On-board Hybrid Neural Network Classifier of Helicopters Turboshaft Engines Operational Status

Serhii Vladov [1], Ruslan Yakovliev [1], Oleksandr Hubachov [1] and Juliia Rud [1]

[1] *Kremenchuk Flight College of Kharkiv National University of Internal Affairs, Peremohy street 17/6, Kremenchuk, 39605, Ukraine*

Abstract

The work is devoted to the development of an on-board neural network classifier of helicopters turboshaft engines operational status. Proposed neural network classifier was developed based on an ensemble of neural networks, which consists of radial basis networks (RBF), perceptron, Kohonen network and hybrid neural network. The main task of the developed on-board neural network classifier is to determine defects in helicopters turboshaft engines units (air inlet section, compressor, combustion chamber, compressor turbine, free turbine, exhaust unit). It has been proven that to obtain the best result, it is advisable to use the following algorithms for training neural networks: a modified gradient algorithm for training an RBF network, a backpropagation method for training a multilayer perceptron, a competition method between neurons for training a Kohonen neural network, a hybrid algorithm – for training a hybrid neural network. The results of testing the developed on-board neural network classifier showed the ability to determine a compressor defect, a compressor turbine defect, and simultaneous compressor and compressor turbine defects. The effectiveness of the developed on-board neural network classifier for recognizing defects in helicopters turboshaft engines has been proven. A comparative assessment of the effectiveness of the developed on-board neural network classifier and existing methods for parametric diagnostics of the operational status of complex dynamic objects was carried out. The results of the studies showed that the developed on-board neural network classifier can identify defects in helicopters turboshaft engines components with an accuracy of up to 99.8%.

Keywords

Helicopters turboshaft engines, neural network classifier, RBF network, multilayer perceptron, Kohonen neural network, hybrid neural network, thermogas-dynamic parameters, training, operational status, diagnostics, defects

1. Introduction

During the operation of aviation equipment such as aircraft and helicopters, a primary objective is to diagnose the parameters of their aircraft engines. The total count of monitored (diagnosed) parameters can extend to 500 or more. The existing methods and techniques for diagnosing gas turbine engines (GTE) demand substantial enhancements, particularly as new generations of aircraft GTE necessitate advanced intelligent computer diagnostic technologies founded on the principles of expert systems (ES), neural networks (NN), fuzzy logic (FL), and genetic algorithms (GA). These technologies should be capable of incorporating the accumulated experience from prior work in this domain and devising (generalizing) innovative methods and techniques for further exploration. This imperative applies equally to aircraft gas turbine engines with a free turbine, known as turboshaft engines (TE),

Proceedings ITTAP'2023: 3rd International Workshop on Information Technologies: Theoretical and Applied Problems, November 22–24, 2023, Ternopil, Ukraine, Opole, Poland

EMAIL: ser2610196@gmail.com (S. Vladov); director.klk.hnuvs@gmail.com (R. Yakovliev); oleksandrgubachov@gmail.com (O. Hubachov); juliarud25@gmail.com (J. Rud)

ORCID: 0000-0001-8009-5254 (S. Vladov); 0000-0002-3788-2583 (R. Yakovliev); 0000-0002-1826-259X (O. Hubachov); 0000-0002-0328-5895 (J. Rud)

which form integral components of helicopter power plants. Among the intricate tasks that significantly enhance the efficiency of GTE diagnostics and elements of automatic control systems (ACS), there is a need to address several issues aimed at overcoming obstacles in identifying the operational status of GTE. These challenges are interconnected [1, 2]:

– in the event of primary information sensor malfunctions, there is a risk of generating false alarms within the gas turbine engine (GTE) control system, leading to a considerable decline in failure identification reliability. To ensure the accurate operation of the GTE operational status monitoring system, it becomes essential to distinguish (classify) deviations stemming from alterations in power plant characteristics from deviations in measured parameters linked to sensor malfunctions. In essence, this involves concurrently identifying the engine status, the parameters of its gas flow duct, and the measurement system, all while simultaneously identifying the program regulation;

– complications arise in distinguishing failures of engine components and sensor malfunctions during engine failures, including its subsystems. This challenge is particularly evident when facing minor deviations in gas-dynamic parameters, such as those occurring when individual turbine blades burn out, which are comparable to random errors in the measuring channels;

– challenges arise in the automatic acquisition and extraction of essential, reliable information during each flight, encompassing both steady and transient operating modes. This involves obtaining independent measurements following each engine transition to a new steady-state mode and ensuring a distinct separation between transient and steady modes. Addressing these challenges is imperative for enhancing the reliability of monitoring and diagnosing the operational status of the engine and the elements of the ACS during flight, especially when dealing with slight deviations from the anticipated standard of the measured parameters.

