The controller synthesis automation using a dynamic mathematical model and genetic algorithms

Victoria Vysotska[1,†], Vasyl Lytvyn[1,†], Serhii Vladov[2,*,†], Oleksandr Muzychuk[3,†] and Oleksii Kryshan[4,†]

[1] Lviv Polytechnic National University, Stepan Bandera Street 12 79013 Lviv, Ukraine
[2] Kremenchuk Flight College of Kharkiv National University of Internal Affairs, Peremohy Street 17/6 39605 Kremenchuk, Ukraine
[3] Kharkiv National University of Internal Affairs, L. Landau Avenue 27 61080 Kharkiv, Ukraine
[4] Interregional Academy of Personnel Management, Frometivska Street 2 03039 Kyiv, Ukraine

Abstract

This research develops a dynamic mathematical model for controller synthesis using genetic algorithms, surpassing traditional methods by integrating adaptive optimization with evolutionary principles. Unlike conventional fixed algorithms and manual tuning, this model dynamically explores a broader parameter space and autonomously adjusts parameters, enhancing performance in complex systems. The research involved a computer experiment applying this model to a PID controller implemented with a dynamic neural network. The results demonstrated significant improvements, including adjustments to PID coefficients that reduced transient process time and overshoot, increased modeling accuracy from 99.523 to 99.783 %, and minimized losses from 2.5 to 0.5 %. These findings suggest that incorporating the developed model into the automatic control system for helicopter turboshaft engines free turbine rotor speed could significantly enhance system performance and reliability. These findings suggest that incorporating the developed model into the automatic control system for helicopter turboshaft engines free turbine rotor speed could significantly enhance system performance and reliability. Future work will focus on validating these results under diverse operational conditions and exploring additional optimization techniques.

Keywords

PID controller, genetic algorithms, mathematical model, optimal control, neural network, helicopter turboshaft engine, free turbine rotor speed

1. Introduction

Automation of controller synthesis is a critical task in control complex dynamic systems, where traditional design methods may be insufficiently effective [1]. In recent years, there has been growing interest in the application of optimization methods based on evolutionary algorithms, among which genetic algorithms hold a prominent position [2, 3]. These algorithms, inspired by the natural selection and genetic evolution processes, enable effective

[1]*BAIT'2024: The 1st International Workshop on "Bioinformatics and applied information technologies", October 02-04, 2024, Zboriv, Ukraine*
* Corresponding author.
† These authors contributed equally.
victoria.a.vysotska@lpnu.ua (V. Vysotska); vasyl.v.lytvyn@lpnu.ua (V. Lytvyn); serhii.vladov@univd.edu.ua (S. Vladov); o.muzychuk23@gmail.com (O. Muzychuk); krishan@ki-maup.com.ua (O. Kryshan)
0000-0001-6417-3689 (V. Vysotska); 0000-0002-9676-0180 (V. Lytvyn); 0000-0001-8009-5254 (S. Vladov); 0000-0001-8367-2504 (O. Muzychuk); 0000-0002-2967-0126 (O. Kryshan)

solutions to optimization tasks in multidimensional spaces, where traditional methods may encounter difficulties [4, 5]. In the controller synthesis context, genetic algorithms provide flexibility and adaptability, allowing for the multiple criteria consideration and modern control systems constraints characteristic.

This research relevance is driven by the modern dynamic systems increasing complexity that require highly efficient and adaptive control methods. Traditional approaches to controller synthesis are often limited due to the systems and multi-criteria requirements complexity, which can lead to suboptimal solutions or excessive design costs. In this context, genetic algorithms, with their ability to effectively search for global optima in multidimensional spaces and adapt to changing conditions [6–8], offer new opportunities for automating controller synthesis. The genetic algorithms application significantly enhances the controller synthesis accuracy and reliability, which is particularly important for control critical systems such as aviation and energy complexes, where failures can have severe consequences.

2. Related Works

Existing research in the controller synthesis automation field using genetic algorithms has shown significant progress in solving optimization tasks for complex dynamic systems. Many researches emphasize the genetic algorithms flexibility and adaptability, allowing the various quality criteria consideration and constraints inherent in real-world control systems [9]. Research demonstrates that genetic algorithms are successfully applied in various industries, including aerospace [10–12], energy [13, 14], robotics [15, 16], and others [17–19]. Their ability to effectively solve multi-objective optimization tasks [20] makes them an attractive tool for automating controller synthesis, especially under conditions of high uncertainty and the numerous local optima presence.

One important area of research is the hybrid methods development that combine genetic algorithms with other optimization techniques, such as neural networks [21–23] and particle swarm [24, 25] methods. These approaches improve convergence and the finding optimal solutions speed, as well as enhance the algorithms against local minima resilience. Notably, successful results have been obtained when applying hybrid methods for synthesizing controllers in complex nonlinear systems, where traditional methods prove to be less effective [26–28]. However, such hybrid approaches require more precise tuning and complicate the process of model development.

Despite the advances in applying genetic algorithms for controller synthesis, there remain several unresolved issues related to these processes mathematical modeling. Specifically, the genetic algorithms scalability when working with large and high-dimensional systems remains insufficiently studied. Additionally, there are limitations in assessing algorithm performance in the synthesis early stages, which can lead to increased computation time and reduced automation efficiency. Moreover, the literature highlights a lack of development in methods for adapting genetic algorithms to changing environmental conditions and parameter instability, which limits their application in real-time systems.

Therefore, further development of mathematical models for automating controller synthesis using genetic algorithms is necessary, focusing on improving the scalability and adaptability of algorithms, as well as reducing computational costs. Research in this direction may include developing new approaches to the genetic algorithm parameters adaptive tuning, enhancing hybridization methods with other optimization techniques, and exploring ways to integrate genetic algorithms into real-time control systems. These research gaps present opportunities for developing more effective and versatile tools for automating controller synthesis, capable of handling control tasks in increasingly complex and uncertain environments.

3. Proposed technique

Genetic algorithms have proven their effectiveness in solving optimization tasks, especially in cases where classical methods encounter difficulties due to the multidimensionality and nonlinearity of tasks [29, 30]. This research presents an innovative dynamic mathematical model for the controllers' synthesis automating based on the genetic algorithms use. The model includes several stages: the target function formation, the controller parameters determination, the genetic algorithm development for the controller optimization and dynamic adaptation during the system operation [31].

