координатой S, горная порода движется с одной и той же скоростью $a(t)$, которая равна скорости движения ленты конвейера. Это позволяет представить систему уравнений для потоковых параметров транспортной системы в следующем виде:

$$\frac{\partial[\chi]_0(t,S)}{\partial t}+a(t)\frac{\partial[\chi]_0(t,S)}{\partial S}=\delta(S)\lambda(t), \quad (1)$$

$$[\chi]_1(t,S)=a(t)[\chi]_0(t,S). \quad (2)$$

Дополним систему уравнений (1), (2) начальным условием

$$[\chi]_0(t_0,S)=H(S)\Psi(S), \quad (3)$$

которое определяет распределение материала вдоль маршрута транспортировки в момент времени $t_0=0$.

Функция $\lambda(t)$ задает поток материала на входе в секцию конвейера. Система уравнений (1)–(3) определяет модель конвейера, которую используем для построения алгоритма оптимального управления скоростью ленты.

DOI 10.15588/1607-3274-2022-3-16

Введем безразмерные параметры и функции:

$$\tau=\frac{t}{T_d},\ \xi=\frac{S}{S_d},\ \delta(\xi)=S_d\delta(S),\ H(\xi S_d)=H(S), \quad (4)$$

$$\psi(\xi)=\frac{\Psi(S)}{\Theta},\ \gamma(\tau)=\lambda(t)\frac{T_d}{S_d\Theta},\ g(\tau)=a(t)\frac{T_d}{S_d}, \quad (5)$$

$$\theta_0(\tau,\xi)=\frac{[\chi]_0(t,S)}{\Theta},\ \Theta=\max\left\{\Psi(S),\frac{\lambda(t)}{a(t)}\right\}. \quad (6)$$

Принимая во внимание безразмерные переменные и функции (4)–(6), запишем уравнения (1)–(3) в безразмерном виде:

$$\frac{\partial\theta_0(\tau,\xi)}{\partial\tau}+g(\tau)\frac{\partial\theta_0(\tau,\xi)}{\partial\xi}=\delta(\xi)\gamma(\tau), \quad (7)$$

$$\theta_0(0,\xi)=\mathrm{H}(\xi)\psi(\xi). \quad (8)$$

Выбор характерного значения Θ позволяет обеспечить масштаб значений функции $\theta_0(\tau,\xi)$. Конвейерные линии, которые имеют одинаковый вид функций $\gamma(\tau)$ и $g(\tau)$ являются подобными, имеют одинаковое поведение при подобном начальном условии $\psi(\xi)$. Это дает возможность строить лабораторные аналоги крупных конвейерных линий для проведения практических экспериментов. Результаты, полученные на модельных конвейерных линиях, могут быть с достаточной степенью точностью перенесены на действующие конвейерные линии, что позволяет сэкономить средства, требуемые для проведения экспериментальных исследований. Особенно актуальным является использование методов теории подобия в тех случаях, когда отсутствует возможность проводить экспериментальные исследования действующих конвейерных линий в связи с их загрузкой производственной программой. Системе уравнений в частных производных (7), (8) соответствует система характеристических уравнений:

$$\frac{d\xi}{d\tau}=g(\tau), \quad (9)$$

$$\xi\big|_{\tau=0}=\beta,$$

$$\frac{d\theta_0(\tau,\xi)}{d\xi}=\delta(\xi)\frac{\gamma(\tau)}{g(\tau)}, \quad (10)$$

$$\theta_0(0,\beta)=\mathrm{H}(\beta)\psi(\beta).$$

Решение уравнения (9) представим в следующем виде

$$\xi = G(\tau) + C_1 = G(\tau) - G(0) + \beta , \qquad (11)$$

$$G(\tau) = \int g(\tau) d\tau ,$$

где константа интегрирования

$$C_1 = \beta - G(0)$$

определяется из начального условия (9).

В результате интегрирования уравнения (10) получается выражение для безразмерной линейной плотности материала $\theta_0(\tau,\xi)$ вдоль маршрута транспортировки в форме:

$$\theta_0(\tau,\xi) = \int \delta(\xi) \frac{\gamma(\tau)}{g(\tau)} d\xi + C_2 =$$
$$= H(\xi) \frac{\gamma\left(G^{-1}(G(\tau)-\xi)\right)}{g\left(G^{-1}(G(\tau)-\xi)\right)} + C_2 .$$

Определив константу интегрирования C_2 в результате решения уравнения (9):

$$\theta_0(0,\beta) = H(\beta) \frac{\gamma\left(G^{-1}(G(0)-\beta)\right)}{g\left(G^{-1}(G(0)-\beta)\right)} + C_2 = H(\beta)\psi(\beta) ,$$

запишем выражение для безразмерной линейной плотности материала $\theta_0(\tau,\xi)$:

$$\theta_0(\tau,\xi) = H(\xi) \frac{\gamma\left(G^{-1}(G(\tau)-\xi)\right)}{g\left(G^{-1}(G(\tau)-\xi)\right)} -$$
$$- H(\beta) \frac{\gamma\left(G^{-1}(G(0)-\beta)\right)}{g\left(G^{-1}(G(0)-\beta)\right)} + H(\beta)\psi(\beta) .$$

Из уравнения (11) выразим переменную β

$$\beta = \xi - \int_0^\tau g(z) dz ,$$

получим решение уравнения (7) при начальном условии (8):

$$\theta_0(\tau,\xi) = \left[H(\xi) - H\left(\xi - \int_0^\tau g(z)dz \right) \right] \frac{\gamma(G^{-1}(G(\tau)-\xi))}{g(G^{-1}(G(\tau)-\xi))} +$$

$$+ H\left(\xi - \int_0^\tau g(z)dz) \right) \psi \left(\xi - \int_0^\tau g(z)dz) \right). \qquad (12)$$

Уравнение (12) позволяет вычислить плотность горной породы для момента времени τ в точке транспортного маршрута с координатой ξ. Полученное решение определяет неравномерность распределения материала вдоль маршрута транспортировки.

Уравнение движения в виде

$$\xi - \int_0^\tau g(z) dz = 0 ,$$

задает закон $\xi(\tau)$ перемещения элемента конвейерной ленте, справа от которой находится материал, линейная плотность которого задана начальными условиями (9), (10). Общее время τ_d переходного режима, при котором начальные условия оказывают влияние на значение выходного потока материала с конвейерной линии, может быть получено в результате решения уравнения

$$1 = \int_0^{\tau_d} g(z) dz . \qquad (13)$$

Неравенство

$$\tau > \tau_d$$

задает режим работы конвейерной линии, когда ее параметры не зависят от начальных условий (8)

$$\theta_0(\tau,\xi) = \frac{\gamma(G^{-1}(G(\tau)-\xi))}{g(G^{-1}(G(\tau)-\xi))} .$$

Если ввести обозначение

$$G^{-1}(G(\tau)-1) = \tau - \Delta\tau_1 ,$$

то для рассмотренного случая на выходе конвейерной линии $\xi = 1$ имеем:

$$\theta_0(\tau,1) = \frac{\gamma(\tau-\Delta\tau_1)}{g(\tau-\Delta\tau_1)} = \frac{\theta_1(\tau-\Delta\tau_1,0)}{g(\tau-\Delta\tau_1)} = \theta_0(\tau-\Delta\tau_1,0) ,$$

$$\theta_1(\tau,1) = \frac{\gamma(\tau-\Delta\tau_1)}{g(\tau-\Delta\tau_1)} g(\tau) = \theta_0(\tau-\Delta\tau_1,0) g(\tau) , \ \tau > \tau_d .$$

В предположении того, что скорость ленты секции конвейера постоянна

$$g(\tau) = g_0 = \text{const} ,$$

решение уравнения (13) позволяет определить длительность задержки [20]

$$\tau_d = \frac{1}{g_0} .$$

DOI 10.15588/1607-3274-2022-3-16

Представленная взаимосвязь между входными и выходными потоковыми параметрами конвейера использована для построения модели секции конвейера [16, 20]. При этом предполагалось, что в начальный момент времени конвейер является незаполненным [16]

$$\theta_0(0,\xi) = \Psi(S) = 0\,, \qquad (14)$$

откуда из выражения (12) следует частное решение:

$$\theta_0(\tau,\xi) = \left[H(\xi) - H\left(\xi - \int_0^{\tau} g(z)dz \right) \right] \frac{\gamma(G^{-1}(G(\tau)-\xi))}{g(G^{-1}(G(\tau)-\xi))}\,,$$

$$\theta_1(\tau,\xi) = \theta_0(\tau,\xi) g(\tau)\,,$$

которое позволяет рассчитать линейную плотность материала $\theta_0(\tau,\xi)$ и поток материала $\theta_1(\tau,\xi)$ в произвольный момент времени τ в заданной точке ξ транспортного маршрута. Начальные условия заполнения конвейерной линии (14) объясняют наличие колебаний значения скорости ленты секции конвейера и значения выходного потока в исследовании [16], в котором рассмотрено моделировании процесса транспортировки материала вдоль секции конвейера с продолжительностью переходного режима составила 250 sec при общей продолжительности эксперимента 2000 sec и критерии управления для моментов времени $t > T_d = 250$ sec [16]

$$\mathrm{M}(t) = \mathrm{M}_{\max} = \mathrm{const}\,. \qquad (15)$$

Разброс амплитуды колебания величины скорости ленты и значения потока материала на выходе конвейера составляет ~30% на протяжении промежутка времени имитационного моделирования [16]. Амплитуда и период колебания потоковых параметров секции конвейера обусловлены начальными условиями (14) и критерием управления (15). Для момента времени $\tau < \tau_d$ выходной поток материала с секции конвейера определяется начальным распределением материала вдоль транспортного маршрута.

Расход электроэнергии на конвейерный транспорт $W_{k,l}$ kWh определяется количеством транспортируемого груза Q_p(t) и средней продолжительности работы конвейерной линии t_p h [2]:

$$W_{k,l} = 0{,}013\left(\frac{\mathrm{kW\cdot sec}}{kg\cdot m}\right) L_k \cdot w \cdot \Big[C \cdot v_k \cdot t_p + $$
$$+ 0{,}28\left(\frac{\mathrm{kg}\cdot h^2}{\mathrm{sec}\cdot t}\right) \cdot Q_p \cdot \left(1 \pm \frac{\sin\beta}{w}\right) \Big].$$

Для стационарных конвейеров $w = 0{,}02 \div 0{,}03$; для конвейеров выемочных участков $w = 0{,}04 \div 0{,}06$; для конвейеров, работающих в тяжелых условия $w = 0{,}08 \div 0{,}12$. Потребляемая мощность стационарным конвейером ($w = 0{,}03$), расположенным горизонтально ($\beta = 0$) определяется выражением:

$$N(t) = \frac{W_{k,l}}{t_p} = 0{,}0039\left(\frac{\mathrm{kW\cdot sec}}{kg\cdot m}\right) \mathrm{S_d}\Big[C\cdot a(t) + $$
$$+ 1008\left(\frac{\mathrm{kg}}{t}\right) \frac{1}{S_d} \int_0^{S_d} [\chi]_1(t,S)dS \Big],$$

где dQ_p/dt t/h– средняя пропускная способность конвейерной линии. Мощность, затрачиваемая на транспортировку породы представляется в виде суммы мощностей на перемещение движущихся частей конвейера и горной породы массой $M(t)$

$$N(t) = 0.0039\left(\frac{\mathrm{kW\cdot sec}}{kg\cdot m}\right) S_d C a(t) + $$
$$+ 0{,}3931\left(\frac{\mathrm{kW\cdot sec}}{t\cdot m}\right) a(t) M(t)\,,\ \mathrm{kW} \qquad (16)$$
$$M(t) = \int_0^{S_d} [\chi]_0(t,S)dS.$$

Используя (16), определим затраты энергии, требуемые для функционирования конвейерной линии 2LU120V [2] длиной $S_d = 730$ m при средней скоростью движения ленты $a(t) = 2.0$ m/sec [7, 14]:

$$N_{xx} = 0.0039\left(\frac{\mathrm{kW\cdot sec}}{kg\cdot m}\right) 730(m)138{,}1\left(\frac{kg}{m}\right)2{,}0\left(\frac{m}{\mathrm{sec}}\right) = $$
$$= 78{,}63\ \mathrm{kW},$$

$$n_1 = 0{,}3931\left(\frac{\mathrm{kW\cdot sec}}{t\cdot m}\right) 2{,}0(m/\mathrm{sec}) = 0{,}786\ \mathrm{kW/t}.$$

Для конвейера 2LU120V, предназначенного для перемещения породы с нормативной скоростью $a(t) = 3{,}15$ m/sec, максимальной производительностью $dQ_p/dt = 1450$ t/h и массой погонного метра движущихся частей $\mathrm{C} = 138.1$ kg/m потребление энергии определяется следующим выражением

$$N(t) = 123{,}84 + 1{,}23\,M(t)\ \mathrm{kW}.$$

В расчетах использован коэффициент сопротивления ($w = 0{,}03$) и угол наклона ($\beta = 0$) при постоянной скорости $a(t) = 3{,}15$ m/sec ленты конвейера. Максимальная мощность, которую потребляет конвейер для перемещения только материала с равномерно распределенной нагрузкой Θ по длина конвейера $\mathrm{S_d}$ и скоростью движения ленты S_d/T_d (4), (5) определяется в соответствие с формулой

DOI 10.15588/1607-3274-2022-3-16

$$N_{\Theta\max} = 0{,}3931\left(\frac{\text{kW}\cdot\text{sec}}{kg\cdot m}\right)\frac{S_d}{T_d}\left(\frac{m}{\text{sec}}\right)S_d(m)\Theta\left(\frac{t}{m}\right),$$

$$\frac{N(t)}{N_{\Theta\max}} = m_c g(t) + m(t)g(t), \qquad (17)$$

$$m(t) = \int_0^1 \theta_0(\tau,\xi)d\xi = \frac{M(t)}{M_{\max}},\; m_c = 10^{-3}\frac{C}{\Theta}\left(\frac{t}{kg}\right).$$

Для максимального значения транспортируемой массы материала на конвейера $M_{\max} = 126$ t [7] следует:

$$\Theta = \frac{126}{730} = 0{,}172 \text{ t/m},$$

$$m_c = 10^{-3}\frac{138}{0{,}172} = 0{,}79 \text{ t/kg}.$$

Разделив левую и правую часть безразмерного соотношения для мощности (15) на коэффициент рациональной загрузки ленты

$$m(t) = \frac{M(t)}{M_{\max}},$$

получим затраты на единицу массы для фактического и номинального режима загрузки

$$\frac{N(t)/M(t)}{N_{\Theta\max}/M_{\max}} = \frac{m_c}{m(t)}g(t) + g(t) \qquad (18)$$

Последнее выражение будем использовать в качестве критерия качества для управляемого процесса. При $g(t) = 0{,}5$ и $m_c = 0{,}79$ критерий оптимальности имеет вид

$$\frac{N(t)/M(t)}{N_{\Theta\max}/M_{\max}} = \frac{0{,}4}{\int_0^1 \theta_0(\tau,\xi)d\xi} + 0{,}5. \qquad (19)$$

Принимая во внимание систему характеристических уравнений (9), (10), дифференциальные связи принимают следующий вид:

$$\frac{d\xi}{d\tau} = u(\tau), \qquad \xi\big|_{\tau=0} = \beta,$$

$$\frac{d\theta_0(\tau,\xi)}{d\xi} = \delta(\xi)\frac{\gamma(\tau)}{u(\tau)},$$

$$\theta_0(0,\beta) = \mathrm{H}(\beta)\psi(\beta),$$

с учетом которых можно записать функцию Понтрягина и сопряженную систему уравнений для поставленной задачи

$$H = \psi_1 u(\tau) - u(\tau) - \frac{u(\tau)m_c}{\int_0^1 \theta_0(\tau,\xi)d\xi} \to \max,$$

$$\frac{d\psi_1}{dt} = \frac{\partial H}{\partial \xi}.$$

Значение переменной $\theta_{0\max}$ определяет максимально допустимую погонную нагрузку на ленту. Так как правый конец фазовой траектории свободен, то

$$\psi_1(\tau_k) = 0,$$

и, следовательно:

$$\psi_1(\tau) = 0,$$

откуда выражение для функции Понтрягина принимает упрощенный вид

$$H = -u(\tau) - \frac{u(\tau)m_c}{\int_0^1 \theta_0(\tau,\xi)d\xi} \to \max.$$

Дифференциальная связь (10) позволяет для установившегося режима функционирования конвейера $\tau > \tau_d$ выразить плотность распределения горной породы вдоль конвейера через входящий поток и скорость движения ленты

$$\theta_0(\tau,\xi) = \frac{\gamma(G^{-1}(G(\tau)-\xi))}{u(G^{-1}(G(\tau)-\xi))}.$$

Рассмотрим построение оптимального алгоритма управления скоростью ленты конвейера для случая, когда входной поток является постоянным

$$\gamma(\tau) = \gamma_0 = \text{const}. \qquad (20)$$

Масса горной породы на конвейерной ленте определяется выражением:

$$m(t) = \int_0^1 \theta_0(\tau,\xi)d\xi = \sum_{j=1}^{N_s}\theta_0(\tau,\xi_j)\Delta\xi_j =$$

$$= \sum_{j=1}^{N_s}\frac{\gamma_{0\Delta\xi_j}}{u(G^{-1}(G(\tau)-\xi_j))} = \gamma_0\sum_{j=1}^{N_s}\Delta\tau_j = \gamma_0\Delta\tau,$$

$$1 = \int_{\tau-\Delta\tau}^{\tau} g(z)dz$$

При $u(\tau) = g(\tau)$ для значения $\xi = 1$. Введем в рассмотрение значение средней скорости за промежуток времени $\Delta\tau(\tau)$:

DOI 10.15588/1607-3274-2022-3-16

$$u_m(\tau)=\frac{1}{\Delta\tau(\tau)}.$$

При максимальной загрузки конвейерной линии

$$m(t)\to m_{\max}$$

Следует

$$\Delta\tau(\tau)=\Delta\tau_{\max},$$

$$\Delta\tau_{\max}=\frac{m_{\max}}{\gamma_0}=\frac{1}{u_{\max}}.$$

С уменьшением скорости ленты увеличивается масса горной породы, которая находится в конвейерной секции в процессе транспортировки. При выполнении условия (20) функцию Понтрягина запишем в виде

$$H=-u(\tau)\left(1+\frac{m_c}{\gamma_0\Delta\tau}\right)=-u_m(\tau)\left(1+u_m(\tau)\frac{m_c}{\gamma_0}\right)\to\max.$$

где $u_m(\tau)$ – среднее значение скорости ленты конвейера на промежутке $\Delta\tau(\tau)$

$$H_m=-u_m(\tau)\left(1+u_m(\tau)\frac{m_c}{\gamma_0}\right)\to\max.$$

Функция Понтрягина H является убывающей функцией при увеличении $u_m(\tau)$, принимает минимальное значение при значении

$$u_m(\tau)=u_{\min}.$$

Последнее условие соответствует режиму управления конвейерной линией, которая максимально загружена горной породой.

4 ЭКСПЕРИМЕНТЫ

Для проведения численных экспериментов разработано программное обеспечение, позволяющее рассчитывать значения параметров потока транспортного конвейера в соответствии с PiKh-моделью (7)–(8). Используя численную реализацию для модели секции конвейера, исследуем особенности применения релейного управления скоростью ленты $u(\tau)=(u_1,u_2)$ для режима функционирования транспортной системы, определяемого критерием качества (19). Минимальные затраты на перемещение материала будут в случае скорости движения ленты

$$u_{\max}=\frac{\gamma_0}{m_{\max}}$$

с периодом релейного регулирования

$$\Delta\tau_{\max}=\frac{m_{\max}}{\gamma_0}.$$

Точку переключения при заданных значениях и u_2 в пределах интервала $\Delta\tau_{\max}$ с режима, характеризующегося скоростью u_1 на режим, характеризующегося скоростью u_2 определим из соотношения:

$$u_1\Delta\tau_1+u_2\Delta\tau_2=1,$$
$$\Delta\tau_1+\Delta\tau_2=\Delta\tau_{\max},$$

откуда

$$\Delta\tau_1=\frac{1-u_2\Delta\tau_{\max}}{u_1-u_2}.$$

Точка переключения делит интервал $\Delta\tau_{\max}$ на два интервала $\Delta\tau_1$ и $\Delta\tau_2$, в пределах которых конвейерная линия функционирует со скоростью u_1 и u_2 соответственно. Величина интервала $\Delta\tau_{\max}$ определяется из условия максимальной загрузки конвейерной системы.