In such circumstances, the application of neural network technologies holds significant promise. A review of research in the domain of gas turbine engine operational status diagnostics utilizing neural networks [3, 4] indicates that ongoing work is underway. However, due to various reasons such as secrecy and the narrow specialization of the tasks at hand, most publications lack engineering methodologies, as well as theoretical and practical guidance for addressing such issues.

Therefore, the objective of this study is to formulate methods and techniques for the comprehensive diagnostics of helicopters TE operational status during flight modes utilizing neural network technologies.

2. Related works

A novel and promising domain in the realm of automatic control for intricate dynamic systems, operational status diagnostics, and predictive tasks involves the utilization of intelligent control systems founded on artificial neural networks [5, 6].

However, prevailing approaches to employing intelligent diagnostic methods are constrained by the specificity of the tasks, the underdeveloped theory pertaining to the use of neural networks in gas turbine engine (GTE) diagnostics, the absence of universal and formalized approaches, and the inherent imperfections within neural network methods themselves [7, 8].

Investigations into the creation of automated systems for diagnosing the operational status of complex dynamic objects [9, 10], including aircraft GTE [11, 12], reveal the inadequacy of relying solely on one of the known diagnostic methods. This is because none of these methods is universally applicable and entirely reliable. Consequently, monitoring and diagnostics systems built on the foundation of a single classifier will likely fall short of meeting the escalating requirements for gas turbine engine diagnosis. To enhance the efficiency of on-board technologies for GTE operational status diagnostics, several directions are identified. The primary focus should be on intellectualizing information processing processes through the application of neural network methods [13, 14], which can enhance the quality of on-board algorithms for GTE operational status diagnostics.

In this context, the development of neural network methods for diagnosing the operational status and identifying potential defects in helicopters' turboshaft engine units during flight modes remains pertinent.

3. Proposed technique

3.1. Problem statement

Let an N-dimensional feature space be given, each point of which can be represented by an N-dimensional vector $X = X_1, \ldots, X_N$. Let's divide this space into Q regions, which correspond to one class or another. Let the training set $\{X, B\} = (X_1, B_1), (X_2, B_2), \ldots, (X_N, B_N)$ be given, where X_i – point in the feature space, B_i – label of the class to which this point belongs. The job of the classifier is to indicate for each new point X that is not included in the training sample, under conditions of partial or complete uncertainty, which class this point belongs to, using the training sample $\{X, B\}$ for this. Let the state of a complex dynamic object (CDO) (helicopter TE) X at each discrete time t be described by an N-dimensional vector $X^t = \left(X_1^t, X_2^t, \ldots, X_N^t\right)$ of variables satisfying N equations $X^{t+1} = F\left(X^t, Q^t\right)$, $k = 1, 2, \ldots, M$, where $Q^t = \left(q_1^t, q_2^t, \ldots, q_m^t\right)$ – reference (defect-free) vector the state of the CDO (helicopter aircraft TE). Changes in CDO operational status at any time can be described by the equation $Y^t = H \cdot X^t$, where $Y^t = \left(y_1^t, y_2^t, \ldots, y_R^t\right)$ – operational status vector of real output parameters, H – transformation matrix. It is required to determine the diagnostic state vector of CDO (helicopter TE), minimizing the root-mean-square error between the reference (desired) Y and the real Y_R^t outputs.

The organization of complex diagnostics for helicopters TE is notably intricate due to the need for objective and comprehensive information about their operational status. This requires incorporating a substantial number of diverse physical quantities (parameters) into the diagnostic procedure to capture the behavior of various subsystems. The transition of a gas turbine engine from one state to another is marked by discernible changes in the controlled and diagnosed parameters. Fig. 1 provides a geometric representation of complex diagnostics for helicopters' turboshaft engines, which can be articulated as follows: based on a limited number of measurements of the object being diagnosed, an optimal decision must be made regarding its classification into one of several classes, specifically: S_1 – the area of serviceable states; S_2 – the area of critical states; S_3 – the area of faulty states; P_1, P_2, P_3 – classes (areas) of uncertain states.