Let $y(t)$ be the system output variable, $y_r(t)$ be the output variable desired value (reference). The output variable from the reference deviation is defined as:

$$e(t) = y_r(t) - y(t). \tag{1}$$

The optimization task is to minimize the objective function J, which is defined as the integral of the squared error over a certain time interval T:

$$J = \int_0^T e^2(t)\,dt. \tag{2}$$

Let us consider an example of a PID controller for the helicopter turboshaft engines (TE) free turbine rotor speed n_{FT}, whose transfer function has the form [32, 33]:

$$C(s) = K_p + \frac{K_i}{s} + K_d \cdot s, \tag{3}$$

where K_p, K_i, and K_d are the proportional, integral, and differential components coefficients, respectively. The genetic algorithm task is to find such values of K_p, K_i, and K_d that minimize the objective function J.

The genetic algorithm application first stage is the initialization of the population. Let N be the population size, then the initial population $P(0)$ can be defined as a set of random values K_p, K_i, and K_d:

$$P(0) = \{(K_{p1}, K_{i1}, K_{d1}), \ldots, (K_{pN}, K_{¿}, K_{dN})\}. \tag{4}$$

Next, the fitness function is evaluated. Each population member is evaluated by the objective function J value as:

$$J_i = \int_0^T e_i^2(t)\,dt, i = 1 \ldots N, \tag{5}$$

Next, it performs a crossover as follows: two parent solutions P_i and P_j produce an offspring P_k using a parameters combination:

$$P_k = \alpha \cdot P_i + (1 - \alpha) \cdot P_j, \alpha \in [0, 1], \tag{6}$$

Next, mutation is carried out by randomly changing one or more parameters of the offspring:

$$P_k = P_k + \delta, \delta \sim N(0, \sigma^2). \tag{7}$$

For the selection process, where M best solutions that minimize the objective function J survive, an algorithm is applied in which there is a solutions population $\{P_1, P_2, \ldots, P_N\}$, where N is the solutions total number. For each solution, the objective function $J(P_i)$ value is calculated according to (5).

At the next stage, the solution is sorted by the objective function $J(P_i)$ value in ascending order:

$$P_{(1)}, P_{(2)}, \ldots, P_{(N)}, \tag{8}$$

where P_i is the solution for which the $J(P_{(i)})$ value is less than for all previous solutions:

$$J(P_{(1)}) \leq J(P_{(2)}) \leq \ldots \leq J(P_{(N)}). \tag{9}$$

After sorting, M best solutions are selected for which $J(P_{(i)})$ is minimal:

$$J(P_{(1)}) \leq J(P_{(2)}) \leq \ldots \leq J(P_{(M)}), \tag{10}$$

where $M \leq N$, and the selected solutions $\{P_{(1)}, P_{(2)}, \ldots, P_{(M)}\}$ become the basis for forming a new population in the algorithm next step. A new population is formed from descendants and parents:

$$P(t + 1) = \{P_{(1)}, P_{(2)}, \ldots, P_{(M)}, \text{new descendants}\}. \tag{11}$$

This process continues until a stopping criterion is reached, such as when the minimum value is reached or when the iterations maximum number is reached.

In real systems, the system parameters may change over time, which requires the regulator to adapt in real time. For this aim, an algorithm is proposed that consists of modeling the change in system parameters, controller adaptive adjustment, and feedback. Let the system parameters $a(t)$, $b(t)$, and $c(t)$ change over time:

$$\dot{x}(t) = A(t) \cdot x(t) + B(t) \cdot u(t), \tag{12}$$

$$y(t) = C(t) \cdot x(t),$$

where $A(t) = a(t)$, $B(t) = b(t)$, $C(t) = c(t)$ are the system matrices that change over time.

The controller adaptive tuning assumes that the controller parameters can change over time in response to changes in the control object dynamics. Let us consider a mathematical model for the PID controller adaptive tuning that updates the controller coefficients $K_p(t)$, $K_i(t)$, and $K_d(t)$ at each time step. The helicopter TE free turbine rotor speed n_{FT} PID controller is presented in the form [32, 33]:

$$u(t) = K_p(t) \cdot e(t) + K_i(t) \cdot \int_0^t e(\tau)\, d\tau + K_d(t) \cdot \frac{de(t)}{dt}, \tag{13}$$

where $e(t) = y_r(t) - y(t)$ is the control error calculated according to (1), $y_r(t)$ is the specified output value (reference).

To estimate the system parameters $A(t)$, $B(t)$, and $C(t)$ in real time, the recursive least squares (RLS) method [34] can be used:

$$\hat{\theta}(t) = \hat{\theta}(t-1) + L(t) \cdot \left[y(t) - \hat{y}(t) \right], \tag{14}$$

where $\hat{\theta}(t)$ is the system parameters estimate at time t, $L(t)$ is the adaptation matrix, $\hat{y}(t) = \hat{C}(t) \cdot \hat{x}(t)$ is the system output estimate, $\hat{x}(t)$ is the system state estimate.

The controller parameters are updated using gradient descent:

$$K_p(t+1) = K_p(t) - \alpha \cdot \frac{\partial J(t)}{\partial K_p}, \tag{18}$$

$$K_i(t+1) = K_i(t) - \alpha \cdot \frac{\partial J(t)}{\partial K_i}, \tag{19}$$

$$K_d(t+1) = K_d(t) - \alpha \cdot \frac{\partial J(t)}{\partial K_d}, \tag{20}$$

where $J(t)$ is the objective function (for example, the squared error integral), α is the training step.

The objective function gradients for each of the parameters can be calculated as follows:

$$\frac{\partial J(t)}{\partial K_p} = -2 \cdot \int_0^t e(\tau) \cdot \frac{\partial u(\tau)}{\partial K_p} d\tau, \tag{21}$$

$$\frac{\partial J(t)}{\partial K_i} = -2 \cdot \int_0^t e(\tau) \cdot \frac{\partial u(\tau)}{\partial K_i} d\tau, \tag{22}$$

$$\frac{\partial J(t)}{\partial K_d} = -2 \cdot \int_0^t e(\tau) \cdot \frac{\partial u(\tau)}{\partial K_d} d\tau . \tag{23}$$

Because:

$$\frac{\partial u(t)}{\partial K_p} = e(t) , \tag{24}$$

$$\frac{\partial u(t)}{\partial K_i} = \int_0^t e(\tau) d\tau , \tag{25}$$

$$\frac{\partial u(t)}{\partial K_d} = \frac{\partial e(t)}{\partial t} . \tag{26}$$

Substituting expressions (24)–(26) into the formulas for gradients (21)–(23), we obtain:

$$\frac{\partial J(t)}{\partial K_p} = -2 \cdot \int_0^t e(\tau) \cdot e(\tau) d\tau = -2 \cdot \int_0^t e^2(\tau) d\tau , \tag{27}$$

$$\frac{\partial J(t)}{\partial K_i} = -2 \cdot \int_0^t e(\tau) \cdot \left(\int_0^\tau e(\xi) d\xi \right) d\tau , \tag{28}$$

$$\frac{\partial J(t)}{\partial K_d} = -2 \cdot \int_0^t e(\tau) \cdot \frac{\partial e(\tau)}{\partial t} d\tau . \tag{29}$$

Thus, at each time step, the parameters $K_p(t)$, $K_i(t)$, and $K_d(t)$ are updated taking into account the control error current values and its derivatives, which allows the controller to adapt to changes in the system dynamics.