Для транспортной системы при величине поступающего потока материала

$$\gamma(\tau)=\gamma_0=0{,}5$$

и коэффициенте загрузки транспортной системы

$$m_{\max}=0{,}5$$

в случае релейного режима управления скоростью конвейерной ленты $u(\tau)=(0{,}5;1{,}5)$ получаем значение точки переключения $\Delta\tau_1=0{,}5$ на интервале (рис. 1)

$$\Delta\tau_{\max}=\frac{m_{\max}}{\gamma_0}=\frac{0{,}5}{0{,}5}=1.$$

Данный закон управления скоростью конвейерной ленты определяет выходной поток $\theta_1(\tau,1)$ (рис. 2) с конвейерной линии и линейную плотность материала $\theta_0(\tau,0)$ на входе конвейерной линии (рис. 3). В течение переходного периода $\tau\in[0,\tau_k=1{,}0]$ выходной поток $\theta_1(\tau,1)$ зависит от погонной плотности материала, распределенного вдоль транспортного маршрута

$$\theta_1(\tau,1)=H\left(1-\int_0^\tau u(z)dz\right)\psi\left(1-\int_0^\tau u(z)dz\right)u(\tau).$$

DOI 10.15588/1607-3274-2022-3-16

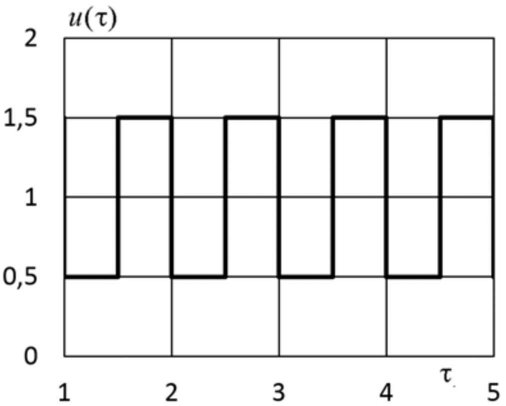


Рисунок 1 – Релейный режим управления скоростью конвейерной ленты $u(\tau)=(0{,}5;1{,}5)$

Начальное распределение материала при вычислительном эксперименте задано функцией

$$\psi(\xi)=\frac{1}{6}+\frac{1}{6}\sin\left(\pi\xi+\frac{\pi}{4}\right).$$

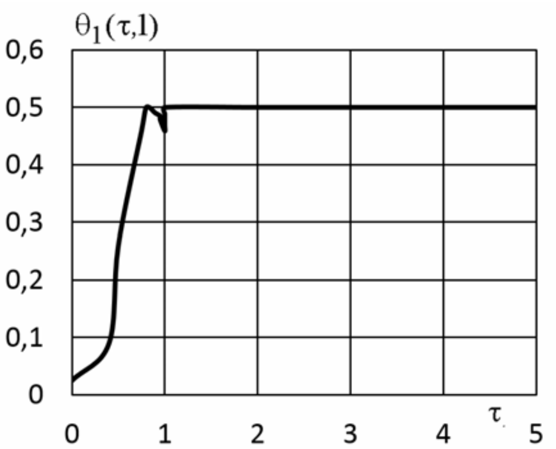


Рисунок 2 – Выходной поток с конвейерной линии $\theta_1(\tau,1)$ при $\gamma_0=0{,}5$, $u(\tau)=(0{,}5;1{,}5)$

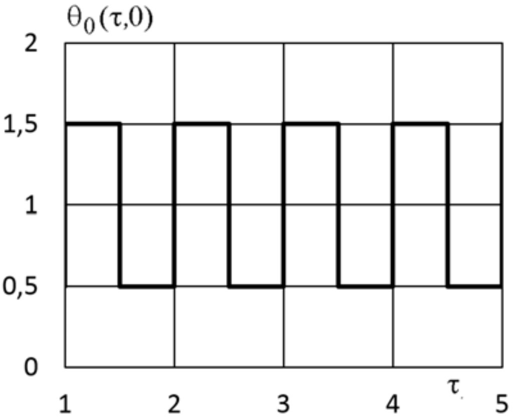


Рисунок 3 – Погонная плотность материала на входе конвейерной линии $\theta_0(\tau,0)$ при $\gamma_0=0{,}5$, $u(\tau)=(0{,}5;1{,}5)$

При $\tau\geq\tau_k=1{,}0$ величина выходного потока материала $\theta_1(\tau,1)$ рассчитывается из соотношения (18) и при заданных данных имеет значение $\theta_1(\tau,1)=0{,}5$ Постоянное значение выходного потока $\theta_1(\tau,1)$ для установившегося режима имеет простое объяснение. Алгоритм оптимального управления, полученный из заданного критерия качества при отсутствие ограничения, связанного с превышением максимального значения погонной нагрузки материала на ленту, обеспечивает постоянное значение материала в транспортной системе. Соответственно, при установившемся режиме величина выходного потока будет соответствовать величине входного потока $\theta_1(\tau,1)=\gamma_0=0{,}5$. Погонная плотность материала на входе конвейерной линии представлена на рис. 3.

5 РЕЗУЛЬТАТЫ

Для критерия оптимальности (19) получен в аналитическом виде алгоритм управление скоростью ленты, который подтверждает предположения об оптимальном управлении потоковыми параметрами секции конвейера, выдвинутые в работах [7, 14, 24, 29, 30]: при прочих равных условиях максимальное наполнение конвейерной ленты горной породой соответствует минимальному значению затрат энергии на перемещение материала. При максимально допустимой загрузке конвейерной линии в течение промежутка $\Delta\tau_{\max}$ энергопотребление является постоянным для режима скоростей $u(\tau)=(u_1,u_2)$, у которых одно и тоже среднее значение скорости движения ленты за период $\Delta\tau_{\max}$.

Таким образом, при максимальной загрузки конвейера потребление энергии за период $\Delta\tau_{\max}$ не зависит от плотности распределения горной породы вдоль маршрута транспортировки. Минимальное потребление энергии обеспечивается для режима постоянной скорости ленты конвейера $u_m(\tau)=u_{\min}$. Закон распределения линейной плотности горной породы вдоль конвейерной линии определяется заданными значениями режимов скоростей. Изменение закона управления приводит к изменению закона распределения линейной плотности материала вдоль маршрута транспортировки. Многоступенчатое управление скоростью ленты представляет интерес для режимов запуска и остановки конвейера, когда коэффициент загрузки секции конвейера отличается от максимального значения.

При построении алгоритма оптимального управления скоростью ленты принято допущение постоянного значения входного потока материала (20). В то же время для шахтных конвейеров характерны возмущения на входе конвейера, которые выражаются в дискретном характере грузопотоков и их переменной интенсивности в период непрерывных поступлений. Это является предметом дальнейших исследований. Принятое в работе упрощение нацелено на то, чтобы

DOI 10.15588/1607-3274-2022-3-16

показать перспективность использования PDE-моделей транспортных систем при проектировании высокоэффективных систем управления. Дополнительно в работе предполагалось, что поступающий грузопоток является измеряемой и контролируемой величиной. В случае, когда невозможно или затруднительно оценить грузопоток, контролируемой величиной является статическое усилие на валу двигателя.

В заключении остановимся оценке продолжительности переходного периода, в течение которого выходной поток материала с секции конвейера определяется начальным распределением материала вдоль маршрута транспортировки. Для расчета скорости конвейерной ленты и выходного потока материала транспортной системы на протяжении переходного периода необходимо использовать модели транспортного конвейера, основанные на прогнозировании значений потоковых параметров. Это приводит к ошибкам в расчете режимов управления скоростью конвейерной линии и к работе системы в ненормативного режиме, а в случае значительных отклонений– к перегрузке приводных электродвигателей и остановке конвейера. В связи с этим, факт наличия переходного периода, а также величина продолжительности переходного периода имеет важное значение при синтезе алгоритмов оптимального управления потоковыми параметрами. Оценим среднюю продолжительность переходного периода для действующих транспортных систем конвейерного типа [1]:

а) Belt conveyor with gearless drive Solution Prosper Haniel Coal Mine (Germany,2010) общая длина 3800 m, средняя скорость движения ленты 5,5 m/sec, средняя продолжительность переходного периода

$$\Delta\tau_1 = \frac{3800}{5,5} = 690 \text{ sec};$$

б) Conveyor line for Neyveli Lignite Corporation (India, 2007) общая длина 14000 m, средняя нормативная скорость движения ленты 5.4 m/sec, средняя продолжительность переходного периода

$$\Delta\tau_1 = \frac{14000}{5,4} = 2590 \text{ sec}.$$

Средняя продолжительность переходного периода составляет от нескольких минут до часа. При использовании систем управления скоростью ленты с диапазоном колебания $[0 \div 5]$ m/sec продолжительность переходного периода достигает несколько часов, что отражается на эффективности работы конвейерной линии. В течение переходного периода предложенный в настоящей работе алгоритм управления не обеспечивает оптимальное управление секцией конвейера для рассмотренного критерия качества управления.

ВЫВОДЫ

Рассмотрена актуальная задача синтеза алгоритмов оптимального управления потоковыми параметрами транспортной системы конвейерного типа со ступенчатым переключением режимов скорости движения ленты секции конвейера.

Научная новизна полученных результатов заключается в том, что для транспортных систем конвейерного типа с двухступенчатым переключением скорости ленты предложена аналитическая модель секции конвейера, которая позволяет рассчитать линейную плотность материала и поток материала вдоль маршрута транспортировки. Аналитическая модель секции конвейера является фундаментом для проектирования высокоэффективных систем управления потоковыми параметрами транспортной системы. Для заданного критерия качества управления транспортной системой предложен метод синтеза оптимального управления параметрами потока транспортной системы со ступенчатым переключением скорости ленты.

Практическая значимость полученных результатов заключается в разработке методики построения алгоритмов оптимального управления параметрами потока транспортной системы со ступенчатым регулированием величины потоковых параметров.

Перспективами дальнейших исследований является разработка систем управления скоростью ленты секции конвейера при стохастическом динамическом входном потоке материала.

БЛАГОДАРНОСТИ

Работа выполнена в рамках госбюджетной научно-исследовательской темы Национального технического университета «Харьковский политехнический институт» «Развитие теории и методов синтеза систем управления квазистатическими технологическими процессами» (ID:0108U001452).

ЛІТЕРАТУРА / ЛИТЕРАТУРА

1. SIMINE for conveyors. Siemens. [Electronic resource]. Access mode:https://new.siemens.com/global/en/markets/mining-industry.html
2. Дегтярев В. В. Нормирование топливно-энергетических ресурсов и регулирование режимов энергопотребления / В. В. Дегтярев. – М. : Недра, 1983. – 244 с.
3. Шахмейстер Л. Г. Динамика грузопотоков и регулирование скорости ленточных конвейеров / Л. Г. Шахмейстер, В. Г. Дмитриев, А. К. Лобачева. – М. : Недра, 1972. – 173 с.
4. Шахмейстер Л. Г. Теория и расчет ленточных конвейеров / Л. Г. Шахмейстер, В. Г. Дмитриев. – М. : Недра, 1978. – 392 с.
5. Шахмейстер Л. Г. Вероятностные методы расчета транспортирующих машин / Л. Г. Шахмейстер. – М. : Машиностроение, 1983. – 312 с.
6. Кариман С. А. Моделирование и оптимизация производственных процессов при добыче угля / С. А. Кариман, А. В. Брайцев, В. М. Шрамко. – М. : Наука, 1983. 135 с.

DOI 10.15588/1607-3274-2022-3-16

7. The impact of an uneven loading of a belt conveyor on the loading of drive motors and energy consumption in transportation / [A. Semenchenko, M. Stadnik, P. Belitsky et al.] // Eastern-European Journal of Enterprise Technologies. – 2016. – №82. – P. 42–51. https://doi.org/10.15587/1729-4061.2016.75936
8. Монастырский В. Ф. Эффективность работы ленточных конвейеров на горных предприятиях / В. Ф. Монастырский, В. Ф. Максютенко, Р. В. Кирия // Геотехническая механика. –2010. – № 88. – С. 185–191.
9. Разумный Ю.Т. Повышение энергоэффективности конвейерного транспорта угольных шахт / Ю. Т. Разумный, А. В. Рухлов, А. В. Козарь // Горная электромеханика и автоматика. – 2006. – № 76. – С. 24–28.
10. Проценко С. Н. Снижение энергопотребления на конвейерном транспорте угольных шахт / С. Н. Проценко // Горная электромеханика и автоматика. – 2008. – № 81. – С. 31–40.
11. Прокуда В. Н. Исследование и оценка грузопотоков на магистральном конвейерном транспорте ПСП «Шахта «Павлоградская» ПАО «Павлоградуголь» / В. Н. Прокуда, Ю. А. Мишанский, С. Н. Проценко // Горная электромеханика. – 2012. – № 88. – С. 107–111.
12. Ставицкий В.Н. Математическое моделирование транспортной системы добычного участка / В. Н. Ставицкий, А. С. Оголобченко // Научные труды ДНТУ. – 2013. – № 1. – С. 53–63.
13. Reutov A. Simulation of load traffic and steeped speed control of conveyor / A. Reutov // IOP Conference Series: Earth and Environmental. – 2017. – № 87. – P.1–6.
14. Кондрахин В. П. Статистический анализ эксплуатационных параметров шахтного ленточного конвейера / В. П. Кондрахин, Н. И. Стадник, П. В. Белицкий // Научные труды ДНТУ. – 2013. – №2. –С. 140–150.
15. Ададуров В. В. Теоретические основы и расчеты транспорта энергоемких производств / В. В. Ададуров, В.В. Ариненков, В. А. Будишевский // ДНТУ. – 1999. – С. 216. ISBN 966–7559–02–5.
16. Ставицкий В. Н. Динамика нагрузки регулируемого привода ленточного конвейера / В. Н. Ставицкий // Научные труды ДНТУ. – 2012. – № 23. – С. 49–53.
17. Ставицкий В. Н. Алгоритм идентификации транспортной задержки конвейера / В. Н. Ставицкий // Научные труды ДНТУ. – 2011. – № 20. – С. 59–66.
18. Ребенков Е. С. Синтез нечеткого регулятора частотно-управляемого электропривода конвейера с упругими связями / Е. С. Ребенков // Горный информационно-аналитический бюллетень. – 2016. – № 4. – С. 68–75.
19. Затонский А.В. Динамическая пространственная модель многоприводного ленточного конвейера / А. В. Затонский // Вестник АГТУ. – 2017. – № 4. – С. 99–110.
20. Кожубаев Ю. Н. Моделирование поточно-пространственной системы ленточных конвейеров / Ю. Н. Кожубаев, О. В. Прокофьев, В. И. Филимонов // Научно-технические ведомости СГПУ. – 2010. – №3. – С. 73–78.
21. Дьяченко В. П. Методы описания величины случайного грузопотока ленточных конвейерных предприятий на основе ее эмпирических распределений / В.П. Дьяченко // Горный информационно-аналитический бюллетень. – 2007. – № 3. – С. 287–289.
22. Маркарян Л. В. Анализ и моделирование случайного шахтного грузопотока на магистральном сборочном конвейере / Л. В. Маркарян // Горный информационно-аналитический бюллетень. – 2016. – № 5. – С. 67–74.
23. Заика В. Т. Влияние регулируемого привода на грузопотоки и эффективность системы шахтного конвейерного транспорта / В. Т. Заика, Ю. Т. Разумный, В. Н. Прокуда // Научный вестник НГУ. – 2015. – № 3. – С. 82–88.
24. Lauhoff H. Speed Control on Belt Conveyors–Does it Really Save Energy? / H. Lauhoff // Bulk Solids Handling. – 2005. – № 25(6). – P. 368–377.
25. Пигнастый О. М. Инженерно-производственная функция предприятия с серийным или массовым выпуском продукции / О. М. Пигнастый // Вопросы проектирования и производства конструкций летательных аппаратов. – 2005. – № 42(3). – С. 111–117.
26. Пигнастый О. М. Использование методов статистической физики для исследования экономико-производственных систем с массовым выпуском продукции / О. М. Пигнастый, В. П. Демуцкий, М. Н. Азаренкова // Вісник Харківського національного університету. – 2005. – № 710(2). – С. 128–134.
27. Pihnastyi O. M. Model of a composite magistral conveyor line / O. M. Pihnastyi, V. D. Khodusov // International Conference on System analysis & Intelligent computing (SAIC 2018): proceedings. – Kyiv, Ukraine : IEEE, 2018. – P. 68–72. https://doi.org/10.1109/saic.2018.8516739
28. Pihnastyi O. M. Statistical theory of control systems of the flow production. – LAP LAMBERT Academic Publishing, 2018. – 436 p. ISBN: 978-613-9-95512-1.
29. DIN 22101:2002-08. Continous conveyors. Belt conveyors for loose bulk materials. Basics for calculation and dimensioning: Standart. – [Effective from 2002]. – Normenausschuss Bergbau (FABERG) im DIN Deutsches Institut für Normung e. V. Normenausschuss Maschinenbau im DIN, 2022. – P. 51.
30. Pihnastyi O. M. Optimal control problem for a conveyor-type production line / O. M. Pihnastyi, V. D. Khodusov // Cybern. Syst. Anal. [Springer Science+Business Media, LLC, 1060-0396/18/5405-0744]. – 2018. – № 54(5). – P. 744–753.

Received 22.05.2022.
Accepted 30.07.2022.

УДК 658.51.012

СИНТЕЗ АЛГОРИТМУ УПРАВЛІННЯ ТРАНСПОРТНИМ КОНВЕЄРОМ

Пігнастий О. М. – д-р техн. наук, професор, професор кафедри розподілених інформаційних систем і хмарних технологій Національного технічного університету «Харківський політехнічний інститут», м Харків, Україна.

Івановська О. В. – канд. техн. наук, доцент, доцент кафедри композиційних конструкцій і авіаційного матеріалознавства Національного аерокосмічного університету ім. М. Є. Жуковського «Харківський авіаційний інститут», м. Харків, Україна.

DOI 10.15588/1607-3274-2022-3-16

АНОТАЦІЯ

Актуальність. Розглянута проблема оптимального управління потоковими параметрами транспортної системи конвейєрного типу при наявності ступенчатого регулювання швидкості ленти секцій конвейєра. Об'єктом дослідження була аналітична модель транспортного конвейєра, яка використана в якості фундаменту для синтезу алгоритмів оптимального управління потоковими параметрами транспортного конвейєра. Мета роботи полягає в розвитку методів проектування системи оптимального управління потоковими параметрами транспортного конвейєра з урахуванням транспортної задержки при ступенчаті регулювання потокових параметрів транспортної системи.

Метод. Розроблено аналітичну модель секції конвеєра з урахуванням ступінчастого регулювання значень потокових параметрів транспортної системи. При побудові моделі секції конвеєра визначення залежностей між потоковими параметрами транспортної системи записані рівняння в приватних похідних. Для синтезу алгоритмів оптимального управління швидкістю руху стрічки секції конвеєра запроваджено критерій якості управління. Використовуючи принцип максимуму Понтрягіна, поставлено завдання оптимального керування потоковими параметрами секції конвеєра. Для транспортної системи записано функцію Гамільтона, що враховує критерій якості управління, обмеження та диференціальні зв'язки між параметрами системи. Продемонстровано методику синтезу алгоритму оптимального управління швидкістю стрічки секції конвеєра. Визначено умови перемикання швидкості руху стрічки конвеєра.

Результати. Розроблена модель секції конвеєра використана синтезу алгоритму оптимального управління потоковими параметрами транспортної системи при ступінчастому перемиканні режимів швидкості стрічки.

Висновки. Розроблено методику синтезу алгоритмів оптимального керування потоковими параметрами транспортної системи при ступінчастому регулюванні швидкості руху стрічки секції конвеєра. Отримані алгоритми можуть бути використані для зниження питомих енергетичних витрат на транспортування матеріалу на підприємствах гірничодобувної промисловості.

КЛЮЧОВІ СЛОВА: конвеєр, розподілена система, PDE-модель, поточна лінія, виробнича лінія, регулювання швидкості стрічки.

UDC 658.51.012

SYNTHESIS OF AN ALGORITHM FOR CONTROL OF A TRANSPORT CONVEYOR

Pihnastyi O. M. – Dr. Sc., Professor of the Department of distributed information systems and cloud technologies, National Technical University "Kharkov Polytechnic Institute", Kharkiv, Ukraine.

Ivanovska O. V. – PhD, Assistant Professor of the Department of Composite Structures and Aviation Materials, National Aerospace University "Kharkiv Aviation Institute", Kharkiv, Ukraine.

ABSTRACT

Context. The problem of optimal control of the flow parameters of a transport system of a conveyor type in the presence of stepwise regulation of the speed of the conveyor section belt is considered. The object of the study is the analytical model of the transport conveyor, which was used as a foundation for the synthesis of optimal control algorithms for the flow parameters of the transport conveyor. The purpose of the work is to develop methods for designing systems for optimal control of the flow parameters of a transport conveyor, taking into account the transport delay with stepwise regulation of the flow parameters of the transport system.

Method. An analytical model of the conveyor section has been developed, taking into account the stepwise regulation of the values of the flow parameters of the transport system. When building a model of a conveyor section to determine the dependencies between the flow parameters of the transport system, equations in partial derivatives are written. For the synthesis of algorithms for optimal control of the speed of the conveyor section belt, a control quality criterion is introduced. Using the Pontryagin maximum principle, the problem of optimal control of the flow parameters of the conveyor section is posed. For the transport system, the Hamilton function is written, which takes into account the criterion of control quality, imposed restrictions and differential relationships between the system parameters. A technique for synthesizing an algorithm for optimal control of the speed of a conveyor section belt is demonstrated. The conditions for switching the speed of the conveyor belt are determined.

Results. The developed model of the conveyor section is used to synthesize an algorithm for optimal control of the flow parameters of the transport system with stepwise switching of belt speed modes.

Conclusions. A technique for synthesizing algorithms for optimal control of the flow parameters of a transport system with stepwise regulation of the speed of the belt of a conveyor section has been developed. The obtained algorithms can be used to reduce the specific energy costs for material transportation at mining enterprises.

KEYWORDS: conveyor, distributed system, PDE model, production line, production line, belt speed control.