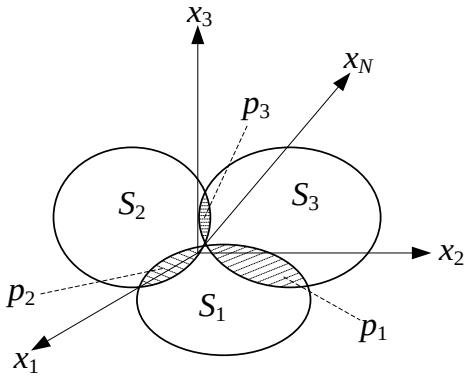


Figure 1: Geometrical interpretation of helicopters TE complex diagnostics problem

Geometrically, the operational status of helicopters TE can be visualized as an N-dimensional vector (X_N) (fig. 1), where the spatial coordinates represent N input parameters of the engine ($X_1, X_2, \ldots, X_N$). The position of this state vector in space corresponds to a specific level of engine performance, and the establishment of standards involves creating separating hypersurfaces within this space. These hypersurfaces act as boundaries between different classes, determined by a decision rule that dictates their construction. Consequently, decision-making revolves around assigning the diagnosed object to a specific class. The input parameters $X_1, X_2, \ldots, X_N$ refer to the thermogas-dynamic parameters of the working process of helicopters TE (Table 1). These parameters are either recorded on board the helicopter or computed using a comprehensive mathematical model of an aircraft engine with a free turbine [15]. The transition from the physical parameters of the engine to the specified values (and vice versa) follows a developed methodology [16, 17].

Table 1
Fragment of the training sample

Engine assembly	Parameter	Determination
Air inlet section	total air pressure behind air inlet section, P_{IN}	calculated analytically according to [15]
	total air temperature behind air inlet section, T_{IN}	calculated analytically according to [15]
Compressor	gas generator rotor r.p.m., n_{TC}	registered on board the helicopter
	air flow through the compressor, G_{air}	calculated analytically according to [15]
	air pressure behind the compressor, P_C	calculated analytically according to [15]
	air temperature behind the compressor, T_C	calculated analytically according to [15]
Combustion chamber	total gas pressure behind the combustion chamber, P_G	calculated analytically according to [15]
	gas temperature in front of the compressor turbine, T_G	registered on board the helicopter
	fuel consumption, G_T	calculated analytically according to [15]
Compressor turbine	total gas pressure behind the compressor turbine, P_{TC}	calculated analytically according to [15]
	gas temperature behind the compressor turbine, T_{TC}	calculated analytically according to [15]
Free turbine	total gas pressure behind the free turbine, P_{FT}	calculated analytically according to [15]
	gas temperature behind the free turbine, T_{FT}	calculated analytically according to [15]
	free turbine rotor speed, n_{FT}	registered on board the helicopter
Exhaust unit	total gas pressure behind the exhaust unit, P_{OUT}	calculated analytically according to [15]
	gas temperature behind the exhaust unit, T_{OUT}	calculated analytically according to [15]

The output diagnostic parameters of helicopters TE are: degree of increase in the total pressure in the compressor π_C^*, compressor efficiency η_C, mechanical compressor efficiency η_{MC}, recovery factor of the total gas pressure in the combustion chamber σ_{CC}, combustion chamber cross-sectional area F_{CC}, compressor turbine efficiency η_{TC}, compressor turbine operation A_{TC}, degree of reduction of the total gas pressure in the compressor turbine π_{TC}^*, total pressure reduction ratio in the free turbine π_{FT}^*, power efficiency of a free turbine $\eta_{\Sigma FT}$, total pressure reduction ratio in the free turbine and exhaust unit $\pi_{\Sigma FT}^*$, total gas pressure recovery factor in the exhaust unit σ_{EU}.

3.2. Neural network classifier development

In addressing the intricate challenges of diagnosing helicopters' turboshaft engines, hybrid ensembles of neural networks [18] can serve effectively as dynamic repositories of expert knowledge. In comparison with traditional neural networks, these ensembles offer additional practical advantages,

including: decomposition of complex dynamic objects (CDO) into simpler entities or subsystems; enhanced adaptability to changing external conditions, positioning them within the class of adaptive and self-adjusting systems; optimization of the neural ensemble's structure to cater to specific diagnostic tasks; superior speed and accuracy compared to classical fully connected networks; improved approximation of piecewise continuous functions by the neural ensemble [19, 20].