Feedback is a control system key element that allows the system behavior to be adjusted based on the system current state deviation from a given value. To control a nonlinear system, the input-output linearization method is used, which consists of selecting the control action so that the closed system becomes linear. The control signal is selected as:

$$u(t) = \frac{1}{g(x(t))} \cdot \left(\frac{-d^{(r)} \cdot h(x(t))}{dt^{(r)}} + v(t) \right) , \tag{30}$$

where r is the system relative degree, and $v(t)$ is the linear system new control action. Substituting $u(t)$ into the original equation, we obtain the linear system:

$$v(t)=\frac{d^{(r)}\cdot y(t)}{dt^{(r)}}. \tag{31}$$

The control task is reduced to choosing $v(t)$ so that the output signal $y(t)$ desired trajectory is achieved.

The developed dynamic mathematical model for the controllers synthesis automating using genetic algorithms represents a significant advancement over traditional approaches by integrating adaptive optimization mechanisms with evolutionary principles. Unlike conventional methods that rely on fixed algorithmic structures and heuristic tuning, this model employs genetic algorithms to dynamically explore and optimize a broader parameter space, facilitating the more effective and robust controller configurations discovery. The model's innovation lies in its ability to autonomously adjust controller parameters through a continuous evolutionary process, thereby enhancing performance in complex and variable system environments. This adaptive capability contrasts with traditional methods that often require manual intervention and lack the flexibility to respond to real-time changes in system dynamics, thus offering a more versatile and efficient solution for modern control applications.

4. Results

Based on [32, 33], the work solves the task of synthesizing a PID controller (Figure 1) for the helicopters TE free turbine rotor speed n_{FT} controlling task.

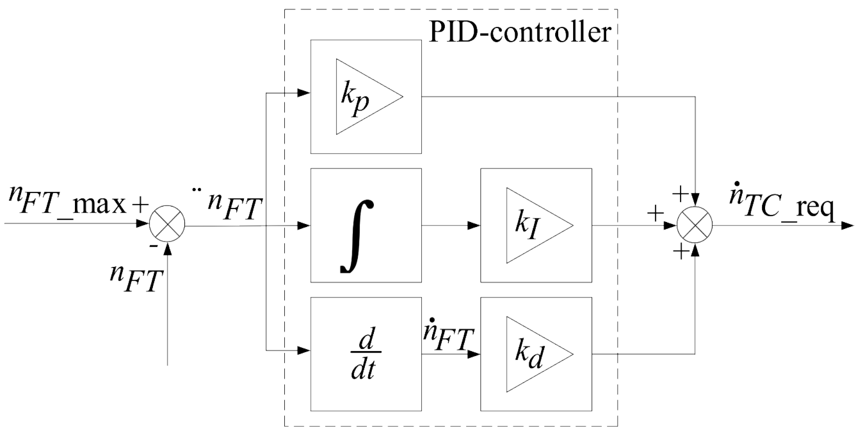

Figure 1: Updated block diagram illustrating the system for the helicopter turboshaft engines free turbine rotor speed controlling using a PID controller (author's research).

The input parameters are the atmospheric parameters (h is the flight altitude, T_N is the temperature, P_N is the pressure, ρ is the air density) values. The parameters recorded on board of the helicopter (n_{FT} is the free turbine rotor speed) reduced to absolute values according to the gas-dynamic similarity theory (table 1). We assume in this research that the atmospheric parameters are constant (h is the flight altitude, T_N is the temperature, P_N is the pressure, ρ is the air density) [35–37].

Table 1
Training dataset fragment

Number	n_{FT}	Number	n_{FT}
1	0.983	132	0.992
…	…	…	…
28	0.979	…	…
…	…	256	0.974

The training dataset homogeneity evaluation, as described in [35–37], employed the Fisher-Pearson [38] and Fisher-Snedecor [39] tests. Based on these criteria, the dataset was found to be homogeneous, with the computed Fisher-Pearson and Fisher-Snedecor statistics falling below their respective critical limits, specifically $\chi^2 = 5.365 < \chi^2_{critical} = 6.6$ and $F = 2.177 < F_{critical} = 2.58$. To assess the dataset's representativeness, cluster analysis using the k-means method [40–42] was conducted. The dataset was divided into training and testing subsets in a 2:1 ratio, corresponding to 67 % (172 samples) and 33 % (84 samples), respectively. The cluster analysis (Table 1) identified seven distinct classes (I…VII), confirming their presence and demonstrating consistency between the training and test datasets (Figure 2). These results allowed for the optimal sample datasets determination: the training dataset consists of 256 elements (100 %), the validation dataset comprises 172 elements (67 % of the training dataset), and the test dataset contains 84 elements (33% of the training dataset).

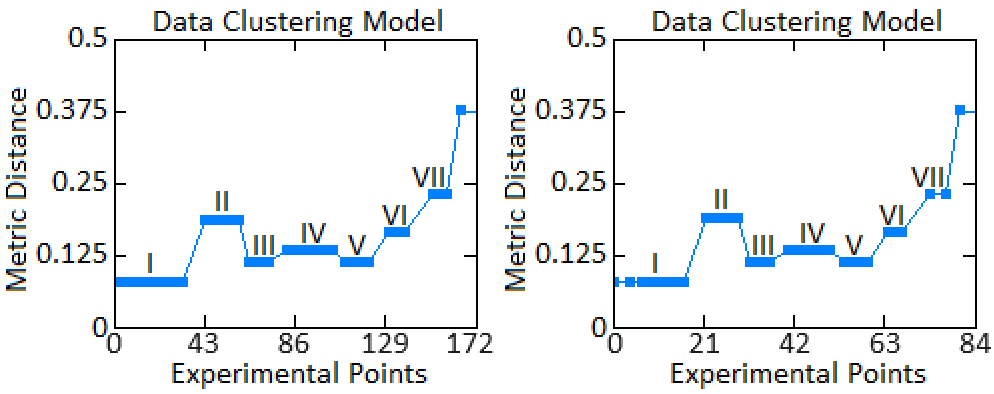

Figure 2: The cluster analysis results, where "left figure" is the training dataset, "right figure" is the test (author's research) (author's research).