REFERENCES

1. SIMINE for conveyors. Siemens, 2017. https://new.siemens.com/global/en/markets/mining-industry.html
2. Degtyarev V.V. Normirovanie toplivno-energeticheskih resursov i regulirovanie rezhimov energopotrebleniya. Moscow, Nedra, 1983, 244 p.
3. Shahmeyster L. G., Dmitriev V. G., Lobachyova A.K. Dinamika gruzopotokov i regulirovanie skorosti lentochnyih konveyerov. Moscow, Nedra, 1972, 173 p.
4. Shahmeyster L.G. Teoriya i raschet lentochnyih konveyerov / L.G. Shahmeyster, V.G. Dmitriev. – M.: Nedra, 1978. – 392 s.
5. Shahmeyster L.G. Veroyatnostnyie metodyi rascheta transportiruyuschih mashin. Moscow, Mashinostroenie, 1983, 312 p.
6. Kariman S. A., Braytsev A. V., Shramko V. M. Modelirovanie i optimizatsiya proizvod-stvennyih protsessov pri dobyiche uglya. Moscow, Nauka, 1983, 135 p.
7. Semenchenko A., Stadnik M., Belitsky P. et al. The impact of an uneven loading of a belt conveyor on the loading of drive

DOI 10.15588/1607-3274-2022-3-16

motors and energy consumption in transportation, *Eastern-European Journal of Enterprise Technologies*, 2016, No. 82, pp. 42–51. https://doi.org/10.15587/1729-4061.2016.75936

8. Monastyirskiy V. F., Maksyutenko V. F., Kiriya R.V. Effektivnost rabotyi lentochnyih konveyerov na gornyih predpriyatiyah, *Geotehnicheskaya mehanika,* 2010, No. 88, pp. 185–191.
9. Razumnyiy Yu. T., Ruhlov A. V., Kozar A. V. Povyishenie energoeffektivnosti konvey-ernogo transporta ugolnyih shaht, *Gornaya elektromehanika i avtomatika*, 2006, No. 76, pp. 24–28.
10. Protsenko S.N. Snizhenie energopotrebleniya na konveyernom transporte ugolnyih shaht, *Gornaya elektromehanika i avtomatika,* 2008, No. 81, pp. 31–40.
11. Prokuda V. N., Mishanskiy Yu. A., Protsenko S.N. Issledovanie i otsenka gruzopotokov na magistralnom konveyernom transporte PSP «Shahta «Pavlogradskaya» PAO «Pavlogradugol», *Gornaya elektromehanika*, 2012, No. 88, pp. 107–111.
12. Stavitskiy V.N., Ogolobchenko A.S. Matematicheskoe modelirovanie trans-portnoy sistemyi dobyichnogo uchastka, *Nauchnyie trudyi DNTU,* 2013, No. 1, pp. 53–63.
13. Reutov A. Simulation of load traffic and steeped speed control of conveyor, *IOP Conference Series: Earth and Environmental*, 2017, No. 87, pp.1–6.
14. Kondrahin V. P. Stadnik N.I. , Belitskiy P.V. Statisticheskiy analiz ekspluatatsionnyih parametrov shahtnogo lentochnogo konveyera, *Nauchnyie trudyi DNTU*, 2013, No. 2, pp. 140–150.
15. Adadurov V. V., Arinenkov V. V., Budishevskiy V. A. Teoreticheskie osnovyi i raschetyi trans-porta energoemkih proizvodstv, *DNTU,* 1999, P. 216. ISBN 966–7559–02–5.
16. Stavitskiy V.N. Dinamika nagruzki reguliruemogo privoda lentochnogo konveyera, *Nauchnyie trudyi DNTU*, 2012, No. 23, pp. 49–53.
17. Stavitskiy V. N. Algoritm identifikatsii transportnoy zaderzhki konveyera, *Nauchnyie trudyi DNTU*, 2011, No. 20, pp. 59–66.
18. Rebenkov E.S. Sintez nechetkogo regulyatora chastotno-upravlyaemogo elektroprivoda konveyera s uprugimi svyazyami, *Gornyiy informatsionno-analiticheskiy byulleten*, 2016, No. 4, pp. 68–75.
19. Zatonskiy A.V. Dinamicheskaya prostranstvennaya model mnogoprivodnogo lentochnogo konveyera, *Vestnik AGTU*, 2017, No. 4, pp. 99–110.
20. Kozhubaev Yu. N., Prokofev O.V., Filimonov V.I. Modelirovanie potochno-prostranstvennoy sistemyi lentochnyih konveyerov, *Nauchno-tehnicheskie vedomosti SGPU,* 2010, No. 3, pp. 73–78.
21. Dyachenko V.P. Metodyi opisaniya velichinyi sluchaynogo gruzopotoka lentochnyih konveyernyih predpriyatiy na os-nove ee empiricheskih raspredeleniy, *Gornyiy informatsionno-analiticheskiy byulleten,* 2007, No. 3, pp. 287–289.
22. Markaryan L.V. Analiz i modelirovanie sluchaynogo shahtnogo gruzopotoka na magistralnom sborochnom konveyere, *Gornyiy informatsionno-analiticheskiy byulleten*, 2016, No. 5, pp. 67–74.
23. Zaika V. T., Razumnyiy Yu. T., Prokuda V. N. Vliyanie reguliruemogo privoda na gruzopotoki i effektivnost sistemyi shahtnogo konveyernogo transporta, *Nauchnyiy vestnik NGU*, 2015, No. 3, pp. 82–88.
24. Lauhoff H. Speed Control on Belt Conveyors–Does it Really Save Energy? *Bulk Solids Handling*, 2005, No. 25(6), pp. 368–377.
25. Pihnastyi O.M. Inzhenerno-proizvodstvennaya funktsiya predpriyatiya s seriynyim ili massovyim vyipuskom produktsii, *Voprosyi proektirovaniya i proizvodstva konstruktsiy letatelnyih apparatov*, 2005, No. 42(3), pp. 111–117.
26. Pihnastyi O. M., Demutskiy V. P., Azarenkova M. N. Ispolzovanie metodov statisticheskoy fiziki dlya issledovaniya ekonomiko-proizvodstvennyih sistem s massovyim vyipuskom produktsiitsii, *Visnik Kharkivskogo natsionalnogo universitetu,* 2005, No. 710(2), pp. 128–134.
27. Pihnastyi O. M., Khodusov V. D. Model of a composite magistral conveyor line, *International Conference on System analysis & Intelligent computing (SAIC 2018): proceedings*. Kyiv, Ukraine, IEEE, 2018, pp. 68–72. https://doi.org/10.1109/saic.2018.8516739
28. Pihnastyi O.M. Statistical theory of control systems of the flow production. LAP LAMBERT Academic Publishing, 2018, 436 p. ISBN: 978-613-9-95512-1,
29. DIN 22101:2002-08. Continous conveyors. Belt conveyors for loose bulk materials. Basics for calculation and dimensioning: Standart. – [Effective from 2002]. Normenausschuss Bergbau (FABERG) im DIN Deutsches Institut für Normung e. V. Normenausschuss Maschinenbau im DIN, 2022. – p 51.
30. Pihnastyi O. M., Khodusov V.D. Optimal control problem for a conveyor-type production line, *Cybern. Syst. Anal. [Springer Science+Business Media, LLC, 1060-0396/18/5405-0744],* 2018, No. 54(5), pp. 744–753.

DOI 10.15588/1607-3274-2022-3-16

UDC 004:007

BEHAVIOR CLASSIFICATION OF CONTROL UNIT OF SYSTEMS

Poliakov M. O. – Dr. Sc., Associate professor, Professor of the Electrical and Electronic Apparatuses Department, National University "Zaporizhzhia Polytechnic", Zaporizhzhia, Ukraine.

ABSTRACT

Context. The behavior of the system is included in the basic concepts that characterize its functioning. In an event-driven system, behavior is modeled using a state machine. Known classifications of behavior take into account the genus and type of automaton. At the same time, in modern systems, control automata are integrated into hierarchies and have a number of new properties that are not reflected in their classifications.

Objective. The purpose of the work is to systematize the forms of specifying the behavior of integrated systems and methods for changing the behavior in the process of their use. The novelty of the proposed classification lies in taking into account the behavior of new types of non-binary, semantic, controlled and changeable individual automata and the structures of these automata.

Method. The essence of behavior is presented as the ambiguity of reactions to the input signals of the control automaton, which manifests itself in a certain pattern of changing its states and outputs. When classifying behaviors, the expediency of exploratory behavior is determined. Such ways of achieving the goal as adaptation, change or absorption of the environment, change in the goals of behavior are noted. According to the level of complexity of behavior, systems with predetermined, regulated, organizing, predictable and autonomous behavior are distinguished. Along with the automaton model of behavior, the importance of modeling behavior in the form of a combination of statements is noted. The importance of describing the possible and emergency behavior of the system is noted. A classification of the system's behavior in terms of constancy and variability is proposed. The structure and principles of the implementation of changeable behavior within the framework of the processes of external control of the automaton and its self-government are described. Based on the concept of arity of behavior, the functional and technological behavior of a finite automaton are singled out. As part of the classification of behavior by the level of formation, the switching, combinational and automatic behavior of states, as well as the behavior of the automaton in the contours of activity and the typical behavior of the automaton in the hierarchy, are described.

Experiments. With the use of the proposed classification features, the behavior of control devices of monitoring systems for power transformer parameters, object temperature control and integrated hierarchical systems is analyzed.

Results. The proposed classification describes the directions for specifying behavior in complex integrated systems according to 13 main and 84 detailing features, which facilitates the process of designing behavior and highlights new system capabilities.

Conclusions. The actual problem of systematization of the behavior of control devices of systems has been solved. Classification features give directions for the use of standard solutions for describing the behavior of the system, which simplifies the process and reduces the complexity of designing its functional structure.

KEYWORDS: system behavior, control automaton, hierarchy of automata, integrated system, behavior classification.

ABBREVIATIONS

A is actuators;
CA is control automata;
CO is the control object;
CU is control unit;
IOA is input operational automata;
MOA is the intermediate operational automata;
OA is operational automata;
OOA is output operating automata;
S is sensors.

NOMENCLATURE

$C_{from\ i}$ – is control inputs from the i-th level;
$C_{from\ (i+1)}$ – is control inputs from (i+1)-th level;
$C_{to\ (i-1)}$ is control outputs to the (i–1)-th level;
H_{ij} is the probability characterizing the change of state with the number i to the state with the number j;
Q_{ij} is the probabilities characterizing the appearance of the output with the number j, if the current state of the automaton has the number i;
S is the set of automaton states;
s_0 is the initial state of the automaton;
X is the set of automaton inputs;
X_1 is IOA inputs (sensor outputs);
Y is the set of automaton outputs;
Y_1 is OOA outputs (actuator inputs);
Δ is an array of automaton outputs;
$\Delta[i]$ is the i-th output in the array Δ;
λ is the transition function;
μ is the function of the automaton outputs;
Ψ is an array of automaton transitions;
$\Psi\ [i, j]$ is the transition from i to j in the array Ψ.

INTRODUCTION

The tasks of cognition of a person and an artificial, for example, a technical system qualitatively coincides: based on the information received, understand and predict their actions, actions and behavior. At the same time, human behavior is studied as a means to satisfy his needs, achieve a goal, adapt to the external environment, as a system of actions or a manifestation of a two-link "stimulus-response" scheme [1]. The same motives for studying behavior are applicable to the processes of studying the behavior of technical systems. The concept of the behavior of a technical system is interpreted by researchers in different ways and develops along with the development of ideas about systems and the use of new types of behavior.

Behavior (British English: behaviour) is a set of actions and manners performed by individuals, organisms, systems or artificial objects in some environment. This is the calcu-

DOI 10.15588/1607-3274-2022-3-17

lated response of a system or organism to various stimuli or influences, internal or external [2]. From the point of view of behavioral informatics, behavior consists of an actor, an operation, interactions and their properties [3].

As noted in [4], behavior is one of the levels of description of technical systems during their design. According to [5], behavior is included in the basic concepts that characterize the functioning and development of the system. It is associated with the ability of the system to move from one state to another. These transitions are connected by cause-and-effect relationships, and in the process of functioning, one of the possible options for transitions is selected. The logic of choice characterizes the behavior of the system. More details about the behavior of the system can be judged after determining the process model in which the behavior is used: continuous, event-driven or hybrid [6, 7].

The object of study is the process of designing the behavior of event-driven systems.

The subject of study is the classification of the behavior of such systems. Classifications of the behavior of control automata of control systems are known according to the type and type of automaton [8, 9]. At the same time, complex technical systems, as a rule, are integrated, hierarchical, with various forms of implementation and relationships between the behavior of their control automata, with a description of the system behavior not only with the help of automata, but also in the form of knowledge [10–15]. The paper [16] noted the need to determine the combination and integration of system behaviors. In [17], the behavior of a system is defined as a way of interaction in a hierarchy of environments and agents immersed in these environments. At the same time, it is noted that the model of interaction between the control and operational automata should be considered as the ideological prototype of insertion models of such interaction [18, 19].

In the known literature, there is no classification of the forms for specifying the behavior of systems for technical purposes, as well as ensuring that this behavior changes during the operation of the system. As a consequence, this leads to an incomplete use of the possibilities of adapting the behavior of the system and a decrease in the effectiveness of its application.

The purpose of the work is to systematize the forms of specifying the behavior of integrated and cognitive systems and methods for changing behavior in the process of using a technical system**.**

1 PROBLEM STATEMENT

The generalized model of the technical control system [8] is shown in fig. 1. It includes a control object, which is connected to the control device with the help of sensors and actuators. In turn, CU is subdivided into operational and control automata, and OA – into input, intermediate and output operational automata. The system with the structure of Fig. 1 can have continuous, event-driven, and hybrid behavior. Continuous behavior is carried out in the circuit: CO – S – IOA – MOA – OOA – A – CO. In this circuit, operating automata perform the functions of a process controller in the control object. The regulation law determines the continuous behavior of the system. Event-driven behavior is performed in the circuit: CO – S – IOA – CA – OOA – A – CO. The IOA automata in this circuit perform the functions of converters of signals from sensors into the inputs of the control automaton, and the OOA automata act as converters of the SA outputs into actuator control signals.

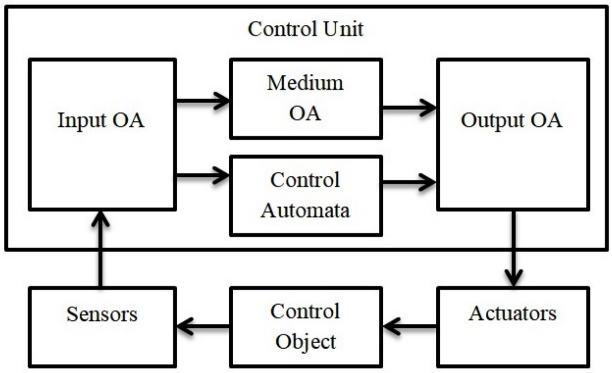


Figure 1 – Structure of the control system

With hybrid behavior, several continuous and event-based control loops can operate in the system, which affect each other [11]. In the classical version of the hybrid system, in each state of the control automaton in the event-driven circuit, the corresponding circuit of continuous behavior is activated.

The classical CA model is a finite automaton [8], which is described by a tuple:

$$\langle X, Y, S, s_0, \mu, \lambda \rangle. \tag{1}$$

At the same time, the behavior of the technical system is given by the functions of outputs μ and transitions λ and characterizes the possible ambiguity of the dependence of the state of outputs *Y* CA on its inputs *X*.

To illustrate this situation, in [9], an example of the behavior of a control automaton with the input “Your feet are stepped on in transport” and different values of the output “Your actions” at the first and fifth such event are given.

The described structure of the CA of the “classical” control system can be implemented in hardware (in the form of a logical node with memory elements) [20] or in software (in the form of a program that runs in the operating environment of the system’s computing device) [21]. Changing the behavior of such a system occurs by replacing this circuit/program with new ones that represent new functions μ and λ. In this case, it may be necessary to change the sets *X*, *Y*, *S* and the nodes that form them (operating machines, sensors and actuators).

The task of the study is to analyze the known methods for constructing control systems, to identify and classify ways to specify behavior in the system.

DOI 10.15588/1607-3274-2022-3-17

2 REVIEW OF THE LITERATURE

In [22, 23], a variety of approaches to the study of behavior is noted, but it is noted that they are united by the representation of behavior as a process within which the system interacts with the environment. That is, this behavior is a response formed by the system to signals from the environment.

In [22], system behaviors are divided into normative (ritual, imitative, and role-playing) and situational (analytical, play, and entertaining) types. But such a classification does not reflect the form of implementation of behavior using finite automata.

In the dictionary [24] there is an article "Automaton with a variable structure", which emphasizes the importance of this area of research on control automata. At the same time, she gives a link to the article "Stochastic automaton", that is, an automaton in which, instead of transition and output functions, in the general case, probability distributions of a discrete type are specified. For transitions, the probabilities H_{ij} are given, and for the output, the probabilities Q_{ij}. Depending on the success or failure of the actions of the stochastic automaton, H_{ij} and Q_{ij} are recalculated, which leads to a change in the structure and behavior of the control automaton.

In [23, 25, 26], the collective behavior of control automata and multi-agent systems are considered, in which the behavior is formed by changing their impact on the external environment as a result of the reaction of the environment or other automata/agents to this impact. This topic requires a separate consideration, therefore, in this paper, the behavior of only one automaton is considered.

In [27], the issues of behavior reconfiguration in discrete-event systems are considered, the concept of a controlled discrete-event system is introduced, in which input events can be disabled by an external controller-supervisor. Such a shutdown makes it possible to change, within certain limits, the behavior of the control machine. At the same time, we note that switching off the automaton inputs does not exhaust all the possibilities of supervisory control.

It can be seen from the literature review that control automata are being studied from various angles, but there is no study and classification of automaton behavior of hierarchical integrated and cognitive systems.

3 MATERIALS AND METHODS

Automaton behavior in the system is proposed to be classified according to the following features:

– goal;
– a way to achieve the goals of behavior;
– the level of difficulty;
– type of model;
– type of behavior in relation to the mode of operation of the system;
– stability;
– arity;
– the level of formation of behavior;
– type of management in the hierarchy;
– function;
– the nature of the processes.

These signs, in turn, are subdivided into signs of the second rank, and so on. Each feature is assigned an alphanumeric code that reflects its place in the classification. An example of an alphanumeric feature code is shown in Fig. 2.

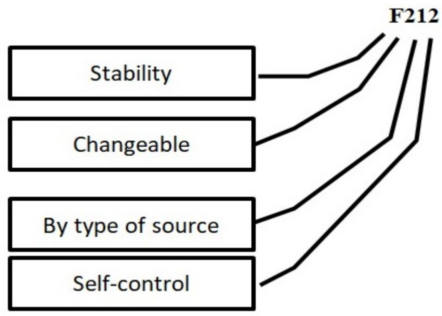


Figure 2 – An example of an alphanumeric feature code

The structure of the signs of behavior is shown in Fig. 3.

We will distinguish between the basic and research goals of behavior. The basic goal (A1) of behavior is effective management based on existing knowledge. But, with the development of knowledge-based systems, the goal of research (A2) of the object and / or control device, the environment was added to it to improve the efficiency of control behavior in the future [28–30].

Behavioral goals can be achieved through parametric (B11) or structural (B12) adaptation [31], transition to a more favorable environment (B2), inclusion of a part of the environment into the system (B3) or change in behavioral goals (B4).

By the level of complexity, we will distinguish (in ascending order of complexity) systems with predetermined (C1), regulated (C2), organizing (C3), predictive (C4) and autonomous behavior (C5).

Predefined behavior is the simplest behavior with fixed functions of operating and control automata of the system control device. Regulated behavior provides for the possibility of changing the parameters of operational automata leading to a change in the conditions for the formation of an event at their output.

Organizing behavior changes the functions of outputs and/or transitions of the control automaton. That is, with this behavior, it is allowed to change all elements of the control automaton tuple.

Predictive behavior assumes the presence in the control device of a system of operating automata in which the model of the object and the environment is executed. As a result of the flow of predicted parameters arriving at the inputs of these automata, the object model enters a certain state, which is taken into account when planning future behaviors.

DOI 10.15588/1607-3274-2022-3-17

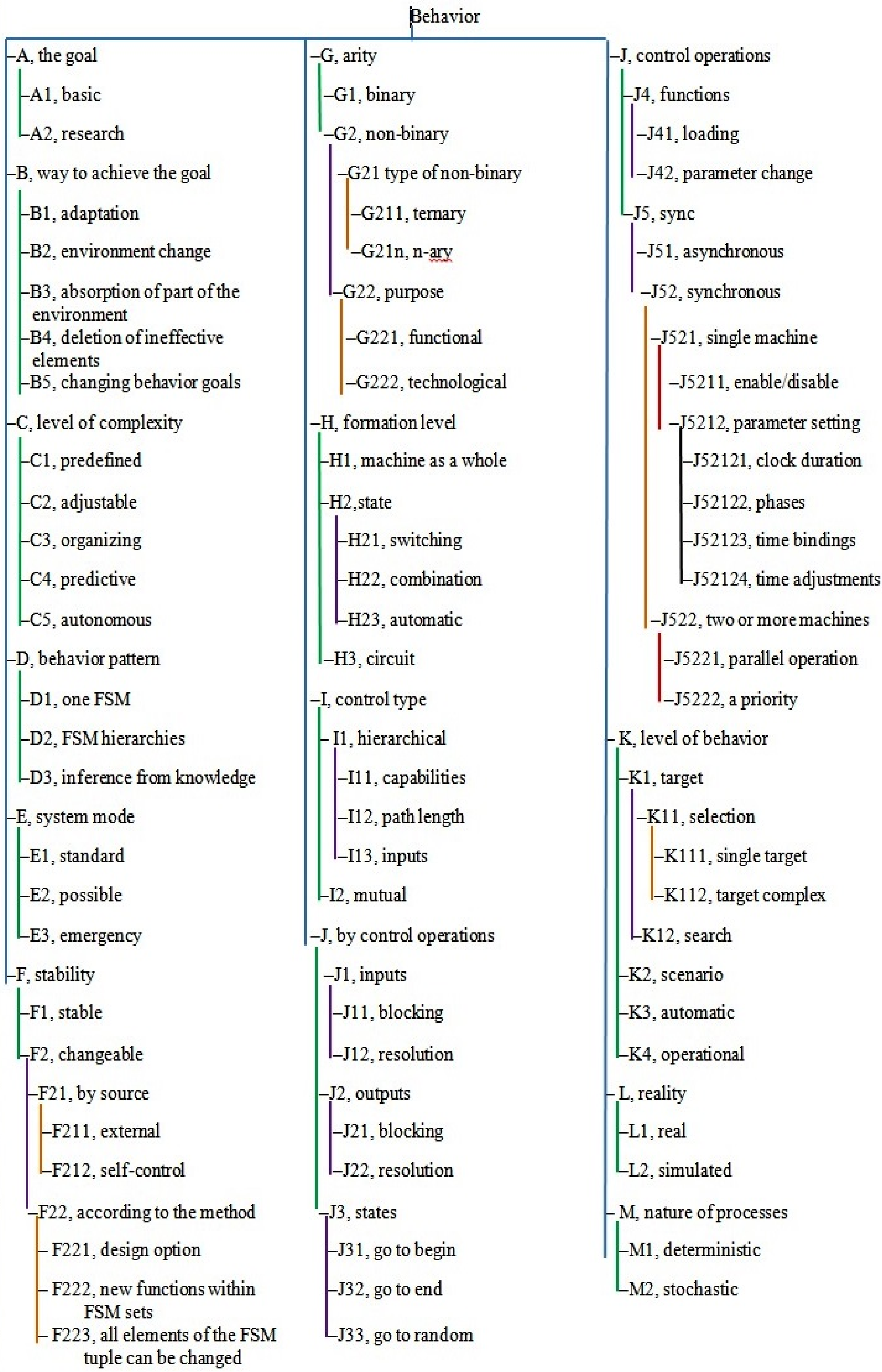


Figure 3 – The structure of classification features

DOI 10.15588/1607-3274-2022-3-17

Autonomous behavior implies, at a minimum, the participation of the system in the choice of a set of goals for its functioning. In an extended version of autonomous behavior, the system itself performs a search and formulates the goals of its functioning and builds its behavior under them.

By the type of behavior model, we will distinguish between setting behavior using the functions of one (D1) or some hierarchy (D2) control automata and behavior based on logical inference (D3) from knowledge stored in the system base [14]. The advantage of a behavior model based on logical inference is the ability to share knowledge that describes the normal (normal) behavior of the system and knowledge that does not fit into this behavior.

The characteristic of behavior regarding the mode of operation (E) of the system includes the following gradations: regular (E1), possible (E2) and emergency (E3). Regular behavior corresponds to the goals of the current level of development of the system and knowledge about it. Deviations from regular behavior can be caused by a change in the properties of the system object or control device (its development or degradation), knowledge about them and their influence on the process of achieving the goal of the system functioning. At least some of these deviations are recognized as possible, rechecked and become part of regular behavior. Emergency behavior is a response to unacceptable system states. In complex systems, emergency behavior takes on many gradations and is not reduced to a trivial "turn everything off" action.

The classification of behavior in the categories of stability (constancy) (behavior F1) – variability (behavior F2) allows, on the one hand, to characterize the predicted results of the system's functioning under conditions of constant external influences, and on the other hand, the degree of flexibility of the system's behavior when these conditions change.

Changeable behavior (F2) is classified according to the source (initiator) of changes (F21) and the way they are carried out (F22). The control device in a system with variable control contains elements of the implementation of one or another variant of the type of control and control inputs, with the help of which the current control variant is selected. Changes in the behavior of the system or its subsystem may be initiated by an external control device (F211) and/or be the result of self-management actions (F212).