As detailed in [32, 33], an improved standard configuration for the helicopter TE free turbine rotor speed controlling has been introduced, featuring a PID controller. This enhancement is realized by integrating a dynamic neural network with direct data transmission, where the first layer consists of neurons with a radial-basis activation function, and the second layer includes adalines are the neurons with a linear activation function. Since maintaining the main rotor rotational speed is a critical objective during helicopter flight, this modification is especially relevant. In [32, 33], a neural network was employed to adjust the PID controller K_p, K_d, and K_i coefficients, which ultimately enabled the main rotor speed dynamic regulation.

In [33] it is implemented a discrete PID controller using a dynamic neural network with direct data flow, where only the activation function in the second layer is linear, while the activation functions in the first layer are nonlinear, specifically radial-basis functions (see Figure 3, where Δ represents a one-step delay).

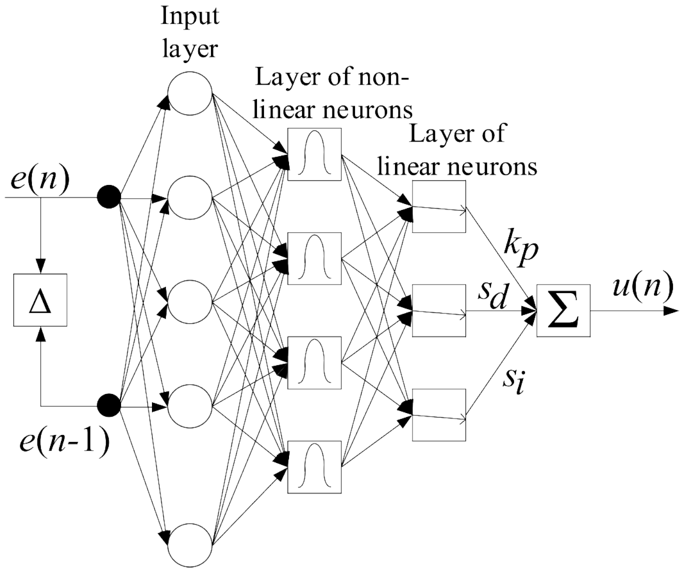

Figure 3: The nonlinear PID controller neural network-based representation (author's research).

To fine-tune the PID controller, a dynamic neural network with direct data flow was utilized. This network comprises neurons with a radial basis activation function in the first layer and adalines is the neurons with a linear activation function is the in the second layer. The network's structural parameters include a learning rate set to 1.5, 20 neurons in the hidden layer, a delay line length of 5 for input signals, and 1000 training epochs. The initial coefficients were set at K_p = 0.5, K_i = 5, K_d = 0.01. For initial training, recorded data from the control object's operation were used, specifically the first 50 data points, with a window size of 10 and a training accuracy of 0.00005. If during operation the control error exceeded 2 over ten time cycles, the neural network underwent further training, with a training accuracy of 0.0001 and a limit of 10 training iterations. As per [32, 33], the first step involved optimizing the PID controller coefficients (Figure 1). The transient response after optimization is depicted by curve 1 in Figure 4, where the linear control law resulted in significant overshoot and a large static error. The transient response is shown as "blue curve" in Figure 4, which indicates that while the response became aperiodic, the static error remained, and the rise time increased slightly. In the third step, the neuron's activation function parameters, corresponding to the integral component of the PID controller, were optimized (Figure 3). This optimization resulted in the transient response shown as "red curve" in Figure 5, where the response became oscillatory again. It is evident that with changes in the task, the transient response remained consistent, maintaining a low overshoot due to the PID coefficients static nature. The overshoot was minimal, the slew rate significantly improved, and the static error was negligible.

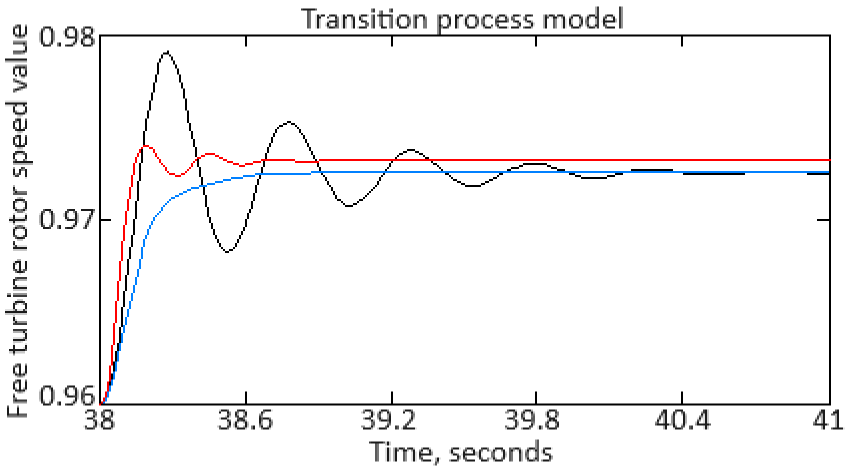

Figure 4: The PID controller computer simulation results (author’s research).

According to Figure 4 initial coefficient values are: K_p = 0.5, K_i = 5, K_d = 0.01, after the add-on K_p = 0.45, K_i = 4.985, K_d = 0.007, overshoot value does not exceed 0.3 %.

To evaluate the neural network's performance in the next training phase, both accuracy (Figure 5) and loss (Figure 6) are measured. The Accuracy metric represents the correct predictions percentage, while the Loss metric shows the predictions average squared error, indicating how much they deviate from the true values. To evaluate the precise calculations proportion for the free turbine rotor speed n_{FT}, the Accuracy metric is used (Figure 5), and it is computed at training epoch t using the following expression [35–37]:

$$Accuracy_t = \frac{1}{N} \cdot \sum_{i=1}^{N} I\left(\hat{n}^t_{TCi} = n_{TC}\right). \tag{32}$$

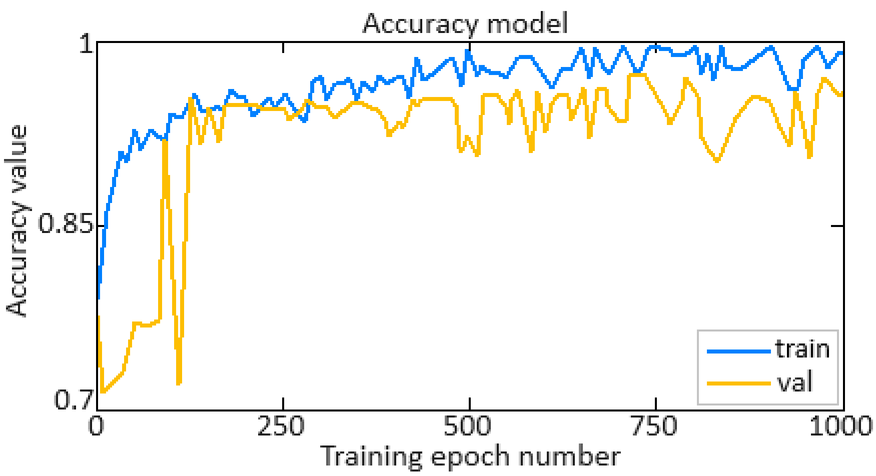

Figure 5: The accuracy metric diagram (author's research).

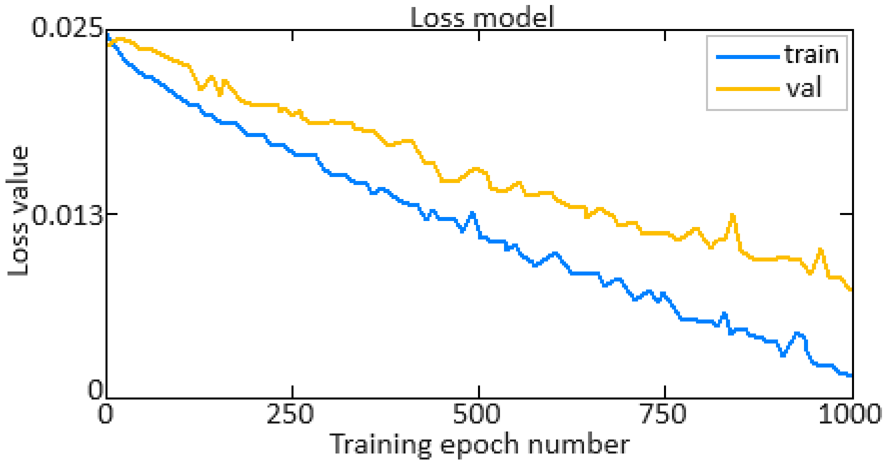

Figure 6: The loss metric diagram (author's research).

As shown in Figures 5 and 6, these metrics indicate that the neural network model delivers high prediction accuracy (99.77 %) and performs efficiently, with the mean squared error staying under 2.5 %. Furthermore, the significant reduction in the loss function from 2.5 to 0.5 % reflects an enhancement in the model's quality over the training process course.

Similarly to the method described in [32, 33], the approach quality is assessed using classification metrics derived from the confusion matrix presented in Table 2. In this matrix, *TP* denotes true positives (instances correctly classified as defects by the model), *FP* refers to false positives (non-defects mistakenly classified as defects), *TN* represents true negatives (cases accurately identified as non-defective data), and *FN* indicates false negatives (defects incorrectly identified as non-defective) [43, 44].

Table 2

The error matrix [43, 44]

Category	Positive	Negative
Predicted positive	TP	FP
Predicted negative	FN	TN

To assess the developed method quality in this research, the following metrics were chosen [45–47]: $Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$ is the objects percentage calculates for which the classifier accurately made decisions, $Accuracy = \frac{TP}{TP+FP}$ is the relevant parameters percent among all researched, $Recall = \frac{TP}{TP+FN}$ is the crucial parameter in defect detection is precision, as the detected defects ratio signifies to the defective instances overall number, $F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$ is the *F*-measure, which is the "harmonic" average between

Precision and *Recall*. Table 3 presents the model's training outcomes average results, including the mean and variance for the accuracy metrics.

Table 3
The testing indicators average values

Metric	Value	
	PID-controller developed on the neural network basis [33]	PID-controller developed on the neural network basis with genetic algorithms
Accuracy	0.99523	0.99783
Precision	0.96238	0.98472
Recall	1.0	1.0
F-measure	0.98165	0.99362
Average time, seconds	1201.99	1095.38
Average Accuracy	0.99319	0.99525
Dispersion Accuracy	0.00000886	0.00000322

Table 4 provides an accuracy comparative analysis provided by each of the evaluated controllers, highlighting the Type I and Type II errors [48–50] probabilities in identifying the optimal parameter n_{FT}.

Table 4
The 1st and 2nd types errors determining results

Controller type	Error probability in determining the parameter n_{FT} optimal value	
	Type 1st error	Type 2nd error
Linear PD-controller	1.95	1.42
PD-controller with reduced K_d	1.74	1.21
Quadratic controller	1.46	1.03
PD-controller with a variable amplification	1.32	0.95

factor		
Fuzzy logical P-controller	1.08	0.77
Fuzzy logical P-controller with a corrective differential link	0.97	0.64
PID-controller developed on the neural network basis [32]	0.58	0.22
Modified PID-controller developed on the neural network basis [33]	0.36	0.14
PID-controller developed on the neural network basis with genetic algorithms	0.22	0.10

As shown in Table 4, incorporating a dynamic neural network with direct data flow into the PID controller with genetic algorithms, where the first layer consists of neurons with a radial basis activation function and the second layer includes adalines with a linear activation function is led to a decrease in Type I and Type II errors by 30 to 40 % compared to the PID-controller described in [33].

5. Discussions

The research is aimed at creating a dynamic mathematical model (1)–(31) for the controllers (see Figure 1) synthesis using genetic algorithms. The new dynamic mathematical model for controller synthesis, utilizing genetic algorithms, surpasses traditional methods by integrating adaptive optimization with evolutionary principles. Unlike fixed algorithms and manual tuning, this model dynamically explores a broader parameter space, autonomously adjusting parameters for improved performance in complex systems. This approach offers greater flexibility and efficiency, responding to real-time changes more effectively than conventional methods.

In this research, a computer experiment was conducted to determine the helicopter TE free turbine rotor speed transient process by introducing the developed mathematical model (1)–(31) into a PID controller implemented utilizing a dynamic neural network with direct data flow, where only the second layer has a linear activation function, while the first layer features nonlinear, radial-basis activation functions (see Figure 3).

According to the obtained results (see Figure 4), the developed mathematical model (1)–(31) application made it possible to adjust the PID controller coefficients: the K_p coefficient from 0.5 to 0.45, the K_i coefficient from 5.0 to 4.985, the K_d coefficient from 0.01 to 0.007, which made it possible to minimize the transient process time and overshoot from 0.5 to 0.3% compared to the PID controller developed in [32, 33].