Changing behavior can be done in the following ways:

1. Activation of a behavioral variant from a certain set of transition functions, outputs of the automaton and the initial state s_0 of the automaton, laid down in the design of the system (F221).

2. Synthesis of new behavior within the existing sets of inputs, outputs, states (F222) or by changing these sets (F223).

The next classification feature G is the arity of behavior. The behavior of a finite automaton can be classified based on the arity of the elements of its sets. Arity determines the number of values that an element of each of the sets of the automaton can take – inputs, outputs, states [12]. We will distinguish between binary (G1) and nonbinary (G2) automata. And by the type of nonbinarity – ternary (G211) and, in the general case, *n*-ary (G21n). Classical finite automata have binary elements of sets. So the binary element of the state set has two meanings "active" and "passive".

In the presence of non-binary sets of the automaton, we will distinguish between the behavior of G221 for the implementation of the target function of the system and the technological behavior of G222 of the control automaton [32]. In the first case, the outputs of the control automaton depend on the purpose of operation and the properties of the system object. In the second, it depends on the features of the implementation of this behavior, such as the controllability of the structure, the mechanisms for transferring activities from one state to another, the possibility of generating outputs depending on the value of the state, and others.

The behavior of the automaton can be classified according to the level of formation (behavior H):

– The behavior of the H1 automaton as a whole. Such behavior, as already noted, is specified by the output and transition functions.

– The behavior of H2 at the state level. The behavior of the automaton is specified through the set of behaviors in its states [34]. The behavior of an active state describes the logical (causal) relationships of state inputs to its outputs and actions. In [35], a variation of the behavior of H2 is described, which is specified on a certain subset of states of the automaton. In this case, different behaviors are implemented in other subsets. Such an automaton is called multi-behavioral.

– The behavior at the level of logical connections of the state with the control objects (behavior H3). In [34], activity contours are proposed that describe the function of the control object, that is, the processes resulting from the impact on the object from the control device and the reactions of the object, which are fixed by the control device.

The type of control behavior in the hierarchy of control automata is also a classification feature of behavior I. Subtypes of this behavior are hierarchical (I1) and mutual (I2) behavior. There is an extensive class of hierarchical systems in which their subsystems form an integrated system [10]. The interoperability of subsystems located at neighboring levels is ensured by the behavior of typical interconnection elements that consist of a controlled and controlling automaton or are covered by a mutual control loop [13].

We classify the operations of controlling the behavior of J according to the elements of the controlled automaton that they affect: control of inputs X (behavior of J1); control of outputs Y (behavior of J2); state control S (behavior J3); control of functions μ, λ (J4 behavior) and synchronization control (J4 behavior). Details of these behaviors are provided in the next section.

In an integrated multilevel system, different levels implement different behavior [10, 13, 15, 34]. Therefore, belonging to the functional level of the system is a classi-

DOI 10.15588/1607-3274-2022-3-17

fication feature K, according to which we will distinguish between the behavior of the target (K1), scenario (K2), automatic (K3) and operational (K4) levels.

The result of the target behavior is the choice of the current goal of functioning and the activation of the scenario behavior automaton, which selects the current scenario for achieving the goal. This script selects and activates a variant of automatic behavior, and so on.

A feature of cognitive systems is the use of operations for predicting the results of decisions made on the choice of goals, scenarios, and so on. In the course of forecasting, the behavior of the object and the control device is modeled in a certain flow of events. Thus, the behavior in the system can be not only real (L1), but also simulated (L2).

According to the nature of the processes occurring in the system, deterministic (M1) and stochastic (M2) behavior is distinguished [36, 37]. With the behavior of M1, the results of repeated repetition of any admissible sequence of values at the inputs of the automaton must be the same. At the same time, with the behavior of M2 and the same conditions at the inputs, the results may differ in the trajectory of the change of states and the outputs in them.

Thus, in the behavior of the control automata of the systems, classification features are distinguished for choosing the types of behavior in the system during the design process.

EXPERIMENTS

The task of the experimental part of the work is to identify behavioral properties in known technical systems that could be classified according to the proposed features. If a certain type of behavior is missing in a known system, then the principles of its implementation and the advantages of using it are described. The author has not been able to find a technical system in which it is advisable to use all the described behaviors, because the systems in the examples given have some subsets of the proposed behaviors. Examples of systems for experiments with the classification of behaviors are control systems for traffic lights, elevators, conveyors, machine tools, robots, power transformers, vehicles, and technological processes. The variety of types of behavior increases significantly for "smart" varieties of such systems.

Such systems are characterized by the presence of a cognition subsystem with knowledge form converters, whose work expands the knowledge base for management, and an activity subsystem in the form of a hierarchy of control automata that use this knowledge to improve management efficiency.

Behavior A1 is basic and is implemented with different efficiency in all event-driven systems. An example of the behavior of A2 is given in [28]. Its essence lies in the activation of the cooling modes of the power transformer, which are not used in the current operating conditions. The information obtained as a result of such behavior on the technical condition of the elements of these systems makes it possible to increase the cooling efficiency.

The reason for launching adaptive mechanisms (behavior B1) may be a change in the object of the system, environmental parameters, or the technical state of system elements [31, 34]. Thus, a reaction to a change in the mass of an object of the system can be an increase in the waiting time for its heating. Forced cooling is used to thermally stabilize an object under conditions of an increase in ambient temperature. If individual elements fail, the load is redistributed to serviceable elements or redundant ones are connected.

An example of the actions of a transport robot aimed at achieving the goal of functioning in the optimal temperature regime can be the search for a location with protection from solar radiation or with a large wind blowing, that is, a change in the environment (behavior B2).

An example of achieving goals by including part of the environment in the system (behavior B3) is the power transformer control system. If the increase in the temperature of the transformer elements due to the increase in load could not be compensated by cooling, then the load can be limited. For this purpose, the object load control circuit is included in the control system.

A transformer control system can also serve as an example of a system with dynamically changing behavior goals (behavior B4). So the original goal of "maximizing service life" under certain conditions can be changed to the goal of "provide a sustainable energy supply, ignoring the accelerated wear of equipment".

An example of C1 behavior predetermined at the design stage is the operation of a control unit based on a classical finite state machine [8, 9]. For example, the event "Object overheating" occurs at the same temperature of the object. The processing of this event in the control machine always leads to the transition of the machine to the "Emergency" state and the action "Turn off the heater". If the outputs of the control machine control the parameters of the operating machines (for example, "Superheat temperature") of the control unit, then the behavior of C2 takes place. The behavior of C3 implies a change in the structure of the operating and/or control automata of the control unit. Predictive Behavior C4 uses simulation results to improve management efficiency. For example, predicting system load and ambient temperature can optimize the performance of a facility's cooling/heating system.

Autonomous behavior of C5 is characteristic of biological systems, starting with the simplest bacteria [38]. The principles of implementing the autonomy of these systems are used, for example, in the construction of artificial agents for the search control of robots [29].

The behavior model D1 given at the set-theoretic level by a tuple (1) has two varieties: Mealy and Moore automata [8, 9]. These automata differ in the way they bind outputs. For Moore automata, the current output value depends only on the current state. And for Mealy automata, the value of the output depends on the input that caused the transition to this state. An example of D1 behavior is the behavior of digital nodes with memory, traffic light control systems, and others. A variation of the

DOI 10.15588/1607-3274-2022-3-17

behavior of D1 is the behavior of recognizing the input sequence [8].

The D1 model is included in more complex behavioral formalisms, such as the state diagram of the UML language [39], the Harel formalism implemented in the Stateflow software of the MathLab system [40], the programming language SFC (Sequential Function Chart) [41] and others.

The behavior of D2 involves the selection of sub automata and the modeling of the processes of their interaction. Possible variants of interaction include embedding a sub automaton in the state of a super automaton and integrating sub automata. The nesting of the automaton in the state by goals is similar to the use of subroutines in programming – it allows you to increase visibility, reduce the dimension of the model. The integration of automata is used in the construction of hierarchies, in which the control element at the *i*-th level is the control object at the (*i* + 1)-th level of the hierarchy, which ensures the interoperability of control processes [13]. For example, it is the processes of choosing the goal of the functioning of the system and the strategy for achieving it.

The behavior specified by the state machine functions can be described in the form of a combination of statements (D3 behavior) and processed by means of logical inference. For example, the statement "If the current state is S_i and the signal X_k has arrived, then go to the state S_j" corresponds to an arc in the automaton graph between these vertices.

We will consider the behaviors of E using the example of a simple heating control system. The behavior of E1 is to turn on the heater if the temperature of the object is less than θ_1 and turn it off if this temperature is greater than θ_2. The behavior of E2 takes into account the possibility of a delay in heating due to changes in the characteristics of the object. Such a delay is not considered as a sign of an accident and is investigated further. An example of the behavior of E3 in such a system is the introduction of additional states for processing events recognized as emergency. For example, it is the state of "Emergency" in which the emergency shutdown of the furnace is performed if the temperature of the object exceeded θ_3.

Variable behavior of automata F2 is the main and promising type of behavior of systems in the category F.

The structure of external control F211 and self-control F212 at the *i*-th level of the system is shown in Fig. 4 [34].

The CA machine has four groups of outputs:

– *Y* group is connected to OOA inputs;

– group $C_{from\ i}$ is connected with the control inputs of the OA, through which the parametric adaptation is performed;

– group $C_{to\ (i-1)}$ controls the structural adaptation of SA at the (*i*–1)-th level;

– group *X* is connected to additional information inputs of the CA, the impact on which can bring the CA to additional states, thereby changing the structure of the CA by self-control (behavior F212).

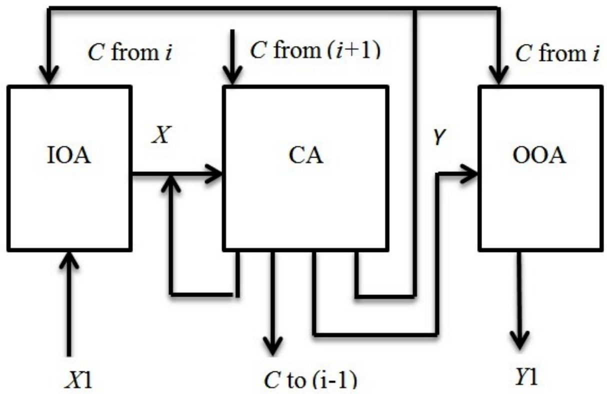


Figure 4 – Structure of external control and self-control at the *i*-th level of the system

The behavior of F221 for the automaton of the *i*-th level (Fig. 4) is set by the code at the inputs $C_{from\ (i+1)}$, which activates one of the variants of the automaton specified during the design. The same kind of behavior is possible to change the IOA and OOA functions using $C_{from\ i}$ inputs,

If in the process of the system functioning, the functions of the control automaton are performed by a universal program, and the specifics of a particular automaton are described by data arrays in matrix form, then a change in the structure of the control automaton (behavior F222 – F224) is reduced to changes in its arrays of transitions Ψ; and outputs Δ.

The main operations of changing the structure of such a control automaton are:

– adding / removing (*i*, *j*)-th transition (ADDIN / DELIN Ψ [*i*, *j*]) in the array of transitions Ψ;

– adding / deleting the *i*-th output (ADDOUT / DELOUT Δ [*i*]) in the array of outputs Δ;

– adding / removing state (ADDS [*n* + 1] / DELS [*n* – 1]) in the arrays Ψ and Δ, which boils down to increasing / decreasing the size of these arrays;

– assignment of a new initial state s_0 of the control automaton.

In [32], the behavior with features G211 (ternary) and G221 (technological) was used to reduce the dimension of the control automaton graph. At the same time, the essence of technological behavior is that the activity of the outputs is manifested in a certain neighborhood of the active state.

In the classical automaton with tuple (1), the behavior H1 takes place, which is specified by the functions of the automaton. The work [12] describes an automaton in which the behavior is specified by a set of behaviors in its states. We will consider such behavior as a variant of H1 behavior with the possibility of implementing various variants of H2 behavior in different states.

The classical behavior of H22 is combinational and lies in the fact that the action (output) in the state does not depend on which input caused the state to be activated. The output will be the same for any variant of entering the

DOI 10.15588/1607-3274-2022-3-17

state. The work [33] describes commutation, combinational, and automaton behaviors. The switching behavior of H21 defines a hard coupling when the state is activated: each state input corresponds to one of its outputs. With the combinational behavior of H22, the state of the output is determined by a logic function from the inputs that are associated with this output. For the classic state behavior mentioned above, it is the binary logical OR function. In other cases, it is possible to use other functions and implement the behavior corresponding to them. With the automaton behavior of H23, the state of the *i*-th level of the system is described as a state machine of another, usually lower (i–1)-th level, which is initialized at the moment when the state of the *i*-th level becomes active.

In [14], the behavior of H3 is implemented as a chain of logical statements that form the contour of activity. This behavior was used to diagnose the technical condition of the elements of the heating control system of an object, such as actuators, sensors and operating machines, which made it possible to expand the possibilities of adapting the system.

When studying the behavior of I1 using a simulator [42], the following typical control algorithms were identified: control with a sequential increase/decrease in functionality (behavior I11), increase/decrease in the length of the path in the control cycle (behavior I12), decrease/increase in the number of automaton inputs used (behavior of I13). The combination of these options determines the typical behavior of the control at that level of the hierarchy.

We classify the behavior control operations J according to the elements of the controlled automaton that they affect:

The behavior J1 of control of inputs X is blocking J11 / allowing J12 all or part of the inputs from the set X. In this case, blocking all inputs will lead to a stop, "freezing" the automaton in a certain state, and blocking some of the inputs will lead to a change in the functions μ and λ.

The behavior J2 of controlling outputs Y is blocking J21 / allowing J22 all or part of the outputs from the set Y, outputs of a certain state from the set S. In the case of blocking all outputs, it should be taken into account that the state of the control object can change under the influence of external factors.

Behavior J3 of state control S is the reset of the automaton to the initial state J31, the transition to the final state s_0 J32 or to an arbitrary given intermediate state J33.

The behavior J4 of the control of the functions μ, λ is the loading of the new function J41; changing the parameters of the J42 function by blocking / allowing all or part of the transitions defined by the mapping $\lambda{:}S\times X\rightarrow S$ or outputs defined by the mappings $\mu{:}S\times X\rightarrow Y$, $\mu{:}S\times X\rightarrow Y$ in some "maximum" mapping. The behavior of J42, unlike the behavior of J41, requires that the maximum mapping be set beforehand. As a result of loading a new function, it is possible to change the type of automaton, that is, the replacement of a Mealy automaton by a Moore automaton and vice versa.

The J5 behaviors are J51 asynchronous and J52 asynchronous synchronization behaviors. For synchronous automata, this behavior has features that depend on the number of automata.

The timing control behavior of a single synchronous machine includes enabling/disabling the timing input J5211, setting the clock duration J52121 and the phases of the timing signals J52122, how they are related to the system time J52123, and the time correction J52124. In this case, blocking the synchronization signals is equivalent to blocking all inputs of the automaton. Synchronization control of the execution of several automata is the task of the higher control automaton, which is reduced to the implementation of the behavior of allowing / disabling parallel operation of controlled automata at the current time J5221, changing the priority level of the execution of automata J5222 [34].

We will choose the system of continuous monitoring of the technical condition of a power transformer [43] as the object of classifying the behavior of the control device on the basis of K. At the target level of behavior, this system is represented by the target selection behaviors K411 and K412. The selected goal or set of goals is associated with the active state of the goal machine. When this state is activated, the output events of the input operational automata and the knowledge base of the target system level are used.

Examples of goals for the behavior of the K4 system include extending the life of the transformer, ensuring the required load current, regardless of the accelerated wear of the transformer, and others. An example of the scenario behavior of K3 for the purpose of extending the service life is the scenario of increasing the cooling regime.

The reality of the system's behavior, in accordance with the classification feature L, consists in the use of physically real input information and influences on the system object. This behavior is basic. In cases where the physically real information is incomplete and/or the impact on the system object is destructive, the behavior of L2 modeling of the object, the environment and the system control device is used. Simulation results are used both in real time and for predicting the parameters and state of the system. For example, the simulation results of ambient temperature and load current are used to feed-forward control the cooling of a power transformer.

With deterministic behavior, transitions from one state of the automaton to another on the initiative of the controlling automaton itself and at random times are unacceptable. Another type is non-deterministic behavior, in which the probabilistic nature of the formation of outputs in an automatic state takes place.

As noted in [44], in systems with M2 behavior, nondeterminism appears due to various reasons, such as the level of abstraction, system simplification in modeling, partial controllability/observability of the input/output ports of the system.

DOI 10.15588/1607-3274-2022-3-17

RESULTS

During the experiments, the behavior of the control devices of the systems was classified according to 13 main features, which were detailed according to 40 features of the second, 23 of the third, 13 of the fourth, four of the fifth and four of the sixth levels.

Comparison of the classification of the behavior of the control devices of systems with known classifications is shown in the petal diagram in Fig. 5.

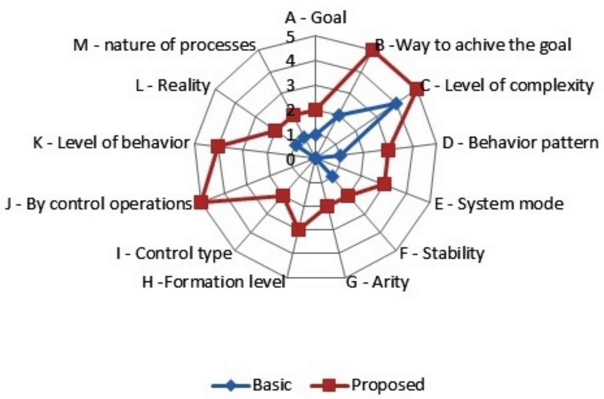


Figure 5 – Comparison of the number of classification features in the directions of the basic and proposed classifications

As can be seen from the diagram in Fig. 5, the number of classification features in all areas of the proposed classification is increased by 1–5 features compared to the base one. A number of new features have been introduced, such as H, I, J, K.

DISCUSSION

Experiments on examples have shown the practical value of the proposed classification for describing the variety of reactions of complex systems.

The classification makes it possible to single out new possibilities in the behavior of systems, such as the behavior of mutual control in the loops of continuous and event control; exploratory behavior, system expansion behavior and continuous revision of current system goals; setting a system behavior model as a result of logical inference, taking into account possible and emergency behaviors; external control or self-management of the control machine of the system; application of behavior with increased arity; formation of switching, combinational and automaton behaviors at the level of a separate state of the automaton; inclusion in the composition of the behavior of knowledge presented in the contours of the automaton activity and the interaction of control automata in a hierarchical integrated system.

The use of new behaviors increases the adaptive properties of systems, reduces the dimension of their behavior model, and increases the knowledge base that is used to select behavior. The carried out structuring of classification features simplifies the process of designing the structure of the behavior of the control device of the system.

The proposed classification is not final and will be corrected and supplemented as systems develop, including those with cognitive behavior.

CONCLUSIONS

The actual problem of systematization of the behavior of control automata of control systems is solved.

The scientific novelty of the results obtained lies in the fact that the classification of the behavior of control automata of control systems has been further developed, which takes into account the hierarchical integrated structures of the control devices of systems, the behavior of new types of non-binary, semantic, controlled and changeable individual automata and the structures of these automata.

The practical significance of the results obtained lies in the fact that they allow us to form standard solutions and, as a result of their use, simplify the process, reduce the complexity of designing the functional structure of the system.

Prospects for further research are to study combinations of technical solutions for the construction of control devices of systems proposed on the basis of their classification for the construction of a wide class of systems.

ACKNOWLEDGMENTS

The work was supported by the state budget research work of the National University “Zaporozhzhia Polytechnic” “Intelligent information technologies for data processing” (state registration number 0118U100063).

REFERENCES

1. Koverzneva I. A. avt.-sost. Psikhologiya aktivnosti i povedeniya: ucheb.-metod. kompleks. Minsk, izd-vo MIU, 2010. 316 p.
2. Minton E. A., Lynn R. Belief Systems, Religion, and Behavioral Economics. New York, Business Expert Press LLC, 2014, 160 p. ISBN 978-1-60649-704-3
3. Cao L. In-depth behavior understanding and use: The behavior informatics approach, *Information Sciences,* 2010, Vol. 180, Issue 17, pp. 3067–3085. https://doi.org/10.1016/j.ins.2010.03.025
4. Gero J., Kannengiesser U., Chakrabarti A., Blessing L.T.M. Eds. The Function-Behaviour-Structure Ontology of Design, *An Anthology of Theories and Models of Design.* Berlin, Springer, 2014, pp. 1–45.https://link.springer.com/chapter/10.1007/978-1-4020-5131-9_21
5. Volkova V. N., Denisov A. A. Teoriya sistem i sistemnyy analiz : uchebnik dlya akademicheskogo bakalavrata, 2-ye izd. Moscow, Yurayt, 2014, 616 p.
6. Shornikov Yu., Bessonov A., Dostovalov D. Specification and instrumental analysis of hybrid systems, *Science Bulletin of the Novosibirsk State Technical University*, 2015, No. 4(61), pp. 101–117 (in Russian). http://dx.doi.org/10.17212/1814-1196-2015-4-101-117
7. Maryasin O., Kolodkina A. S., Ogarkov A. A. Computer Modelling «Smart Building», *Modeling and Analysis of Information Systems,* 2016, Vol. 23(4), pp. 427–439. https://doi.org/10.18255/1818-1015-2016-4-427-439
8. Glushkov, V. M. Sintez tsifrovykh avtomatov. Moscow, Fizmatizdat, 1962, 476 p.