The developed mathematical model (1)–(31) application made it possible to increase the accuracy of the helicopter TE free turbine rotor speed transient process modeling from 99.523

to 99.783 % (see Figure 5 and Table 3), and also to minimize loss from 2.5 to 0.5% (see Figure 6), and also reduce the first and second types errors by 30...40 % compared to the PID controller developed in [32, 33].

Thus, it seems appropriate to implement the developed mathematical model (1)–(31) into the helicopter TE free turbine rotor speed automatic control system, including a neural network, which task is to determine the PID controller coefficients (Figure 7).

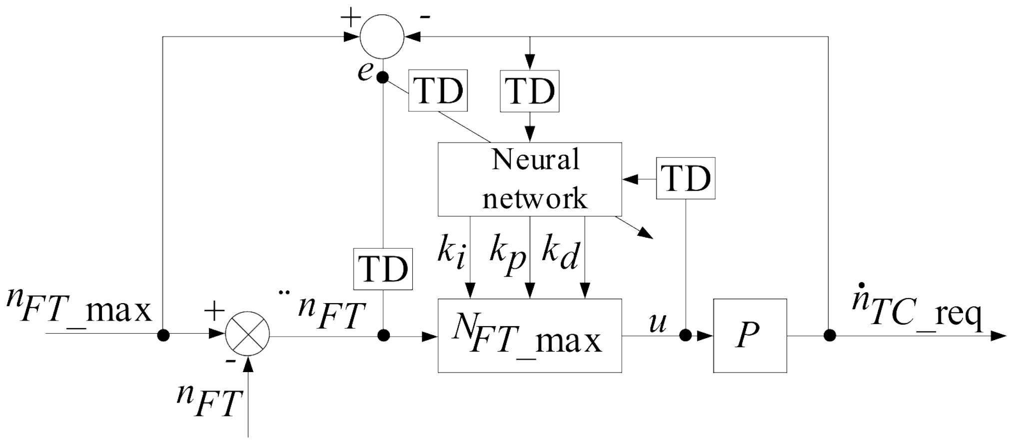

Figure 7: Modified PID controller design with an auto-tuning block based on a neural network, where TD represents the delay operator (author’s research).

The limitations of the obtained results may be as follows. First, the proposed mathematical model (1)–(31) and optimization methods based on genetic algorithms might not account for all possible external and internal influences, limiting their applicability to real-world operational conditions. Second, the improvement in modeling accuracy and reduction in losses achieved by adjusting PID controller coefficients may be due to the specific parameters and neural network architecture used, which does not necessarily guarantee similar results in other systems or under different operating conditions. Third, the experimental data used in the study may not fully capture complex dynamic processes, potentially affecting the proposed method accuracy and reliability in various real-world scenarios. Specifically, the controller's performance accuracy is contingent on the sensor data precision, and any significant deviation in sensor readings can negatively affect the dynamic response. Additionally, the genetic algorithm computational complexity, combined with the neural network, can lead to increased processing time, particularly in real-time applications, which may necessitate further optimization for onboard systems.

Future research prospects include several key directions. First, additional experiments are needed to verify the proposed mathematical model and optimization methods universality under various operational conditions and different system types. Second, integrating the model with more advanced adaptive control algorithms should be considered to enhance its ability to handle dynamic changes and unforeseen disturbances. Third, exploring new neural network architectures that may improve modeling accuracy and efficiency is promising. Additionally, investigating the model's application in real operational environments will be valuable for assessing its practical utility and reliability.

6. Conclusions

The research demonstrates the newly developed dynamic mathematical model (1)–(31) effectiveness for synthesizing controllers using genetic algorithms. The model offers significant advantages over traditional methods by integrating adaptive optimization with evolutionary principles, allowing for the broader parameter space dynamic exploration and parameters autonomous adjustment. This approach enhances flexibility, efficiency, and responsiveness to real-time changes.

The computer experiment results show that applying the model to a PID controller significantly improved performance. Adjustments to the PID coefficients led to a reduction in transient process time and overshoot, with modeling accuracy increasing from 99.523 to 99.783 % and losses minimized from 2.5 to 0.5 %. These improvements indicate the incorporating potential benefits the developed model into the automatic control system for helicopter TE free turbine rotor speed.

However, limitations include potential gaps in accounting for all external and internal influences, the neural network parameters specificity affecting generalizability, and the experimental data potential inadequacy in capturing complex dynamics. Future research should focus on validating the model's universality under diverse conditions, integrating it with advanced adaptive control algorithms, exploring new neural network architectures, and assessing its practical applicability in real operational environments.

Acknowledgements

The research was supported by the Ministry of Internal Affairs of Ukraine "Theoretical and applied aspects of the development of the aviation sphere" under Project No. 0123U104884. The research was carried out with the grant support of the National Research Fund of Ukraine "Methods and means of active and passive recognition of mines based on deep neural networks", project registration number 273/0024 from 1/08/2024 (2023.04/0024). Also, we would like to thank the reviewers for their precise and concise recommendations that improved the presentation of the results obtained.

References

[1] S. Li, S. Li, X. Yin, Synthesis of Non-blocking Controllers for Linear Temporal Logic Tasks under Partial Observations, IFAC-PapersOnLine 56:2 (2023) 11350–11356. doi: 10.1016/j.ifacol.2023.10.418.

[2] X. Yin, B. Gao, X. Yu, Formal synthesis of controllers for safety-critical autonomous systems: Developments and challenges, Annual Reviews in Control 57 (2024) 100940. doi: 10.1016/j.arcontrol.2024.100940.

[3] A. Didier, M. N. Zeilinger, Generalised Regret Optimal Controller Synthesis for Constrained Systems, IFAC-PapersOnLine 56:2 (2023) 2576–2582. doi: 10.1016/j.ifacol.2023.10.1341.

[4] L. Yuan, S. Chen, C. Zhang, G. Yang, Structured controller synthesis through block-diagonal factorization and parameter space optimization, Automatica 147 (2023) 110709. doi: 10.1016/j.automatica.2022.110709.

[5] X. Ban, H. Zhang, F. Wu, Advanced controller synthesis for fuzzy parameter varying systems, Journal of the Franklin Institute 359:2 (2022) 762–785. doi: 10.1016/j.jfranklin.2020.12.038.

[6] C. F. Verdier, M. Mazo Jr., Formal Controller Synthesis via Genetic Programming, IFAC-PapersOnLine 50:1 (2017) 7205–7210. doi: 10.1016/j.ifacol.2017.08.1362.