DOI 10.15588/1607-3274-2022-3-17

9. Karpov YU. G. Teoriya avtomatov. Sankt Peterburg, Piter, 2002, 224 p.
10. Poliakov M. A. Teoretiko-mnozhestvennyye modeli elementov i struktur integrirovannykh kontrollernykh sistem, *Systemni tekhnolohiyi : rehional'nyy mizhvuzivs'kyy zbirnyk naukovykh prats',* 2012, No. 2 (79), pp. 75–81.
11. Poliakov M. A., Andriyas I. A. Teoretiko-mnozhestvennyye modeli funktsionalʹnykh struktur gibridnykh avtomatov sistem upravleniya, *Systemni tekhnolohiyi : rehional'nyy mizhvuzivs'kyy zbirnyk naukovykh prats'*, 2018, No. 3 (116), pp. 146–152.
12. Poliakov M. A., Andrias I. A. Konechnyye avtomaty s nebinarnymi elementami mnozhestv, *Systemni tekhnolohiyi : rehional'nyy mizhvuzivs'kyy zbirnyk naukovykh prats'*, 2019, No. 2 (121), pp. 85–94.
13. Poliakov M. O., Subbotin S. O., Poliakov O. M. Interoperability of Integrated Hierarchial Systems / M. O. Poliakov, // *Systemni tekhnolohiyi : rehional'nyy mizhvuzivs'kyy zbirnyk naukovykh prats',* 2021, No. 2 (133), pp. 68–78. https://doi.org/10.34185/1562-9945-2-133-2021-08
14. Poliakov M. O., Subbotin S., Andrias I. // Control System Control Unit FSM Semantic Models, *Systemni tekhnolohiyi, rehional'nyy mizhvuzivs'kyy zbirnyk naukovykh prats'*, 2019, No. 5 (124), pp. 43 – 53. https://doi.org/10.34185/1562-9945-5-124-2019-05.
15. Poliakov M. Cognitive Control Systems: Structures and Models, *Electrotechnic and Computer Systems*, 2017, No. 101, pp. 387–393. https://dx.doi.org/ 10.15276/eltecs.25.101.2017.46.
16. Stetter R. Approaches for Modelling the Physical Behavior of Technical Systems on the Example of Wind Turbines, *Energies*, 2020, No. 13(8), pp. 2087. https://doi.org/10.3390/en13082087
17. Letichevskiy A. Ad. Insertsionnoye modelirovaniye, *Upravlyayushchiye sistemy i mashiny*, 2012, No. 6, pp. 3–14. http://dspace.nbuv.gov.ua/handle/123456789/83102
18. Letichevskiy, A. A. V. M. Glushkov and Modern Informatics, *Visnyk Natsional'noyi akademiyi nauk Ukrayiny*, 2013, No. 08, pp. 21–33. https://doi.org/10.15407/visn2013.08.021
19. Glushkov V. M. Editor(s): Academician J. G. Gvishiani. Mathematics and Cybernetics, *Science, Technology and Global Problems*. Pergamon, 1979, pp. 173–180. https://doi.org/10.1016/B978-0-08-024469-3.50029-2
20. Pedroni Volnei A. Finite State Machines in Harware: Theory and Design (with VHDL and SystemVerilog), Pedroni, The MIT Press, 2013. https://doi.org/10.7551/mitpress/9657.001.0001
21. Drumea A., Popescu C. Finite state machines and their applications in software for industrial control, *27th International Spring Seminar on Electronics Technology: Meeting the Challenges of Electronics Technology Progress,* 2004, Vol. 1, pp. 25–29. https://doi.org/10.1109/ISSE.2004.1490370
22. Gaaze-Rapoport, M. G., Pospelov D. A. Ot ameby do robota: modeli po-vedeniya. Moscow, Nauka. Gl. red. fiz.-mat. lit., 1987, 288 p. (Problemy nauki i tekhnicheskogo progressa).
23. Tsetlin M. L. Issledovaniya po teorii avtomatov i modelirovaniyu biologicheskikh sistem. Glavnaya redaktsiya fiziko-matematicheskoy literatury izd-va «Nauka», Moscow, 1969, 316 p.
24. Averkin A. N., Gaaze-Rapoport M. G., Pospelov D. A. Tolkovyy slovarʹ po iskusstvennomu intellektu. Moscow, Radio i svyazʹ, 1992, 256 p.
25. Varshavskiy, V. I. O povedenii stokhasticheskikh avtomatov s peremennoy strukturoy, *Avtomatika i telemekha-nika*, 1969, Vol. 24, No. 3, pp. 353–360.
26. Tîrnăucă C., José L. Behavioral Modeling Based on Probabilistic Finite Automata: An Empirical Study. Montaña, Santiago Ontañón et al. Sensors (Basel), 2016 Jul; No. 16(7), P. 958. Published online 2016 Jun 24. https://doi.org/10.3390/s16070958
27. Wonham W., Cai K. Supervisory Control of Discrete-Event Systems, *Communication and Control Engineering*, 2019, Springer, 487 p. https://doi.org/10/1007/978-3-319-77452-7
28. Poliakov M. A. Identifikatsiya teplovykh parametrov silovogo maslyanogo transformatora po dannym monitoringa parametrov, *Visnyk Skhidnoukrayins'koho natsional'noho universytetu*, 2007, No. 11, Ch. 1 (117), pp. 167–173.
29. Nepomnyashchikh, V. A. Modeli avtonomnogo poiskovogo povedeniya, *Ot modeley povedeniya k iskusstvennomu intellektu,* Kollektivnaya monografiya pod obshch. red. Redʹko V. G. Moscow, URSS, 2006, pp. 200–242.
30. Wittenmark B. (1995). Adaptive Dual Control Methods, An Overview. Lund University, 1995, June CiteSeerX 10.1.1.25.7446.
31. Rastrigin L. A. Adaptatsiya slozhnykh sistem. Riga, Zinatne, 1981, 375 p.
32. Poliakov M., Subbotin S., Poliakov O. Performance indicators of models of non-binary control automates, *Experience of Designing and Application of CAD Systems: IEEE 16th International Conference*. Lviv, 22–26 Feb. 2021, proceedings, pp. 38–42. http:doi.org/10.1109/CADSM52681.2021.9385220
33. Poliakov M., Subbotin S., Poliakov O. The contour of causality in control automata of systems [Electronic resource], *Proceedings of the Fourth International Workshop on Computer Modeling and Intelligent Systems (CMIS-2021).* Zaporizhzhia, April 27, 2021, ed.: S. Subbotin. Aachen: CEUR-WS, 2021, pp. 368–378. (CEUR-WS.org, vol. 2608). Access mode: http://ceur-ws.org/Vol-2864/paper32.pdf https://doi.org/10.32782/cmis/2864-32
34. Poliakov M., Subbotin S., Poliakov O. Set-theoretical FSM Models Activity Subsystem for Cognitive Control Systems, *Experience of Designing and Application of CAD Systems: IEEE 15th International Conference*. Polyana, 26 Feb. – 2 March. 2019, proceedings, pp. 1–4. https://doi.org/10.1109/CADSM.2019.8779283
35. Xuesen L. Multi-behaviors Finite State Machine, *Information, Computing and Telecommunication: IEEE Youth Conference on 2009: proceedings,* pp. 201–203. https://doi.org/10.1109/YCICT.2009.5382390
36. Gramlich G. Probabilistic and Nondeterministic Unary Automata, *Lecture Notes in Computer Science.* Springer, 2003, Vol. 2747, pp. 460–469. http://dx.doi.org/10.1007/978-3-540-45138-9_40
37. Campeanu C. Non-Deterministic Finite Cover Automata, *Scientic Annals of Computer Science,* 2015, Vol. 25 (1), pp. 3–28. http://doi.org/10.7561/SACS.2015.1.3
38. Anokhin P. K. Idei i fakty v razrabotke teorii funktsionalʹnykh sistem, *IV-ya Vserossiyskaya nauchno-tekhnicheskaya konferentsiya «Neyroinformatika-2002». Problemy intellektual'nogo upravleniya – obshchesistemnyye, evolyutsionnyye i neyrosetevyye aspekty. Materialy diskussii*. Moscow, Izd-vo Moskovskogo gosudarstvennogo inzhenerno-fizicheskogo instituta (MIFI), 2003, pp. 40–57.
39. Drusinsky D. Modeling and Verification Using UML Statecharts. A Working Guide to Reactive System Design, Runtime Monitoring and Execution-Based Model Checking.

DOI 10.15588/1607-3274-2022-3-17

Book Elsevier Inc., 2006, P. 309. https://doi.org/10.1016/B978-0-7506-7949-7.X5000-4
40. Colgren R. D. Basic MATLAB, Simulink and Stateflow [Electronic resource], AIAA, 2007. Access mode: https://arc.aiaa.org/doi/book/10.2514/4.861628
41. Kontrollery programmiruyemyye. CH. 3 Yazyki programmirovaniya GOST R MEK 61131–3–2016 (IEC 61131–3:2013, IDT). Moscow, Standartinform, 2016, 227 p.
42. FSM Simulator code. Available [Electronic resource]. – Access mode: https://gitlab.com/Eonus/fsm-simulator (Accessed Feb. 24, 2022).
43. Poliakov M. A., Andriyas I. A., Konogray S. P., Vasilevskiy V. V. Kognitivnoye upravleniye zhiznennym tsiklom izolyatsii obmotok maslonapolnennogo silovogo transformatora, *Visnyk Natsional'noho tekhnichnoho universytetu "KHPI". Seriya "Elektrychni mashyny ta elektromekhanichne peretvorennya enerhiyi"*, 2018, No. 5 (1281), pp. 90–96.
44. Kushik, N. G. Metody sinteza ustanovochnykh i razlichayushchikh eksperimentov s nedeterminirovannymi avtomatami. Avtoreferat dissertatsii na soiskaniye uchenoy stepeni kandidata fiziko-matematicheskikh nauk: 05.13.01 «Sistemnyy analiz, upravleniye i obrabotka informatsii (po otraslyam)». Tomsk, TGU, 2013, 20 p.

Received 25.06.2022.
Accepted 21.08.2022.

УДК 004:007

КЛАСИФІКАЦІЯ ПОВЕДІНКИ ПРИСТРОЇВ КЕРУВАННЯ СИСТЕМ

Поляков М. О. – д-р техн. наук, доцент, професор кафедри електричних та електронних апаратів Національного університету «Запорізька політехніка», м. Запоріжжя, Україна.

АНОТАЦІЯ

Актуальність. Поведінка системи входить у основні поняття, що характеризують її функціонування. У подієво-керованій системі поведінка моделюється за допомогою кінцевого автомата. Відомі класифікації поведінки враховують рід і тип автомата. Разом з тим, у сучасних системах керуючі автомати інтегровані в ієрархії та мають низку нових властивостей, які не відображені у їх класифікаціях.

Ціль. Метою роботи є систематизація форм завдання поведінки інтегрованих систем та методів зміни поведінки у процесі їх використання. Новизна запропонованої класифікації полягає в обліку поведінок нових типів небінарних, семантичних, керованих та змінюваних окремих автоматів та структур цих автоматів.

Метод. Сутність поведінки представлена як неоднозначність реакцій на вхідні сигнали керуючого автомата, яка проявляється у певній закономірності зміни його станів та виходів. При класифікації поведінок визначено доцільність дослідницької поведінки. Відзначено такі способи досягнення мети як адаптація, зміна або поглинання довкілля, зміна цілей поведінки. За рівнем складності поведінки виділені системи з визначеною, регульованою, організуючою, прогнозованою та автономною поведінкою. Поруч із автоматною моделлю поведінки відзначено важливість моделювання поведінки як комбінації висловлювань. Відзначено важливість опису можливої та аварійної поведінки системи. Запропоновано класифікацію поведінки системи в категоріях сталості та мінливості. Описано структуру та принципи реалізації мінливої поведінки в рамках процесів зовнішнього управління автоматом та його самоврядування. Спираючись на поняття арності поведінки, виділено функціональну та технологічну поведінку кінцевого автомата. У рамках класифікації поведінки за рівнем формування описані комутаційна, комбінаційна та автоматна поведінка станів, а також поведінка автомата в контурах діяльності та типова поведінка автомата в ієрархії.

Експерименти. Із застосуванням запропонованих класифікаційних ознак проаналізовано поведінку пристроїв керування систем моніторингу параметрів силового трансформатора, керування температурою об'єкта та інтегрованих ієрархічних систем.

Результати. Запропонована класифікація визначає напрями завдання поведінки у складних інтегрованих системах за 13 основними та 84 ознаками що деталізують, що полегшує процес проектування поведінки, виділяє нові можливості систем.

Висновки. Вирішено актуальне завдання систематизації поведінки пристроїв управління систем. Класифікаційні ознаки дають напрями використання типових рішень опису поведінки системи, що спрощує процес, зменшує трудомісткість проектування її функціональної структури.

КЛЮЧОВІ СЛОВА: поведінка системи, керуючий автомат, ієрархія автоматів, інтегрована система, класифікація поведінки.

УДК 004:007

КЛАССИФИКАЦИЯ ПОВЕДЕНИЙ УСТРОЙСТВ УПРАВЛЕНИЯ СИСТЕМ

Поляков М. О. – д-р техн. наук, доцент, профессор кафедры электрических и электронных аппаратов Национального университета «Запорожская политехника», г. Запорожье, Украина.

АННОТАЦИЯ

Актуальность. Поведение системы входит в основные понятия, характеризующие ее функционирование. В событийно-управляемой системе поведение моделируется с помощью конечного автомата. Известные классификации поведения учитывают род и тип автомата. Вместе с тем, в современных системах управляющие автоматы интегрированы в иерархии и имеют ряд новых свойств, которые не отражены в их классификациях.

DOI 10.15588/1607-3274-2022-3-17

Цель. Целью работы является систематизация форм задания поведения интегрированных систем и методов изменения поведения в процессе их использования. Новизна предложенной классификации заключается в учете поведений новых типов небинарных, семантических, управляемых и изменяемых отдельных автоматов и структур этих автоматов.

Метод. Сущность поведения представлена как неоднозначность реакций на входные сигналы управляющего автомата, которая проявляется в определенной закономерности смены его состояний и выходов. При классификации поведений определена целесообразность исследовательского поведения. Отмечены такие способы достижения цели как адаптация, изменение или поглощение окружающей среды, изменение целей поведения. По уровню сложности поведения выделены системы с предопределенным, регулируемым, организующим, прогнозируемым и автономным поведением. Наряду с автоматной моделью поведения отмечена важность моделирования поведения в виде комбинации высказываний. Отмечена важность описания возможного и аварийного поведений системы. Предложена классификация поведения системы в категориях постоянства и изменчивости. Описана структура и принципы реализации изменчивого поведения в рамках процессов внешнего управления автоматом и его самоуправления. Опираясь на понятие арности поведения, выделены функциональное и технологическое поведение конечного автомата. В рамках классификации поведения по уровню формирования описаны коммутационное, комбинационное и автоматное поведение состояний, а также поведение автомата в контурах деятельности и типовое поведение автомата в иерархии.

Эксперименты. С применением предложенных классификационных признаков проанализировано поведение устройств управления систем мониторинга параметров силового трансформатора, управления температурой объекта и интегрированных иерархических систем.

Результаты. Предложенная классификация описывает направления задания поведения в сложных интегрированных системах по 13 основным и 84 детализирующим признакам, что облегчает процесс проектирования поведения, выделяет новые возможности систем.

Выводы. Решена актуальная задача систематизации поведения устройств управления систем. Классификационные признаки дают направления использования типовых решений описания поведения системы, что упрощает процесс, уменьшает трудоемкость проектирования ее функциональной структуры.

КЛЮЧЕВЫЕ СЛОВА: поведение системы, управляющий автомат, иерархия автоматов, интегрированная система, классификация поведений.

ЛІТЕРАТУРА / ЛИТЕРАТУРА

1. Коверзнева И. А. Психология активности и поведения: учеб.-метод. комплекс / авт.-сост. И. А. Коверзнева. – Минск : изд-во МИУ, 2010. – 316 с.
2. Minton E. A. Belief Systems, Religion, and Behavioral Economics / Elizabeth A. Minton, Lynn R. – New York : Business Expert Press LLC, 2014. – 160 p. ISBN 978-1-60649-704-3
3. Cao L. In-depth behavior understanding and use: The behavior informatics approach / L. Cao // Information Sciences. – 2010. – Vol. 180, Issue 17. – P. 3067–3085. https://doi.org/10.1016/j.ins.2010.03.025
4. Gero J. The Function-Behaviour-Structure Ontology of Design / J. Gero, U. Kannengiesser ; Chakrabarti A., L.T.M. Blessing Eds // An Anthology of Theories and Models of Design. – Berlin : Springer. – 2014. – P. 1–45. https://link.springer.com/chapter/10.1007/978-1-4020-5131-9_21
5. Волкова В. Н. Теория систем и системный анализ : учебник для академического бакалаврата / В. Н. Волкова, А. А. Денисов. – 2-е изд. – М. : Юрайт, 2014. – 616 с.
6. Shornikov Yu. Specification and instrumental analysis of hybrid systems/ Yu, Shornikov, A. Bessonov, D. Dostovalov // Science Bulletin of the Novosibirsk State Technical University. – 2015. – No. 4(61). – P. 101–117 (in Russian). http://dx.doi.org/10.17212/1814-1196-2015-4-101-117
7. Maryasin O. Computer Modelling «Smart Building» / O. Maryasin, A. S. Kolodkina. A. A. Ogarkov // Modeling and Analysis of Information Systems. – 2016. – Vol. 23(4). P. 427–439. https://doi.org/10.18255/1818-1015-2016-4-427-439
8. Глушков В. М. Синтез цифровых автоматов / В. М. Глушков. – М. : Физматиздат, 1962. – 476 с.
9. Карпов Ю. Г. Теория автоматов / Ю. Г. Карпов. – Санкт Петербург : Питер, 2002. – 224 с.
10. Поляков М. А. Теоретико-множественные модели элементов и структур интегрированных контроллерных систем управления / М. А. Поляков // Системні технології : регіональний міжвузівський збірник наукових праць. – 2012. – № 2 (79). – С. 75–81.
11. Поляков М. А. Теоретико-множественные модели функциональных структур гибридных автоматов систем управления / М. А. Поляков, И. А. Андрияс // Системні технології : регіональний міжвузівський збірник наукових праць. – 2018. – № 3 (116). – С. 146–152.
12. Поляков М. А. Конечные автоматы с небинарными элементами множеств / М. А. Поляков, И. А. Андриас // Системні технології : регіональний міжвузівський збірник наукових праць. – 2019. – № 2 (121). – С. 85–94.
13. Poliakov M. O. Interoperability of Integrated Hierarchial Systems / M. O. Poliakov, S. O. Subbotin, O. M. Poliakov // Системні технології : регіональний міжвузівський збірник наукових праць. – 2021. – № 2 (133). – С. 68–78. https://doi.org/10.34185/1562-9945-2-133-2021-08
14. Poliakov M. O. Control System Control Unit FSM Semantic Models / M. Poliakov, S. Subbotin, I. Andrias// Системні технології:регіональний міжвузівський збірник наукових праць. – 2019. – №5 (124). – С. 43–53. https://doi.org/10.34185/1562-9945-5-124-2019-05.
15. Poliakov M. Cognitive Control Systems: Structures and Models / M. Poliakov // Electrotechnic and Computer Systems. – 2017. – № 101. – P. 387–393. https://dx.doi.org/10.15276/eltecs.25.101.2017.46.
16. Stetter R. Approaches for Modelling the Physical Behavior of Technical Systems on the Example of Wind Turbines / R. Stepper // Energies. – 2020. – №13(8). – P. 2087. https://doi.org/10.3390/en13082087
17. Летичевский А. Ад. Инсерционное моделирование / А. Ад. Летичевский // Управляющие системы и машины. – 2012. – № 6. – С. 3–14. http://dspace.nbuv.gov.ua/handle/123456789/83102
18. Letichevskiy A. A. V. M. Glushkov and Modern Informatics / A. A. Letichevskiy // Вісник Національної академії наук України. – 2013. – № 08. – P. 21–33. https://doi.org/10.15407/visn2013.08.021

DOI 10.15588/1607-3274-2022-3-17

19. Glushkov V. M. Mathematics and Cybernetics /V. M. Glushkov // Science, Technology and Global Problems /Editor(s): Academician J. G. Gvishiani. – Pergamon. – 1979. – P. 173–180. https://doi.org/10.1016/B978-0-08-024469-3.50029-2
20. Pedroni Volnei A. Finite State Machines in Harware: Theory and Design (with VHDL and SystemVerilog) /Volnei A. Pedroni. – The MIT Press, 2013. https://doi.org/10.7551/mitpress/9657.001.0001
21. Drumea, A. Finite state machines and their applications in software for industrial control / A. Drumea, C. Popescu // 27th International Spring Seminar on Electronics Technology: Meeting the Challenges of Electronics Technology Progress. – 2004. – Vol. 1. – P. 25–29. https://doi.org/10.1109/ISSE.2004.1490370
22. Гаазе-Рапопорт М. Г. От амебы до робота: модели поведения / М. Г. Гаазе-Рапопорт, Д. А. Поспелов. – М. : Наука. Гл. ред. физ.-мат. лит., 1987. – 288 с.– (Проблемы науки и технического прогресса).
23. Цетлин М. Л. Исследования по теории автоматов и моделированию биологических систем / М. Л. Цетлин. – М. : Главная редакция физико-математической литературы изд-ва «Наука», 1969. – 316 с.
24. Аверкин А. Н. Толковый словарь по искусственному интеллекту / А. Н. Аверкин, М. Г. Гаазе-Рапопорт, Д. А. Поспелов. – М. : Радио и связь, 1992. – 256 с.
25. Варшавский В. И. О поведении стохастических автоматов с переменной структурой / В. И. Варшавский // Автоматика и телемеханика. – 1969. – Т.24, № 3. – С.353–360.
26. Tîrnăucă C. Behavioral Modeling Based on Probabilistic Finite Automata: An Empirical Study / Cristina Tîrnăucă, José L. Montaña, Santiago Ontañón et al. – Sensors (Basel), 2016 Jul. – №16(7). – P. 958. Published online 2016 Jun 24. https://doi.org/10.3390/s16070958
27. Wonham W. Supervisory Control of Discrete-Event Systems / W. Wonham, K. Cai // Communication and Control Engineering. – 2019. – Springer. – 487 p. https://doi.org/10/1007/978-3-319-77452-7
28. Поляков М. А. Идентификация тепловых параметров силового масляного трансформатора по данным мониторинга параметров / М. А. Поляков // Вісник. Східноукраїнського національного університету. – 2007. – № 11. – Ч. 1 (117). – С. 167–173.
29. Непомнящих В. А. Модели автономного поискового поведения / В. А. Непомнящих // От моделей поведения к искусственному интеллекту : Коллективная монография под общ. ред. Редько В. Г. – М. : УРСС, 2006. – С. 200–242.
30. Wittenmark B. Adaptive Dual Control Methods: An Overview / B. Wittenmark. – Lund University, June. – 1995. – CiteSeerX 10.1.1.25.7446.
31. Растригин Л. А. Адаптация сложных систем / Л. А. Растригин. – Рига : Зинатне, 1981. – 375 с.
32. Poliakov, M. Performance indicators of models of non-binary control automates / M. Poliakov, S. Subbotin, O. Poliakov // Experience of Designing and Application of CAD Systems: IEEE 16th International Conference, Lviv, 22–26 Feb. 2021: proceedings. – P. 38–42. http:doi.org/10.1109/CADSM52681.2021.9385220
33. Poliakov, M. The contour of causality in control automata of systems [Electronic resource] / M. Poliakov, S. Subbotin, O. Poliakov // Proceedings of the Fourth International Workshop on Computer Modeling and Intelligent Systems (CMIS-2021). Zaporizhzhia, April 27, 2021 / ed.: S. Subbotin – Aachen: CEUR-WS, 2021. – P. 368–378. – (CEUR-WS.org, vol. 2608). – Access mode: http://ceur-ws.org/Vol-2864/paper32.pdf https://doi.org/10.32782/cmis/2864-32
34. Poliakov M. Set-theoretical FSM Models Activity Subsystem for Cognitive Control Systems / M. Poliakov, S. Subbotin, O. Poliakov /// Experience of Designing and Application of CAD Systems: IEEE 15th International Conference, Polyana, 26 Feb. – 2 March. 2019: proceedings. – P. 1–4. https://doi.org/10.1109/CADSM.2019.8779283
35. Xuesen L. Multi-behaviors Finite State Machine / L. Xuesen // Information, Computing and Telecommunication: IEEE Youth Conference on 2009: proceedings. – P. 201–203. https://doi.org/10.1109/YCICT.2009.5382390
36. Gramlich G. Probabilistic and Nondeterministic Unary Automata / G. Gramlich// Lecture Notes in Computer Science. – Springer. – 2003. – Vol. 2747. – P. 460 – 469. http://dx.doi.org/10.1007/978-3-540-45138-9_40
37. Campeanu C. Non-Deterministic Finite Cover Automata / C. Campeanu // Scientic Annals of Computer Science. – 2015. – Vol. 25 (1). – P. 3–28. http://doi.org/10.7561/SACS.2015.1.3
38. Анохин П. К. Идеи и факты в разработке теории функциональных систем / П. К. Анохин // IV-я Всероссийская научно-техническая конференция «Нейроинформатика-2002». Проблемы интеллектуального управления – общесистемные, эволюционные и нейросетевые аспекты. Материалы дискуссии / М.: Изд-во Московского государственного инженерно-физического института (МИФИ), 2003. – С. 40–57.
39. Drusinsky D. Modeling and Verification Using UML Statecharts. A Working Guide to Reactive System Design, Runtime Monitoring and Execution-Based Model Checking / D. Drusinsky. – Book Elsevier Inc. – 2006. – P. 309. https://doi.org/10.1016/B978-0-7506-7949-7.X5000-4
40. Colgren R. D. Basic MATLAB, Simulink and Stateflow [Electronic resource] / R. D. Colgren. – AIAA, 2007. – Access mode: https://arc.aiaa.org/doi/book/10.2514/4.861628
41. Контроллеры программируемые. Ч. 3 Языки программирования ГОСТ Р МЭК 61131–3–2016 (IEC 61131-3:2013, IDT). – Москва : Стандартинформ, 2016. – 227 с.
42. FSM Simulator code. Available [Electronic resource]. – Access mode: https://gitlab.com/Eonus/fsm-simulator (Accessed Feb. 24, 2022).
43. Когнитивное управление жизненным циклом изоляции обмоток маслонаполненного силового трансформатора / [М. А. Поляков, И. А. Андрияс, С. П. Конограй, В. В. Василевский] // Вісник Національного технічного університету «ХПІ». Серія: «Електричні машини та електромеханічне перетворення енергії». – 2018. – № 5 (1281). – С. 90–96.
44. Кушик Н. Г. Методы синтеза установочных и различающих экспериментов с недетерминированными автоматами. Автореферат диссертации на соискание ученой степени кандидата физико-математических наук: 05.13.01 «Системный анализ, управление и обработка информации (по отраслям)» / Н. Г. Кущик. – Томск : ТГУ, 2013. – 20 с.

DOI 10.15588/1607-3274-2022-3-17

UDC 004.852

METHOD OF IMPROVING THE ACCURACY OF NAVIGATION MEMS DATA PROCESSING OF UAV INERTIAL NAVIGATION SYSTEM

Fesenko O. D. – Lecturer of the Department of Technical and Metrological Support of the Information Technologies Faculty, Military Institute of Telecommunications and Informatization named after Heroes of Kruty, Kiev, Ukraine.

Bieliakov R. O. – PhD, Associate Professor, Senior Lecturer of the Department of Technical and Metrological Support of the Information Technologies Faculty, Military Institute of Telecommunications and Informatization named after Heroes of Kruty, Kiev, Ukraine.

Radzivilov H. D. – PhD, Associate Professor, Deputy Head for Scientific Work of the Military Institute of Telecommunications and Informatization named after Heroes of Kruty, Kiev, Ukraine.

Sasin S. A. – Senior Lecturer of the Department of Combat Application of Communications Units, Military Institute of Telecommunications and Informatization named after Heroes of Kruty, Kiev, Ukraine.

Borysov O. V. – PhD, Senior Lecturer of the Department of Construction of Telecommunication Systems, Military Institute of Telecommunications and Informatization named after Heroes of Kruty, Kiev, Ukraine.

Borysov I. V. – PhD, associate professor, Head of Research Department of the Research Institute of the Ministry of Defense of Ukraine, Kyiv, Ukraine.

Derkach T. M. – Head of the department of education of the Military Institute of Telecommunications and Informatization named after Heroes of Kruty, Kiev, Ukraine.

Kovalchuk O. O. – Senior Lecturer of the Department of Technical and Metrological Support of the Information Technologies Faculty, Military Institute of Telecommunications and Informatization named after Heroes of Kruty, Kiev, Ukraine.

ABSTRACT

Context. Modern theory and practice of preparation and conduct of hostilities on land, at sea, in the air, and recently in cyberspace dictates the relentless modernization of military equipment. The development of fundamentally new weapons is carried out considering one of the main requirements – maximum automation of operational processes, which allows combatants to distance themselves from each other as much as possible.

Among the newest models of armaments on the battlefield, due to the predominantly positional nature of the armed confrontation, unmanned aerial vehicles (UAVs) have become virtually indispensable due to their own multitasking. One of the ways to increase the efficiency of UAVs on the battlefield is to increase the level of technical perfection of flight control systems.

Creating new approaches to the design of unmanned aerial vehicle navigation systems, in particular, based on a platformless inertial navigation system is an urgent task that will provide automatic control of the UAV flight route in the absence of corrective signals from the global satellite navigation system.

Objective. The purpose of this work is to develop a method for improving the accuracy of MEMC navigation data processing of an inertial navigation system of an unmanned aerial vehicle based on an advanced Madgwik filter.

This method will increase the speed of data processing of navigation parameters and the accuracy of determining the positioning parameters in the space of the UAV through the use of an advanced Madgwik filter.

The paper shows the developed block diagram of MEMS PINS filtration on the basis of the improved Madgwik filter, the detailed mathematical description of filtration processes is carried out.

This method was tested experimentally in the MATLAB software environment using a real set of data collected during the flight of the UAV.

Method. To achieve this goal, the following methods were used: intelligent systems, theory of automatic control, pseudo-spectral method; methods based on genetic algorithm and fuzzy neural network apparatus.

Results. A method for improving the accuracy of MEMC navigation data processing of an inertial navigation system of an unmanned aerial vehicle based on an advanced Madgwik filter has been developed. The possibility of practical application of the obtained results and in comparison, with traditional methods is investigated. An experiment was performed in the MatLab software environment, and a comparison was made with the method of processing navigation data based on the Madgwik filter and the Kalman filter.

Conclusions. The developed method of increasing the accuracy of MEMC navigation data processing of an inertial navigation system of an unmanned aerial vehicle based on an advanced Madgwik filter shows an advantage over known methods in the absence of corrective signals from the global satellite navigation system for accuracy and speed of navigation data processing.

KEYWORDS: automatic control intellectual system, navigation system, unmanned aircraft vehicle.

ABBREVIATIONS

MEMS is a micro-electromechanical system;
PINS is a platformless inertial navigation system;
SLERP is a spherical linear interpolation algorithm;
LERP is a linear interpolation algorithm;
RKF is a recursive Kalman filter;
RMSE is a root mean square error.

NOMENCLATURE

ω – angular velocity;

DOI 10.15588/1607-3274-2022-3-18

$q_{\omega,t}$ – quaternion of UAV orientation;

$\omega_{q,t}^{S}$ – the resulting angular velocity vector of the local frame of reference of the gyroscope sensor (rad / s);

${}_{S}^{E}q_{,t-1}$ – the quaternion of the preliminary estimation of orientation (at t–1 step of the local frame of reference S relative to the global frame of reference E);

$\otimes$ – Hamilton's product;

Δt – time interval of initial data processing;

$q_{g,t+1}$ – forecasting quaternion;

$q_{pos,t-1}$ – quaternion of orientation of the previous state;

$a^{S} + \Delta a^{S}$ – initial accelerometer data with an errors;

g^{E} – gravity vector data;

q_{E}^{*S} – inverse quaternion orientation;

$\bar{l}$ – the rotation vector of the magnetic relative field to the sensor reference system;

$m^{S} + \Delta m^{S}$ – magnetometer data with an errors;

G – quaternion of the optimal number of operations of rotation of the vector l in the vector projection lying on the positive half-plane Z relative to the global coordinate system;

$R^{T}(q_{mag})$ – matrix of rotation of the magnetometer quaternion data;

$\hat{\Delta}q_{mag}$ – the result of calculating of the delta quaternion values of the magnetometer relative to the global frame of reference;

$\hat{\Delta}q_{acc}$ – the result of calculating of the delta quaternion the values of the accelerometer relative to the global frame of reference;

$\bar{g}_{p}^{E}$ – vector of predicted gravity relative to the global frame of reference (UAV positioning);

$\bar{u}_{m}$ – unit vector describing the axis of rotation;

$\left\|\tilde{a}^{S}\right\|$ – normalization of accelerometer data of the local frame considering of the gravitational constant $g = 9.81$ м/с^2;

ε – the threshold value of the gain;

α – gain, which characterizes the cut-off frequency of high-frequency pulses of the accelerometer signal;

$q_{\nabla(f)}$ – the quaternion is obtained using the Nesterov gradient descent algorithm;

ξ – response time when tracking the drift of zero displacement of the gyroscope;

b_{pos}^{S} – quaternion of values of orientation deviation;

$\hat{q}_{pos,t}$ – the result of calculating the values of the quaternion of orientation prediction;

β – gain that is set adaptively, based on the characteristics of the sensors and the presence of errors in inertial sensors;

θ – pitch angle navigation parameter, (deg);

φ – yaw angle navigation parameter, (deg);

ψ – roll angle navigation parameter, (deg);

S – the index of the local calculation system;

E – the index of the global calculation system;

pos – the symbol of the local calculation system;

acc – the symbol of accelerometer;

mag – the symbol of magnetometer.

INTRODUCTION

Today, navigation systems are built using completely different technologies, and can perform a wide range of functions, depending on the requirements of the technical task.

The basis of navigation systems for unmanned aerial vehicles is GPS-receivers, which in combination with the block of inertial sensors form the input data for their processing and conversion into navigation.

Thus, the presence of signals from global satellite systems is a prerequisite for maintaining the flight control process of the aircraft. The absence or pre-planned pressure of navigation signals leads to the impossibility of accurately determining their own coordinates and, as a consequence, following a certain route.

Existing methods [1–3] do not allow to ensure minimal deviation of the UAV trajectory in the autonomous mode of flight during the disappearance of signals of GPS, especially in the correlation period close to the disappearance of the GPS signal in the time interval (from 10 up to 300 s), which can be critical for the entire mission of the flight and the loss of the UAV in 38% of cases [4–6].

It is known that the determination of positioning data of UAV miniature type, as a rule, is based on an integrated MEMS free platform inertial navigation system (PINS) based on microcomputers such as Arduino Nano.

Thus, there is a need to reduce the computational load on such microcomputers during dynamic exposure to the environment, ie during nonlinear motion and in the presence of random perturbations.

The use of high-precision inertial navigation systems also does not completely solve the problem for the following reasons:

1) high cost of such systems;

2) restrictions on mass and dimensions;

3) the difficulty of minimizing errors in determining the coordinates with the time of autonomous operation.

The growing interest of scientists in intelligent control systems based on artificial neural networks, gives grounds to argue about the qualitative advantage of the latter on the performance of miniature drones. In addition, their use can significantly reduce the cost of such systems. Therefore, the intellectualization of management systems in modern conditions is one of the main scientific and practical areas of their improvement.

DOI 10.15588/1607-3274-2022-3-18

1 PROBLEM STATEMENT

Suppose that at some small UAV moving at an arbitrary constant given speed, a platformless inertial navigation system built on the basis of MEMS–sensors is installed, with the input data data with an errors $\left\{\omega_{q,t}^{S} + b_{pos}^{S}, a^{S} + \Delta a^{S}, m^{S} + \Delta m^{S}\right\}$.

To compensate for the deviations in the installation of navigation parameters $\{\theta, \varphi, \psi\}$ in the process, it is proposed to apply filtration algorithms – RKF, Madgwick and developed advanced Madgwick, with a minimum RMSE criterion in the conditions of sudden disappearance of GPS signals, in order to minimize the deviation of the UAV flight path from the one $\Delta_{pos}(t) \Rightarrow \min$.

2 REVIEW OF THE LITERATURE

Analysis of recent publications has shown that the basic principle of filtering algorithms for navigation systems of inertial sensors MEMS is based on the evaluation of data comparison of two reference systems, relative to gravity and local magnetic field, compared with the reference vectors of the output signal. However, when the local magnetic field is disturbed by ferromagnetic objects (electrical devices), which leads to problems in determining the course of the UAV, as a consequence, the need for more sophisticated filtering algorithms is stated [5].

To date, the main methods of increasing the accuracy of position estimation in the autonomous mode of UAV based on MEMS sensors of inertial navigation systems are shown in [6], which proposes an optimal algorithm that calculates the estimate in quaternion form taking into account a set of reference vectors in a fixed system. computing data in a local frame of reference relative to a UAV in space that finds the optimal quaternion by parameterizing the orientation matrix, by minimizing quadratic gain, and by using Web loss functions [7]. However, such methods have high computational requirements for sampling rate, often exceeding the bandwidth of the object.

Eston and others [8] introduced a quaternion-based filter, the filter is supplemented by a first-order model of UAV dynamics to compensate for the effect of external acceleration. Mahoney and Hemel [9] investigated the problem of estimating zero drift of a gyroscope using a passive additional filter, and proposed a solution in the form of a nonlinear correcting device, but there is a difficulty in implementing this type of navigation algorithms for micro UAV class (minimum computational computer requirements micro UAV).

Marins and others [10] propose two different approaches to solve the problems of autonomous UAV navigation based on the use of Kalman filter to assess the orientation in the quaternion form of MEMS PINS. The first approach uses each MEMS data output with a magnetometer in a 9-component state vector, which leads to the use of a complex Kalman extended filter (RKF) algorithm, the second approach uses an external Gauss-Newton algorithm to directly estimate the measurement of angular velocity quaternion's. In this case, the relationship between the process and the measurement model is linear, which allows the use of an approximate Kalman filter, but for the process of calculating object kinematics (UAV) in three projections, requires a large number of state vectors and implementation of an extended Kalman filter to linearize the problem. does not meet the requirements for the use of navigation systems based on MEMS sensors.

Scientific work [11] presents a similar approach based on improved RKF, where the process of determining the position of the UAV is based on magnetometer vector data, and the MEMS PINS error model is built as a Gauss-Markov process to predict the reduction of zero drift of the gyroscope in magnetically inhomogeneous media. The advantages of the advanced Kalman filter in [12] include the process of predicting the navigation parameters of UAVs in space using a probabilistic model, which significantly reduces the distortion of the input signals of MEMS sensors, but increases the need for computationally complex iterative processes for linear regression algorithms.

In the work of the Madgwik filter [13], a filtration algorithm with a constant gain is used to assess the state (positioning) of the UAV in quaternion form based on the MEMS data of the inertial navigation system. First, the quaternion estimate is obtained by integrating the original gyroscope data, and then corrected by the quaternion based on the accelerometer and magnetometer data. The next step of the algorithm is the process of calculating data streams using the batch gradient descent algorithm. The Madgwik method can compensate for the effect of ferromagnetic errors on the orientation component, and provides a better estimate of positioning at low computational operations.

3 MATERIALS AND METHODS

The method of increasing the MEMS data processing speed of an inertial, UAV navigation system based on an advanced Madgwik filter is based on quaternion algebra.

Formalization of the proposed method occurs in three stages:

1. Stage of forecasting. At this stage, the process of calculating the angular velocity vector based on the measurement of gyroscope data, which determines the orientation of UAV in space, first calculates the quaternion derivative, which describes the rate of change of orientation, as a product of the previous position in space on the angular velocity vector.

2. Correction stage. In this step, the correction process of navigation parameters using the delta quaternions of the magnetometer and accelerometer.

3. Stage of adaptive adjustment based on gyroscope indicators.

At the time of dynamic motion of the UAV (series of turns) with highly dynamic acceleration, the accelerometer sensor data cannot be corrected [14], so an adaptive correction factor based on gyroscope data is used using the Nesterov gradient descent algorithm [15].

Figure 1 shows a block diagram of the filter of inertial measuring devices based on the advanced Madgwik filter.

DOI 10.15588/1607-3274-2022-3-18

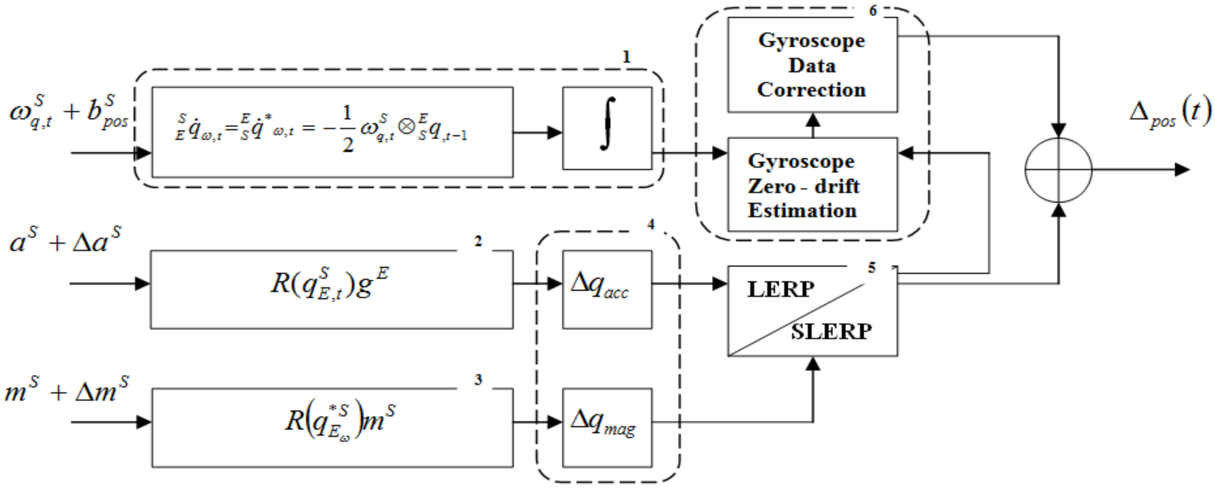


Figure 1 – Block diagram of MEMS PINS filtering based on the advanced Madgwik filter

Description of the blocks of the algorithm for increasing the data processing speed of the MEMS inertial, UAV navigation system based on the advanced Madgwik filter:

Block 1 – adjustment of the initial data of the gyroscope and integration;

Block 2 – accelerometer data processing;

Block 3 – block of magnetometer quaternion deltas;

Block 4 – accelerometer and magnetometer data filtering;

Block 5 – accelerometer and magnetometer data correction in quaternion form;

Block 6 – adaptive gyroscope data correction.

The work of the algorithm begins at the stage of forecasting and initialization of initial data.

In **block 1**, similarly to the algorithm proposed in the work of Madgwik, the initial estimation of UAV orientation in space is performed by calculating the orientation quaternion derivative using array velocity angular velocity MEMS arrays relative to the local frame of reference.

However, it should be noted that in contrast to the Madgwik algorithm, the proposed method uses the derivative of the inverse Valenti orientation function [16], which is calculated using the inverse unit of the conjugate quaternion, given in equation (1):

$$ {}_E^S\dot{q}_{\omega,t} = {}_S^E\dot{q}^*_{\omega,t} = -\frac{1}{2}\omega^S_{q,t} \otimes {}_S^E q_{,t-1}\,, \quad (1)$$

where $\omega^S_{q,t} = [0\ \omega_x\ \omega_y\omega_z]$.

In addition, there is integration (Fig. 1) by processing the input data of the gyroscope in quaternion form

$$q_{g,t+1} = q_{pos,t-1} + q'_{\omega,t}\Delta t. \quad (2)$$

In **block 2,** the data of three axial accelerometers and errors are processed. Functionally, in the MEMS module of the accelerometer, the process of measuring linear acceleration takes place, calculating the vector of the magnitude and direction of the gravitational field relative to the local coordinate system in the form of a quaternion function $R(q^S_{E,t})g^E = a^S + \Delta a^S$.

Calculation of gravity vector data allows you to find a quaternion that performs the conversion operation between two reference frames, based on accelerometer and magnetometer data:

$$R(q_{acc})R(q_{mag})\begin{bmatrix}0\\0\\1\end{bmatrix} = \begin{bmatrix}a_x\\a_y\\a_z\end{bmatrix}.$$

In **block 3,** at the output of the three-axis magnetometer is measuring the magnitude and direction of the Earth's magnetic field in the local frame of reference, taking into account local ferromagnetic distortions. The geomagnetic field is determined relative to the geographical position of the object in space, using the World Magnetic Model [17].

At the next stage of the algorithm, the delta quaternion of the magnetometer and the inverse quaternion of orientation are used, which describes the rotation vector of the magnetic field of the sensor reference system, which is shown in equation (3).

$$R\left(q^{*S}_{E_\omega}\right)m^S = l. \quad (3)$$

The next step is the process of calculating the quaternion $G = \sqrt{l_x^2 + l_y^2}$, to calculate (4)

$$R^T\left(q_{mag}\right) = \begin{bmatrix}l_x\\l_y\\l_z\end{bmatrix} = \begin{bmatrix}\sqrt{G}\\0\\l_z\end{bmatrix}. \quad (4)$$

Thus, there is a process of minimizing the influence of ferromagnetic errors on the magnetometer.

In **block 4**, there is an adaptive correction of the input data of the accelerometer and magnetometer.