[7] S. B. Joseph, E. G. Dada, A. Abidemi, D. O. Oyewola, B. M. Khammas, Metaheuristic algorithms for PID controller parameters tuning: review, approaches and open problems, Heliyon 8:5 (2022) e09399. doi: 10.1016/j.heliyon.2022.e09399.

[8] L. Denisova, V. Meshcheryakov, Synthesis of a Control System Using the Genetic Algorithms, IFAC-PapersOnLine 49:12 (2016) 156–161. doi: 10.1016/j.ifacol.2016.07.567.

[9] N. Katal, S. Narayan, Automated synthesis of multivariate QFT controller and pre-filter for a distillation column with multiple time delays, Journal of Process Control 99 (2021) 79–106. doi: 10.1016/j.jprocont.2020.12.004.

[10] S. Vladov, R. Yakovliev, O. Hubachov, J. Rud, Neuro-Fuzzy System for Detection Fuel Consumption of Helicopters Turboshaft Engines, CEUR Workshop Proceedings 3628 (2023) 55–72. URL: https://ceur-ws.org/Vol-3628/paper5.pdf

[11] S. Vladov, Y. Shmelov, R. Yakovliev, Helicopters Aircraft Engines Self-Organizing Neural Network Automatic Control System, CEUR Workshop Proceedings 3137 (2022) 28–47. doi: 10.32782/cmis/3137-3.

[12] S. Vladov, Y. Shmelov, R. Yakovliev, M. Petchenko, Neural Network Method for Parametric Adaptation Helicopters Turboshaft Engines On-Board Automatic Control System Parameters, CEUR Workshop Proceedings 3403 (2023) 179–195. URL: https://ceur-ws.org/Vol-3403/paper15.pdf

[13] A. Mazouzi, N. Hadroug, W. Alayed, A. Hafaifa, A. Iratni, A. Kouzou, Comprehensive optimization of fuzzy logic-based energy management system for fuel-cell hybrid electric vehicle using genetic algorithm, International Journal of Hydrogen Energy 81 (2024) 889–905. doi: 10.1016/j.ijhydene.2024.07.237.

[14] P. Furda, M. Variny, J. Myšiak, Parallel Genetic Algorithm Interface II: A novel computational tool for accelerated simulation-based optimization, Chemical Engineering Research and Design 207 (2024) 29–48. doi: 10.1016/j.cherd.2024.05.030.

[15] J. W. Kok, E. Torta, M. A. Reniers, J. M. van de Mortel-Fronczak, M. J. G. van de Molengraft, Synthesis-Based Engineering of Supervisory Controllers for Autonomous Robotic Navigation, IFAC-PapersOnLine, 54:2 (2021) 259–264. doi: 10.1016/j.ifacol.2021.06.031.

[16] A. K. Cechinel, E. R. De Pieri, A. L. Fernandes Perez, P. D. Méa Plentz, Multi-robot Task Allocation Using Island Model Genetic Algorithm, IFAC-PapersOnLine 54:1 (2021) 558–563. doi: 10.1016/j.ifacol.2021.08.063.

[17] J. da S. Bohrer, M. Dorn, Enhancing classification with hybrid feature selection: A multi-objective genetic algorithm for high-dimensional data, Expert Systems with Applications 255 (2024) 124518. doi: 10.1016/j.eswa.2024.124518.

[18] E. Choi, S. Kwag, D. Hahm, Multihazard capacity optimization of an NPP using a multi-objective genetic algorithm and sampling-based PSA, Nuclear Engineering and Technology 56:2 (2024) 644–654. doi: 10.1016/j.net.2023.10.041.

[19] Y. Yang, Y. Ma, Y. Zhao, W. Zhang, Y. Wang, A dynamic multi-objective evolutionary algorithm based on genetic engineering and improved particle swarm prediction strategy, Information Sciences 660 (2024) 120125. doi: 10.1016/j.ins.2024.120125.

[20] Q. Ye, W. Wang, G. Li, Z. Wang, Dynamic-multi-task-assisted evolutionary algorithm for constrained multi-objective optimization, Swarm and Evolutionary Computation 90 (2024) 101683. doi: 10.1016/j.swevo.2024.101683.

[21] M. Anand, M. Zamani, Distributed Safety Controller Synthesis for Unknown Interconnected Systems via Graph Neural Networks, IFAC-PapersOnLine 58:11 (2024) 171–176. doi: 10.1016/j.ifacol.2024.07.443.

[22] M. AlShafeey, O. Rashdan, Genetic modification optimization technique: A neural network multi-objective energy management approach, Energy and AI 18 (2024) 100417. doi: 10.1016/j.egyai.2024.100417.

[23] S. López-Ruiz, C. I. Hernández-Castellanos, K. Rodríguez-Vázquez, Multi-objective optimization of neural network with stochastic directed search, Expert Systems with Applications 237 (2024) 121535. doi: 10.1016/j.eswa.2023.121535.

[24] Z. Liang, J. Yan, F. Zheng, J. Wang, L. Liu, Z. Zhu, Multi-objective multi-task particle swarm optimization based on objective space division and adaptive transfer, Expert Systems with Applications 255 (2024) 124618. doi: 10.1016/j.eswa.2024.124618.

[25] Z. Sun, Y. Liu, L. Tao, Attack localization task allocation in wireless sensor networks based on multi-objective binary particle swarm optimization, Journal of Network and Computer Applications 112 (2018) 29–40. doi: 10.1016/j.jnca.2018.03.023.

[26] K. Sun, J. Huo, H. Jia, Q. Liu, J. Yang, C. Cai, Nonlinear optimization of optical camera multiparameter via triple integrated Gradient-based optimizer algorithm, Optics & Laser Technology 179 (2024) 111294. doi: 10.1016/j.optlastec.2024.111294.

[27] F. Feng, S. Xiong, H. Kobayashi, Y. Zhou, M. Tanaka, A. Kawamoto, T. Nomura, B. Zhu, Nonlinear topology optimization on thin shells using a reduced-order elastic shell model, Thin-Walled Structures 197 (2024) 111566. doi: 10.1016/j.tws.2024.111566.

[28] K. Sanjay, R. Vijay Aravind, P. Balasubramaniam, Stabilization analysis for nonlinear interconnected system with memory based coupled sampled data control via quantum-inspired genetic algorithm, Information Sciences 676 (2024) 120857. doi: 10.1016/j.ins.2024.120857.