DOI 10.15588/1607-3274-2022-3-18

At the first stage there is a process of forecasting the quaternion in the components of the angles of roll and tango. The result of calculating the values of the delta quaternion of the accelerometer, we obtain by the formula

$$\Delta q_{acc} = \left[\sqrt{\frac{g_z+1}{2}} - \frac{g_y}{\sqrt{2(g_z+1)}} \frac{g_z}{\sqrt{2(g_z+1)}} 0 \right]^T,$$

then there is the process of calculating the delta quaternion of the magnetometer

$$\Delta q_{mag} = \left[\frac{\sqrt{l+g_x\sqrt{g}}}{\sqrt{2g}} \; 0 \; 0 \frac{l_y}{\sqrt{2\left(g+l_x\sqrt{g}\right)}} \right]^T.$$

The magnetometer delta quaternion describes the process of rotation around the global coordinate system of the Z axis, aligning the global X axis in the positive direction of the magnetic field. With this formulation, the process of calculating the turn does not affect the components of the yaw (course) and pitch, even in the presence of magnetic perturbations, limiting their impact only on the roll angle. Thus,

$$\Delta q_{mag} = \left[\Delta q_{0mag} \; 0 \; 0 \Delta q_{3mag}\right]^T.$$

Then, subject to the receipt of the magnetic field estimate, there is a correction of the vertical component of the quaternion of orientation

$$q_E^S = q_{E_\omega}^S \otimes \hat{\Delta} q_{acc} \otimes \hat{\Delta} q_{mag}. \quad (5)$$

However, predicting the magnitude of gravity has a deviation from the real vector of gravity, so there is a correction using the delta quaternion Δq_{acc}, which converts the gravitational data of the global frame of reference q_E^S in predicted gravity g_p^E:

$$R\left(q_{E_\omega}^{*S}\right)a^S = g_p^E. \quad (6)$$

Next is the transformation of the normalized UAV positioning data vector

$$R(\Delta q_{acc}) = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} g_x \\ g_y \\ g_z \end{bmatrix}. \quad (7)$$

After solving the equation in closed form, the quaternion component is determined $\Delta q_{acc} = 0$.

The result of the accelerometer data processing provides the shortest rotation relative to the Z axis, so the vector $g_z \approx 1$.

In **block 5**, the process of scalar product Δq_{0acc} on the components of the quaternion Δq_{acc}.

Provided that $\Delta q_{0acc} > \varepsilon$, ($\varepsilon \geq 0.9$)

$$q_E^S = \Delta q_{0acc} \otimes \hat{\Delta} q_{acc}.$$

To predict the orientation quaternion in the conditions of influence of high-frequency accelerometer noise (dynamic influence on the determination of roll, UAV pitch), the interpolation algorithm of equation (8) [18] based on the identity quaternion is used $q_I = [1000]^T$, $\alpha \in [0,1]$:

$$Lerp(q_I + \Delta q_{acc}) = \left(\bar{\Delta} q_{acc}(1-\alpha)\right) q_I + \alpha \Delta q_{acc}. \quad (8)$$

The LERP algorithm does not support single normalization of the delta quaternion, so the normalization operation occurs after the application of linear interpolation (9):

$$\hat{\Delta} q_{acc} = \frac{\bar{\Delta} q_{acc}}{\left\|\bar{\Delta} q_{acc}\right\|}. \quad (9)$$

The points of the UAV orientation quaternion lie on the surface of the hyper sphere (4D), provided that, $\Delta q_{0acc} \leq \varepsilon$, therefore, spherical linear interpolation was used [18]:

$$Slerp(q_I + \Delta q_{acc}) = \frac{\sin([1-\alpha]\theta)}{\sin\theta} q_I + \frac{\sin(\alpha\theta)}{\sin\theta} \Delta q_{acc}.$$

Thus, ferromagnetic errors are compensated by the process of "merging" the data of the magnetometer and accelerometer, and switching between the respective algorithms SLERP or LERP, depending on the operating conditions of the algorithm.

In **block 6** adaptive adjustments of gyroscope data.

The process of adaptive adjustment is carried out when the UAV is moving at high acceleration and the magnitude and direction of the acceleration vector differ from gravity, so the orientation estimate can be based on erroneous navigation data, which increases the accumulation of position estimation error in space. However, it is known [8] that the indicators of the gyroscope are not affected by linear acceleration, so in this case the gyroscope data are used as the main source for evaluating the determination of UAV positioning parameters.

To solve the problem of adaptive adjustment of UAV positioning parameters, the error of setting a single vector is determined $\bar{u}_m$, which is given in the following equation:

$$\bar{u}_m = \frac{\left|\left\|\tilde{a}^S\right\| - g\right|}{g}. \quad (10)$$

The accelerometer and magnetometer data correction unit predicts a correction vector that initiates a prognosis to estimate the orientation of the local gyroscope date sensor reference system at the initial time point.

DOI 10.15588/1607-3274-2022-3-18

The process of zero drift compensation of the gyroscope in a dynamic medium is based on the algorithm of the Nesterov gradient descent [19]. This reduces the time to find the extremes of the objective function (components of the quaternion error of the gyroscope at time t).

So, $b_{pos}^{S} = \xi\left(2 \overset{*}{q}_{pos,t-1} \otimes q_{\nabla(f)}\right)$,

$$q_{\nabla(f)} = \gamma \Delta q_t + \eta q_{\nabla}\left(\overset{*}{q}\left(\hat{q}_E^S, \hat{a}^S, \hat{m}^S\right)_t - \gamma \Delta q_t\right), 0 \le \gamma \le 1,$$

$$\overset{*}{q}\left(\hat{q}_E^S, \hat{a}^S, \hat{m}^S\right)_{t+1} = q\left(\hat{q}_E^S, \hat{a}^S, \hat{m}^S\right)_t - \Delta q_{t+1}.$$

Next, the initial data of inertial sensors is normalized and, based on the obtained indicators; the UAV positioning parameters in space are predicted (11)

$$\hat{q}_{pos,t+1} = q_{g,t} - (\beta \Delta t) q_{\nabla,t}. \quad (11)$$

4 EXPERIMENTS

The experiment was conducted in the MatLab software environment using a real set of UAV flight data at speed $v_{UAV} = 40 km/h$, at the time interval of the UAV flight t={1…300}c, sampling frequency of processing of MEMS sensors navigation parameters $\Delta F = 100 Hz$.

As initial data applied navigation parameters of MEMS MPU - 9255 inertial navigation system. It is necessary to achieve a minimum deviation of the UAV flight trajectory $f\left(\hat{q}_E^S, \hat{a}^S, \hat{m}^S, \hat{b}_t^S\right) \to \Delta_{pos}(t) \Rightarrow \min$, in conditions other MEMC sensors and INS elements do not increase the angular velocity setting error.

During the experiment, attention was focused on the response of the system during the dynamic movement (series of turns) of the UAV. The phenomenon of displacement of the sensors is a signal that changes slowly over time. In order to avoid filtering of useful information, the low-pass filter is used only when the sensor is stationary. If the sensor is stationary, the offset is updated; otherwise it is assumed that the correctness of the indicators corresponds to the previous state.

The experiment compares the evaluation of the characteristics of the proposed improved Madgwik filter in different conditions with other MEMS PINS filtration methods based on the original Madgwik filter and.

At the beginning, the general characteristics are evaluated, and then the efficiency of different methods under conditions of magnetic perturbation and high non-gravitational acceleration is compared [20].

In the process of the experiment to ensure the correctness of the measurements (acceleration, angular velocity and value of the magnetic field strength) was used sensor inertial navigation system MEMS "MPU-9250".

The process of determining the orientation of the UAV during the disappearance of GPS signals, was due to the processing of acceleration data and magnetic field data.

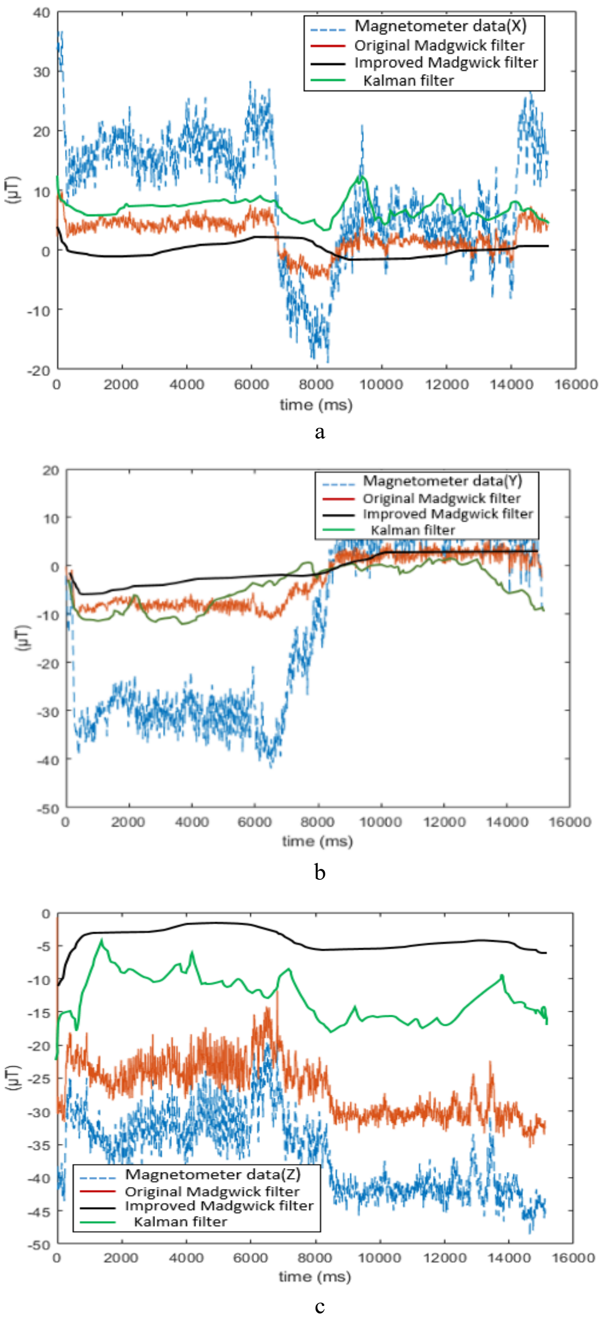


Figure 2 – The process of compensating for MEMS magnetometer errors in the process of changing the ferro-magnetic environment in the axis: a – X, b – Y, c – Z

The first two tests were to apply the effect of magnetic perturbation for 2–3 seconds, while in the third – the perturbation was static until the end of the experiment.

At the beginning of the experiment, the norm of the measured magnetic field is constant; its value differs from the norm of the reference vector of the magnetic field (0.54 Gauss). The graph (Fig. 3, 4) compares the results of three MEMS PINS filtering algorithms:

– the Madgwick filter is marked on the graph with a red line;

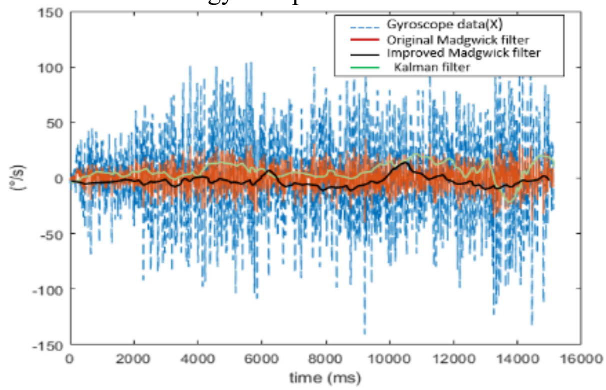
– Kalman filter with green line;

– advanced Madgwick filter with black line.

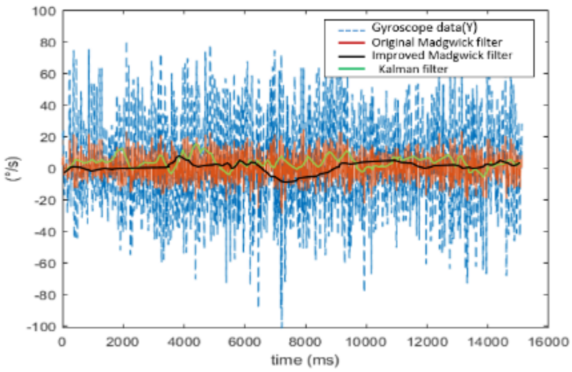
DOI 10.15588/1607-3274-2022-3-18

A popular RMSE error metric is used (Table 2), which shows an estimate of the accuracy of determining the navigation parameters of the PINS during the disappearance of the GPS.

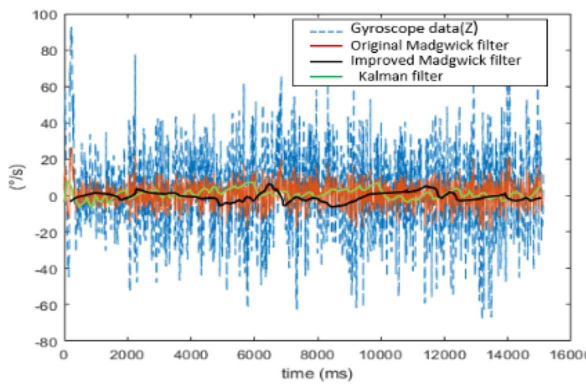
Figure 3 shows the result of the operation of filtering algorithms for the process of compensating for the shift of the drift zero of the gyroscope.



a



b



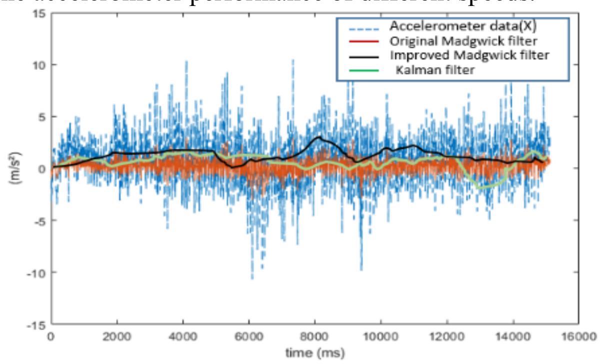
c

Figure 3 – The result of the MEMS gyroscope filtration algorithms in the axis: a – X, b – Y, c – Z

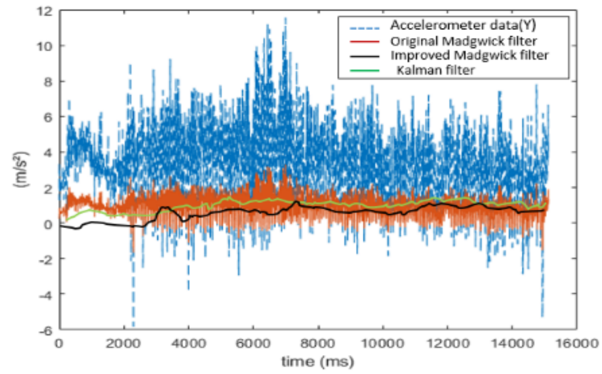
During the operation of the proposed improved Madgwik filter, the effect of ferromagnetic interference on course determination was reduced due to the two-step filtering process of the correcting delta quaternion (accelerometer and magnetometer), while in the algorithm ferromagnetic perturbations, and the restoration of the correctness of the sensor occurs when eliminating the source of ferromagnetic perturbation.

As a rule, the compensation of the drift of zero deviation of the gyroscope occurs in stationary positions in the process of finding the average value of the gyroscope or the incompatibility matrix (Jacobi) is used to linearize complex dynamic processes [10]. However, such methods are not able to eliminate the trend of drift, being in dynamic motion and also increasing the increasing computational complexity. For this purpose, it is proposed in the advanced Madgwik filtering algorithm to alternatively use **the block of the corrector of the gyroscope-quaternion**, at the time to predict the drift of zero displacement (Fig. 1).

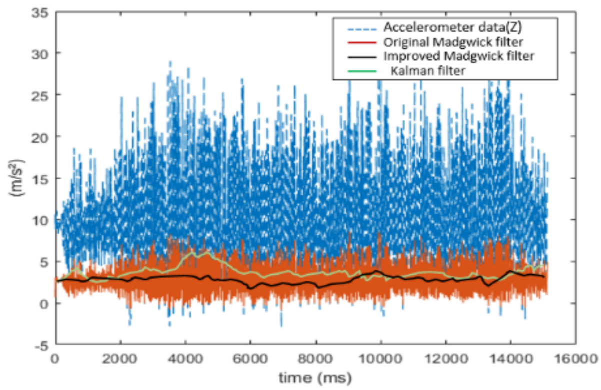
Figure 4 shows the result of the MEMS PINS filtering algorithms in the process of error compensation affecting the accelerometer performance of different speeds.



a



b



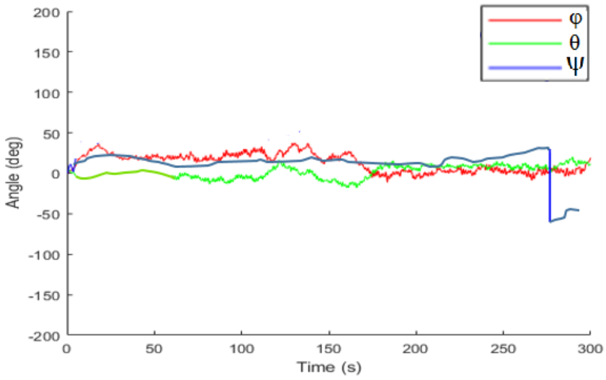
c

Figure 4 – The result of the MEMS accelerometer filtering algorithms in the axis: a – X, b – Y, c – Z

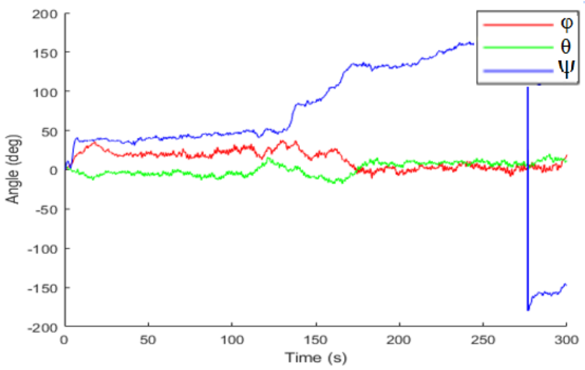
DOI 10.15588/1607-3274-2022-3-18

e-ISSN 1607-3274 Радіоелектроніка, інформатика, управління. 2022. № 3
p-ISSN 2313-688X Radio Electronics, Computer Science, Control. 2022. № 3

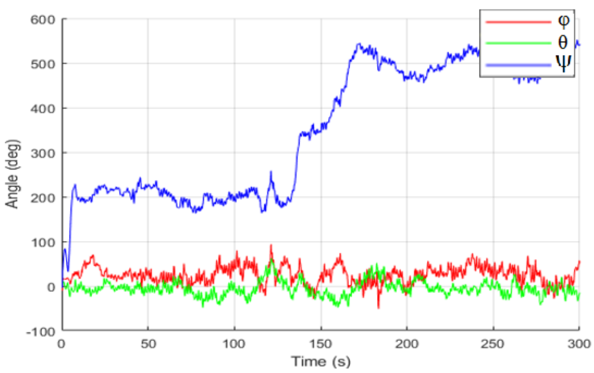
Figure 5 shows graphically the results of determining the navigation parameters (θ, φ, ψ) using the classic and advanced Madgwik filter, and the Kalman filter.



a



b



c

Figure 5 – The result of filtering to determine the navigation parameters of the orientation using
a – Advanced Madgwick filter, b – the Madgwik filter, c – Kalman filter

5 RESULTS

Evaluation of the effectiveness of the method of improving the accuracy of data processing MEMS – inertial navigation system sensors in the autonomous mode of UAV flight is performed using the software environment MatLab 2020b and Python 3.7.

Table 1–2 presents an assessment of the effectiveness of the results of filtering algorithms on the criterion of standard deviation.

The navigation estimation algorithm uses accelerometer, magnetometer, and gyroscope measurements combined into a linked coordinate system using the Earth's magnetic field and gravity vector to compensate for the zero-angle MEMS error of the gyroscope when GPS signals are lost.

Table 1 – Comparison of data processing speed MEMS signals of PINS

Algorithm	Processing time (μs)	Standard deviation
Madgwik filtration	1.2839	0.7101
Advanced Madgwick	0.9846	0.4032
RKF	7.0408	0.2342

Table 2 – The standard deviation of navigation parameters

Navigation parameters	Madgwik filtration	Advanced Madgwik	RKF
θ	1,6	**1,932**	1,9
φ	1,3	**0,292**	6,5
ψ	12,3	**1,163**	32,1

The results of applying the Kalman filter to determine the navigation parameters of the orientation showed the following: angle φ <6,5º, θ<1,9º, course angle ψ <32º, at a time interval t={1…300}s.

Low levels of accuracy of the Kalman filter are caused by the following factors: process of linearization of the dynamic UAV model reduces the accuracy of forecasting taking into account nonlinear errors of the dynamic state of the system.

The results of the application of the Madgwik filter on the value of navigation parameters of orientation showed the following: angle φ<1,3º, angle θ<1,6º, course angle ψ <32º.

Low levels of accuracy of application of the Madgwik filter are caused by the following factors: there is a difficulty of exact definition of positioning in the course of transformation of a quaternion of orientation from local system of reference of the magnetometer sensor and gyroscope, into the global frame of reference. This phenomenon occurs due to the limitation of the degree of freedom in the system of orientation equations proposed by Madgwik [13], which has two free-levels when the UAV moves in a dynamic environment, the magnitude and direction of the total measured acceleration vector different from gravity, in this case the vector state is estimated using noisy data, which leads to a significant deterioration in the determination of UAV orientation in space, the Madgwik algorithm uses a packet gradient descent to find the optimum error function of the orientation quaternion, which in turn limits the signal processing speed of MEMS PINS.

The application of the advanced Majvik filter to determine the navigation parameters of the orientation showed the following results: angle φ<0,292º, angle θ<1,93º, course angle ψ <0,16º at a time interval t={1…300}c.

DOI 10.15588/1607-3274-2022-3-18

6 DISCUSSIONS

In the framework of the work the main theoretical aspects of the method of improving the accuracy of MEMS navigation data processing of the inertial navigation system of UAVs are revealed. Implementation became possible as a result of in-depth study of existing methods of processing UAV navigation systems. With the help of experimental research of the proposed solutions it was possible to obtain the adequacy of the proposed method by comparing the results obtained with the results of their application in the MathLab software environment. Structural and functional schemes of the PINS control system, which is the basis of the methodology of the algorithm for implementing an intelligent automatic control system of the UAV control system, are given, especially in the case of short-term signals from global positioning systems.

The proposed method gives positive results in terms of a significant reduction in the standard deviation of navigation parameters, and as a result of a significant reduction in the course deviation of the UAV.

CONCLUSIONS

The proposed method based on the advanced Madgwik filter shows better speed of data processing of navigation parameters and accuracy of positioning parameters in UAV space based on PINS micro electromechanical system compared to extended filtering methods based on extended filtering. 32%, and Madgwik 20%.

The difference between the proposed method and the existing ones is as follows:

– firstly, it reduces the effect of ferromagnetic noise on the course and pitch components when the magnetometer sensor is perturbed by local ferromagnetic noise;

– secondly, the proposed method does not use complex calculations of matrix inversions while maintaining low computational costs through the use of linear interpolation algorithm;

– thirdly, the fast convergence of the UAV orientation quaternion due to the algebraic solution;

– fourth, two different gain for the process of separate filtration of different speeds and ferromagnetic noise of the magnetic field;

– fifth, during the flight of the UAV in a dynamic environment, the Nesterov gradient descent algorithm is used to calculate the component of the quaternary orientation error, while reducing computational costs and time to find the minimum error function PINS MEMS navigation parameters.

The obtained scientific result is expedient to use in control systems of unmanned aerial vehicles in a complex signal-interfering environment.

ACKNOWLEDGMENTS

The work is the result of research carried out by the employees of educational and scientific structural studies of Military Institute of Telecommunications and Informatization named after Heroes of Kruty according to Research Department of the Research Institute of the Ministry of Defense of Ukraine during the implementation of the project as part of research tasks to improve UAV control systems.

REFERENCES

1. Cox Timothy H., Nagy Christopher J. , Skoog Mark A., Somers Ivan A., Ryan Warner. Civil UAV Capability Assessment [Electronic resource]. Access mode: http://www.nasa.gov/centers/dryden/pdf/111760 main_UAV_Assessment_Report_Overview.pdf.
2. Jaramillo C., Valenti R. G., Guo L. et al. Design and Analysis of a Single Camera Omnistereo Sensor for Quadrotor Micro Aerial Vehicles (MAVs), *Sensors,* 2016, Vol. 16(2), pp. 217–218. DOI: 10.3390/s16020217.
3. Xue L., Jiang C. Y., Wang L. X. et al. Noise Reduction of MEMS Gyroscope Based on Direct Modeling for an Angular Rate Signal, *Micromachines*, 2015, Vol. 6(2), pp. 266–280. DOI: 10.3390/mi6020266.
4. Sheng G. R., Gao G. W., Zhang B. Y. Application of Improved Wavelet Thresholding Method and an RBF Network in the Error Compensating of an MEMS Gyroscope, *Micromachines,* 2019, Vol. 10, pp. 608–619. DOI: 10.3390/mi10090608.
5. Fakharian A., Gustafsson T., Mehrfam M. Adaptive Kalman filtering based navigation: An IMU/GPS integration approach, *International Conference on Networking (ICNSC 2011), Delft, 11–13 April 2011, proceedings*. Los Alamitos, IEEE, 2011, pp. 181–185. DOI: 10.1109/icnsc.2011.5874871.
6. Tian F., Zheng J. Y., Zhang T. Sensor fault diagnosis for an UAV control system based on a strong tracking Kalman filter, *Appl. Mech. Mater,* 2014, Vol. 687, pp. 270–274. DOI: 10.4028/www.scientific.net/amm.687-691.270.
7. Sampedro C. , Bavle H. , Sanchez-Lopez J. L. et al.] Flexible and dynamic mission planning architecture for UAV swarm coordination / [// IEEE International Conference on Unmanned Aircraft Systems, Arlington, USA, 7–10 June 2016 : proceedings. – Los Alamitos: IEEE, 2016. – P. 188 – 203. DOI: 10.1109/icuas.2016.7502669.
8. A complementary filter for attitude estimation of a fixed-wing UAV / [M. Euston, P. Coote, R. Mahony et al.] // IEEE International Conference on Intelligent Robots and Systems, Nice, France, 22–26 September 2008 : proceedings. – Los Alamitos: IEEE, 2008. – P. 340–345. DOI: 10.1109/iros.2008.4650766.
9. Mahony R. Nonlinear Complementary Filters on the Special Orthogonal Group / R. Mahony, T. Hamel, J. Pflimlin // IEEE Trans. Autom. Control. – 2008, Vol. 53. – P. 1203–1218. DOI: 10.1109/tac.2008.923738.
10. An extended Kalman filter for quaternion-based orientation estimation using MARG sensors / [J. L. Marins, X. Yun, E. R. Bachmann et al.] // IEEE International Conference on Intelligent Robots and Systems, Maui, HI, USA, 29 October–3 November 2001 : proceedings. – Los Alamitos: IEEE, 2002. – P. 2003–2011. DOI: 10.1109/iros.2001.976367.
11. Hajiyev C. Robust adaptive Kalman filter for estimation of UAV dynamics in the presence of sensor/actuator faults [Electronic resource] / C. Hajiyev, E. Soken. – Access mode: https://www.sciencedirect.com/science/article/abs/pii/S1270963812002027?via%3Dihub. DOI: 10.1016/j.ast.2012.12.003.
12. Shi E. An improved real-time adaptive Kalman filter for low-cost integrated GPS/INS navigation [Electronic resource. Access mode: https://ieeexplore.ieee.org/abstract/ document/6273443. DOI: 10.1109/mic.2012.6273443.

DOI 10.15588/1607-3274-2022-3-18

13. Madgwick S.O.H., Harrison A. J. L., Vaidyanathan A. Estimation of IMU and MARG orientation using a gradient descent algorithm, *IEEE International Conference on Rehabilitation Robotics, Zurich, Switzerland, 29 June–1 July 2011, proceedings*. Los Alamitos, IEEE, 2011, pp. 1–7. DOI: 10.1109/icorr.2011.5975346.
14. Hsu Y. L., Wang J. S. Random Drift Modeling and Compensation for MEMS-Based Gyroscopes and Its Application in Handwriting Trajectory Reconstruction, *IEEE Access* 2019, pp. 17551–17560. DOI: 10.1109/access.2019.2895919.
15. Nesterov Yu. Gradient methods for minimizing composite functions. Mathematical Programming [Electronic resource]. Access mode: https://link.springer.com/article/10.1007/s10107-012-0629-5. DOI: 10.1007/s10107-012-0629-5.
16. Shi G., Li X., Wang Z. A new measurement for yaw estimation of land vehicles using MARG sensors [Electronic resource]. Access mode: https://doi.org/10.1108/SR-10-2018-0276.
17. Brown A., Alken W., Macmillan P., Paniccia S. Modeling Earth's ever-shifting magnetism [Electronic resource]. Access mode: https://doi.org/10.1029/ 2021EO153457.
18. Shoemaker K. Animating Rotation with Quaternion Curves, *Conference of Special Interest Group on Graphics and Interactive Techniques 22–26 July 1985, proceedings, SIGGRAPH,* 1985, pp. 245–254. DOI: 10.1145/325165.325242.
19. Shi Y. S., Gao Z. F. Study on MEMS Gyro Signal De-Noising Based on Improved Wavelet Threshold Method, *Appl. Mech. Mater*, 2013, Vol. (433), pp. 1558–1562. DOI: 10.4028/www.scientific.net/amr.466-467.986.
20. Fesenko O., Bieliakov R., Radzivilov H. et al. Trajectory Control Method Of UAV In Autonomous Flight Mode Using Neural Network MELM Algorithm, *IEEE 2nd International Conference on Advanced Trends in Information Theory (ATIT), 15–18 December 2020, proceedings, IEEE,* 2021, pp. 114–118. DOI: 10.1109/ATIT50783.2020.9349317.

Received 14.02.2022.
Accepted 28.07.2022.

УДК 004.852

МЕТОД ПІДВИЩЕННЯ ТОЧНОСТІ ОБРОБКИ НАВІГАЦІЙНИХ ДАНИХ МЕМС ІНЕРЦІАЛЬНОЇ НАВІГАЦІЙНОЇ СИСТЕМИ БПЛА

Фесенко О. Д. – викладач кафедри Технічного та метрологічного забезпечення факультету Інформаційних технологій Військового інституту телекомунікацій та інформатизації імені Героїв Крут, Київ, Україна.

Бєляков Р. О. – канд. техн. наук, доцент, старший викладач кафедри Технічного та метрологічного забезпечення факультету Інформаційних технологій Військового інституту телекомунікацій та інформатизації імені Героїв Крут, Київ, Україна.

Радзівілов Г. Д. – канд. техн. наук, доцент, заступник начальника з наукової роботи Військового інституту телекомунікацій та інформатизації імені Героїв Крут, Київ, Україна.

Сасін С.А. – старший викладач кафедри Бойового застосування підрозділів зв'язку Військового інституту телекомунікацій та інформатизації імені Героїв Крут, Київ, Україна.

Борисов О. В. – канд. техн. наук, старший викладач кафедри Побудови телекомунікаційних систем Військового інституту телекомунікацій та інформатизації імені Героїв Крут, Київ, Україна.

Борисов І. В. – канд. техн. наук, доцент, начальник науково-дослідного управління науково-дослідного інституту Міністерства Оборони України, Київ, Україна.

Деркач Т. В. – Начальник відділення навчально-лабораторного забезпечення Військового інституту телекомунікацій та інформатизації імені Героїв Крут, Київ, Україна.

Ковальчук О.О. – старший викладач кафедри Технічного та метрологічного забезпечення факультету Інформаційних технологій Військового інституту телекомунікацій та інформатизації імені Героїв Крут, Київ, Україна.

АНОТАЦІЯ

Актуальність. Сучасна теорія і практика підготовки та ведення воєнних дій на суші, на морі, повітрі, а віднедавна і у кіберпросторі диктує невпинну модернізацію військової техніки. Розробка принципово нового озброєння здійснюється з урахуванням однієї із основних вимог – максимальної автоматизації процесів експлуатації, що дозволяє комбатантам максимально дистанціюватися один від одного.

Серед новітніх зразків озброєння на полі бою, через переважно позиційний характер ведення збройного протистояння, стали фактично незамінними, через власну мультизадачність, безпілотні літальні апарати (БПЛА). Одним з шляхів підвищення ефективності БПЛА на полі бою є підвищення рівня технічної досконалості систем керування польоту.

Створення нових підходів для проектування навігаційних систем безпілотних літальних апаратів, зокрема, на основі безплатформенної інерціальної навігаційної системи є актуальним завданням, що дозволить забезпечити автоматичне керування маршрутом польоту БПЛА за відсутності коригувальних сигналів від глобальної системи супутникової навігації.

Мета. Метою роботи є розробка методу підвищення точності обробки навігаційних даних МЕМС інерціальної навігаційної системи безпілотного літального апарату на основі вдосконаленого фільтру Маджвіка.

Зазначений метод дозволить підвищити швидкість обробки даних навігаційних параметрів та точність визначення параметрів позиціювання в просторі БПЛА за рахунок застосування вдосконаленого фільтру Маджвіка.

В роботі показано розроблену блок-схему фільтрації МЕМС БІНС на основі вдосконаленого фільтра Маджвіка, проведено деталізований математичний опис процесів фільтрації.

Зазначений метод був апробований експериментально в програмному середовищі MatLab використовуючи реальний набір даних зібраний в процесі польоту БПЛА.

Метод. Для досягнення поставленої мети використано такі методи: інтелектуальні системи, теорія автоматичного управління, псевдоспектральний метод; методи на базі генетичного алгоритму та апарат нечіткої нейронної мережі.

DOI 10.15588/1607-3274-2022-3-18

Результати. Розроблено метод підвищення точності обробки навігаційних даних МЕМС інерціальної навігаційної системи безпілотного літального апарату на основі вдосконаленого фільтру Маджвіка. Досліджено можливість практичного застосування отриманих результатів та порівняно з традиційними методами. Проведено експеримент в програмному середовищі MatLab, та проведено порівняння із методом обробки навігаційних даних на основі фільтру Маджвіка і фільтру Калмана.

Висновки. Розроблений метод підвищення точності обробки навігаційних даних МЕМС інерціальної навігаційної системи безпілотного літального апарату на основі вдосконаленого фільтру Маджвіка показує перевагу порівняно із відомими методами за відсутності коригувальних сигналів від глобальної системи супутникової навігації із точності та швидкості обробки навігаційних даних.

КЛЮЧОВІ СЛОВА: інтелектуальна система автоматичного управління, навігаційна система, безпілотний літальний апарат.

УДК 004.852

МЕТОД ПОВЫШЕНИЯ ТОЧНОСТИ ОБРАБОТКИ НАВИГАЦИОННЫХ ДАННЫХ МЭМС ИНЕРЦИАЛЬНОЙ НАВИГАЦИОННОЙ СИСТЕМЫ БПЛА

Фесенко О. Д. – преподаватель кафедры Технического и метрологического обеспечения факультета Информационных технологий Военного института телекоммуникаций и информатизации имени Героев Крут, Киев, Украина.

Беляков Р. О. – канд. техн. наук, доцент, старший преподаватель кафедры Технического и метрологического обеспечения факультета Информационных технологий Военного института телекоммуникаций и информатизации имени Героев Крут, Киев, Украина.

Радзивилов Г. Д. – канд. техн. наук, доцент, заместитель начальника по научной работе Военного института телекоммуникаций и информатизации имени Героев Крут, Киев, Украина.

Сасин С.А. – старший преподаватель кафедры Боевого применения подразделений связи Военного института телекоммуникаций и информатизации имени Героев Крут, Киев, Украина.

Борисов О. В. – канд. техн. наук, старший преподаватель кафедры Построения телекоммуникационных систем Военного института телекоммуникаций и информатизации имени Героев Крут, Киев, Украина.

Борисов И. В. – канд. техн. наук, доцент, начальник научно-исследовательского управления научно-исследовательского института Министерства Обороны Украины, Киев, Украина.

Деркач Т. М. –Начальник отделения учебно-лабораторного обеспечения Военного института телекоммуникаций и информатизации имени Героев Крут, Киев, Украина.

Ковальчук О.О. – старший преподаватель кафедры Технического и метрологического обеспечения факультета Информационных технологий Военного института телекоммуникаций и информатизации имени Героев Крут, Киев, Украина.

АННОТАЦИЯ

Актуальность. Современная теория и практика подготовки и ведения военных действий на суше, на море, воздухе, а с недавних пор и в киберпространстве диктует неустанную модернизацию военной техники. Разработка принципиально нового вооружения осуществляется с учетом одного из основных требований – максимальной автоматизации процессов эксплуатации, позволяет комбатантам максимально дистанцироваться друг от друга.

Среди новейших образцов вооружения на поле боя, через преимущественно позиционный характер ведения вооруженного противостояния, стали фактически незаменимыми, через собственную мультизадачность, беспилотные летательные аппараты (БПЛА). Одним из путей повышения эффективности БПЛА на поле боя является повышение уровня технического совершенства систем управления полетом.

Создание новых подходов для проектирования навигационных систем беспилотных летательных аппаратов, в частности, на основе бесплатформенной инерциальной навигационной системы является актуальной задачей, что позволит обеспечить автоматическое управление маршруту полета БПЛА при отсутствии корректирующих сигналов от глобальной системы спутниковой навигации.

Цель. Целью работы является разработка метода повышения точности обработки навигационных данных МЕМС инерциальной навигационной системы беспилотного летательного аппарата на основе усовершенствованного фильтра Маджвика.

Указанный метод позволит повысить скорость обработки данных навигационных параметров и точность определения параметров позиционирования в пространстве БПЛА за счет применения усовершенствованного фильтра Маджвика.

В работе показано разработанную блок-схему фильтрации МЭМС БИНС на основе усовершенствованного фильтра Маджвика, проведено детализированное математическое описание процессов фильтрации.

Указанный метод был апробирован экспериментально в программной среде MatLab используя реальный набор данных, собранных в процессе полета БПЛА.

Метод. Для достижения поставленной цели использованы следующие методы: интеллектуальные системы, теория автоматического управления, псевдоспектральний метод, методы на базе генетического алгоритма и аппарат нечеткой нейронной сети.

Результаты. Разработан метод повышения точности обработки навигационных данных МЕМС инерциальной навигационной системы беспилотного летательного аппарата на основе усовершенствованного фильтра Маджвика. Исследована возможность практического применения полученных результатов и по сравнению с традиционными методами. Проведен эксперимент в программной среде MatLab, и проведено сравнение с методом обработки навигационных данных на основе фильтра Маджвика и фильтра Калмана.

DOI 10.15588/1607-3274-2022-3-18

Выводы. Разработанный метод повышения точности обработки навигационных данных МЕМС инерциальной навигационной системы беспилотного летательного аппарата на основе усовершенствованного фильтра Маджвика, показывает преимущество по сравнению с известными методами, при отсутствии корректирующих сигналов от глобальной системы спутниковой навигации с точности и скорости обработки навигационных данных.

КЛЮЧЕВЫЕ СЛОВА: интеллектуальная система автоматического управления, навигационная система, беспилотный летательный аппарат.

ЛІТЕРАТУРА / ЛИТЕРАТУРА

1. Civil UAV Capability Assessment [Electronic resource] / Timothy H. Cox, Christopher J. Nagy, Mark A. Skoog et al.]. – Access mode: http://www.nasa.gov/centers/dryden/pdf/111760main_UAV_Assessment_Report_Overview.pdf.
2. Design and Analysis of a Single Camera Omnistereo Sensor for Quadrotor Micro Aerial Vehicles (MAVs) / [C. Jaramillo, R. G. Valenti, L. Guo et al.] // Sensors. – 2016. – Vol. 16(2). – P. 217 – 218. DOI: 10.3390/s16020217.
3. Noise Reduction of MEMS Gyroscope Based on Direct Modeling for an Angular Rate Signal / [L. Xue, C. Y. Jiang, L. X. Wang et al.] // Micromachines. – 2015. – Vol. 6(2). – P. 266–280. DOI: 10.3390/mi6020266.
4. Sheng G. R. Application of Improved Wavelet Thresholding Method and an RBF Network in the Error Compensating of an MEMS Gyroscope / G. R. Sheng, G. W. Gao, B. Y. Zhang // Micromachines. – 2019. – Vol. 10. – P. 608–619. DOI: 10.3390/mi10090608.
5. Fakharian A. Adaptive Kalman filtering based navigation: An IMU/GPS integration approach / A. Fakharian, T. Gustafsson, M. Mehrfam // International Conference on Networking (ICNSC 2011), Delft, 11–13 April 2011 : proceedings. – Los Alamitos: IEEE, 2011. – P. 181–185. DOI: 10.1109/icnsc.2011.5874871.
6. Tian F. Sensor fault diagnosis for an UAV control system based on a strong tracking Kalman filter / F. Tian, J. Y. Zheng, T. Zhang // Appl. Mech. Mater. – 2014. – Vol. 687. – P. 270–274. DOI: 10.4028/www.scientific.net/amm.687-691.270.
7. Flexible and dynamic mission planning architecture for UAV swarm coordination / [C. Sampedro, H. Bavle, J. L. Sanchez-Lopez et al.] // IEEE International Conference on Unmanned Aircraft Systems, Arlington, USA, 7–10 June 2016 : proceedings. – Los Alamitos: IEEE, 2016. – P. 188 – 203. DOI: 10.1109/icuas.2016.7502669.
8. A complementary filter for attitude estimation of a fixed-wing UAV / [M. Euston, P. Coote, R. Mahony et al.] // IEEE International Conference on Intelligent Robots and Systems, Nice, France, 22–26 September 2008 : proceedings. – Los Alamitos: IEEE, 2008. – P. 340–345. DOI: 10.1109/iros.2008.4650766.
9. Mahony R. Nonlinear Complementary Filters on the Special Orthogonal Group / R. Mahony, T. Hamel, J. Pflimlin // IEEE Trans. Autom. Control. – 2008, Vol. 53. – P. 1203–1218. DOI: 10.1109/tac.2008.923738.
10. An extended Kalman filter for quaternion-based orientation estimation using MARG sensors / [J. L. Marins, X. Yun, E. R. Bachmann et al.] // IEEE International Conference on Intelligent Robots and Systems, Maui, HI, USA, 29 October–3 November 2001 : proceedings. – Los Alamitos : IEEE, 2002. – P. 2003–2011. DOI: 10.1109/iros.2001.976367.
11. Hajiyev C. Robust adaptive Kalman filter for estimation of UAV dynamics in the presence of sensor/actuator faults [Electronic resource] / C. Hajiyev, E. Soken. – Access mode: https://www.sciencedirect.com/science/article/abs/pii/S1270963812002027?via%3Dihub. DOI: 10.1016/j.ast.2012.12.003.
12. Shi E. An improved real-time adaptive Kalman filter for low-cost integrated GPS/INS navigation [Electronic resource] / E. Shi. – Access mode: https://ieeexplore.ieee.org/abstract/ document/6273443. DOI: 10.1109/mic.2012.6273443.
13. Madgwick S. O. H. Estimation of IMU and MARG orientation using a gradient descent algorithm / S. O. H. Madgwik, A. J. L. Harrison, A. Vaidyanathan // IEEE International Conference on Rehabilitation Robotics, Zurich, Switzerland, 29 June–1 July 2011 : proceedings. – Los Alamitos: IEEE, 2011. – P. 1–7. DOI: 10.1109/icorr.2011.5975346.
14. Hsu Y.L. Random Drift Modeling and Compensation for MEMS-Based Gyroscopes and Its Application in Handwriting Trajectory Reconstruction / Y. L. Hsu, J. S. Wang // IEEE Access. – 2019. – P 17551–17560. DOI: 10.1109/access.2019.2895919.
15. Nesterov Yu. Gradient methods for minimizing composite functions. Mathematical Programming [Electronic resource] / Yu Nesterov. – Access mode: https://link.springer.com/article/ 10.1007/s10107-012-0629-5. DOI: 10.1007/s10107-012-0629-5.
16. Shi G. A new measurement for yaw estimation of land vehicles using MARG sensors [Electronic resource] / G. Shi, X. Li, Z. Wang. – Access mode: https://doi.org/10.1108/SR-10-2018-0276.
17. Modeling Earth's ever-shifting magnetism [Electronic resource] / [A. Brown, W. Alken, P. Macmillan, S. Paniccia]. – Access mode: https://doi.org/10.1029/ 2021EO153457.
18. Shoemaker K. Animating Rotation with Quaternion Curves / K. Shoemaker // Conference of Special Interest Group on Graphics and Interactive Techniques 22–26 July 1985 : proceedings. – SIGGRAPH, 1985. – P. 245–254. DOI: 10.1145/325165.325242.
19. Shi Y.S. Study on MEMS Gyro Signal De-Noising Based on Improved Wavelet Threshold Method / Y. S. Shi, Z. F. Gao // Appl. Mech. Mater. – 2013. – Vol. (433). – P. 1558–1562. DOI: 10.4028/www.scientific.net/amr.466-467.986.
20. Trajectory Control Method Of UAV In Autonomous Flight Mode Using Neural Network MELM Algorithm / [O. Fesenko, R. Bieliakov, H. Radzivilov et al.] // IEEE 2nd International Conference on Advanced Trends in Information Theory (ATIT), 15–18 December 2020 : proceedings. – IEEE. – 2021. – P. 114–118. DOI: 10.1109/ATIT50783.2020.9349317.

DOI 10.15588/1607-3274-2022-3-18

Наукове видання

Радіоелектроніка,

інформатика,

управління

№ 3/2022

Науковий журнал

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 14.09.2022. Формат 60×84/8.
Папір офс. Різогр. друк. Ум. друк. арк. 24,18.
Тираж 300 прим. Зам. № 715.

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

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