[29] S. Xu, W. Bi, A. Zhang, Z. Mao, Optimization of flight test tasks allocation and sequencing using genetic algorithm, Applied Soft Computing 115 (2022) 108241. doi: 10.1016/j.asoc.2021.108241.

[30] O. Rodríguez-Abreo, J. Rodríguez-Reséndiz, A. García-Cerezo, J. R. García-Martínez, Fuzzy logic controller for UAV with gains optimized via genetic algorithm, Heliyon 10:4 (2024) e26363. doi: 10.1016/j.heliyon.2024.e26363.

[31] H. Aygun, O. Turan, Application of genetic algorithm in exergy and sustainability: A case of aero-gas turbine engine at cruise phase, Energy 238 (2022) 121644. doi: 10.1016/j.energy.2021.121644.

[32] S. Vladov, Y. Shmelov, R. Yakovliev, Y. Stushchankyi, Y. Havryliuk, Neural Network Method for Controlling the Helicopters Turboshaft Engines Free Turbine Speed at Flight Modes, CEUR Workshop Proceedings 3426 (2023) 89–108. URL: https://ceur-ws.org/Vol-3426/paper8.pdf

[33] S. Vladov, R. Yakovliev, O. Hubachov, J. Rud, S. Drodova, A. Perekrest, Modified Discrete Neural Network PID Controller for Controlling the Helicopters Turboshaft Engines Free Turbine Speed. in: Proceedings of the first 2023 IEEE 5th International Conference on Modern Electrical and Energy System (MEES), Kremenchuk, Ukraine, September 27–30, 2023, pp. 797–802. doi: 10.1109/MEES61502.2023.10402433.
[34] M. Mahadi, T. Ballal, M. Moinuddin, U. M. Al-Saggaf, A Recursive Least-Squares with a Time-Varying Regularization Parameter, Applied Sciences 12:4 (2022) 2077. doi: 10.3390/app12042077.
[35] S. Vladov, R. Yakovliev, V. Vysotska, M. Nazarkevych, V. Lytvyn, The Method of Restoring Lost Information from Sensors Based on Auto-Associative Neural Networks, Applied System Innovation 7:3 (2024) 53. doi: 10.3390/asi7030053.
[36] S. Vladov, L. Scislo, V. Sokurenko, O. Muzychuk, V. Vysotska, A. Sachenko, A. Yurko, Helicopter Turboshaft Engines' Gas Generator Rotor R.P.M. Neuro-Fuzzy On-Board Controller Development, Energies 17:16 (2024) 4033. doi: 10.3390/en17164033.
[37] S. Vladov, R. Yakovliev, M. Bulakh, V. Vysotska, Neural Network Approximation of Helicopter Turboshaft Engine Parameters for Improved Efficiency, Energies 17:9 (2024) 2233. doi: 10.3390/en17092233.
[38] H.-Y. Kim, Statistical notes for clinical researchers: Chi-squared test and Fisher's exact test, Restorative Dentistry & Endodontics 42:2 (2017) 152.
[39] C. M. Stefanovic, A. G. Armada, X. Costa-Perez, Second Order Statistics of -Fisher-Snedecor Distribution and Their Application to Burst Error Rate Analysis of Multi-Hop Communications, IEEE Open Journal of the Communications Society 3 (2022) 2407–2424. doi: 10.1109/ojcoms.2022.3224835.
[40] S. Babichev, J. Krejci, J. Bicanek, V. Lytvynenko, Gene expression sequences clustering based on the internal and external clustering quality criteria. In Proceedings of the 2017 12th International Scientific and Technical Conference on Computer Sciences and Information Technologies (CSIT), Lviv, Ukraine, 05–08 September 2017.
[41] V. V. Morozov, O. V. Kalnichenko, O. O. Mezentseva, The method of interaction modeling on basis of deep learning the neural networks in complex IT-projects, International Journal of Computing 19:1 (2020) 88–96. doi: 10.47839/ijc.19.1.1697.
[42] Z. Hu, E. Kashyap, O.K. Tyshchenko, GEOCLUS: A Fuzzy-Based Learning Algorithm for Clustering Expression Datasets. Lecture Notes on Data Engineering and Communications Technologies 134 (2022) 337–349.
[43] A. R. Marakhimov, K. K. Khudaybergenov, Approach to the synthesis of neural network structure during classification, International Journal of Computing 19:1 (2020) 20–26. doi: 10.47839/ijc.19.1.1689.
[44] K. Andriushchenko, V. Rudyk, O. Riabchenko, M. Kachynska, N. Marynenko, L. Shergina, V. Kovtun, M. Tepliuk, A. Zhemba, O. Kuchai, Processes of managing information infrastructure of a digital enterprise in the framework of the «Industry 4.0» concept, Eastern-European Journal of Enterprise Technologies 1 (2019) 60–72.
[45] E. M. Cherrat, R. Alaoui, H. Bouzahir, Score fusion of finger vein and face for human recognition based on convolutional neural network model, International Journal of Computing 19:1 (2020) 11–19. doi: 10.47839/ijc.19.1.1688.

[46] Z. Hu, O. Shkurat, M. Kasner, Grayscale Image Colorization Method Based onU-Net Network, International Journal of Image, Graphics and Signal Processing 16:2 (2024) 70–82. doi: 10.5815/ijigsp.2024.02.06.

[47] Z. Hu, I. Dychka, K. Potapova, V. Meliukh, Augmenting Sentiment Analysis Prediction in Binary Text Classification through Advanced Natural Language Processing Models and Classifiers, International Journal of Information Technology and Computer Science 16:2 (2024) 16–31. doi: 10.5815/ijitcs.2024.02.02.

[48] O. Prokipchuk, V. Vysotska, P. Pukach, V. Lytvyn, D. Uhryn, Y. Ushenko, Z. Hu, Intelligent Analysis of Ukrainian-language Tweets for Public Opinion Research based on NLP Methods and Machine Learning Technology, International Journal of Modern Education and Computer Science 15:3 (2023) 70–93. doi: 10.5815/ijmecs.2023.03.06

[49] O. Drozd, G. Nowakowski, A. Sachenko, V. Antoniuk, V. Kochan, M. Drozd, Power-Oriented Monitoring of Clock Signals in FPGA Systems for Critical Application, Sensors 21:3 (2021) 792. doi: 10.3390/s21030792.

[50] V. Kovtun, T. Altameem, M. Al-Maitah, W. Kempa, Simple statistical tests selection based parallel computating method ensures the guaranteed global extremum identification, Journal of King Saud University – Science 36:5 (2024) 103165. doi: 10.1016/j.jksus.2024.103165.