To adapt the recognition of defects by a neural ensemble, as part of the training sample, we distinguish five generalized classes of engine status (table 2): S_0 – serviceable (reference) status corresponding to the vector R = [0; 0; 0]; S_1 – compressor defect corresponding to the vector R = [0; 1; 0]; S_2 – combustion chamber defect corresponding to the vector R = [0; 1; 1]; S_3 – compressor turbine defect corresponding to the vector R = [1; 0; 0]; S_4 – free turbine defect corresponding to the vector R = [1; 1; 0]; S_5 – exhaust unit defect corresponding to the vector R = [1; 0; 1].

Table 2

Diagnostic model input parameters

Status number	Neural ensemble binary outputs			Defect localization
1	0	0	0	Serviceable (reference) status (S_0)
2	0	1	0	Compressor defect (S_1)
3	0	1	1	Combustion chamber defect (S_2)
4	1	0	0	Compressor turbine defect (S_3)
5	1	1	0	Free turbine defect (S_4)
6	1	0	1	Output device defect (S_5)

To address the issue at hand, a neural network classifier was devised (fig. 2), utilizing a composite ensemble of neural networks. This ensemble incorporates radial basis networks (RBF), perceptron, Kohonen network, and a hybrid neural network.

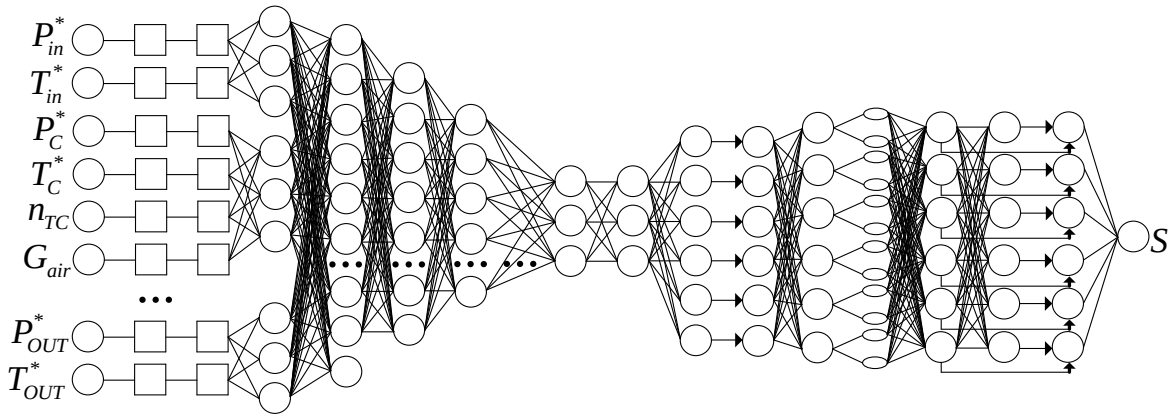


Figure 2: Structure of the developed neural network classifier

Opting for the radial basis function (RBF) architecture as the recognition neural network, as opposed to the perceptron neural network, is more advantageous. This preference stems from the fact that the determination of weight coefficients in the RBF network is accomplished more swiftly and accurately compared to adjusting the parameters of the perceptron. This efficiency arises because the utilization of gradient methods for parameter adjustment in the perceptron often leads to the attainment of local minima.

3.3. RBF network algorithm training

The structure of the RBF network entails a two-layer design, where the first layer executes a predetermined non-linear transformation without engaging in parameter tuning. This process maps the input space to a new space. In this context, considering the 16 input parameters of the helicopters TE (table 1), an optimal configuration, in terms of parameter decomposition, would involve six RBF architecture neural networks. These networks would align with the number of parameters at the input (state vector) based on the engine node (ranging from 2 to 4, depending on the node), and three output

parameters in line with the binary classification of statuses (table 2). The training algorithm for the RBF neural network employs the modified gradient training of RBF networks, as developed in [16], with the block diagram illustrated in fig. 3, where n – number of parts in the first layer; x_1, x_2, ..., x_n – input signals; m – number of elements in the second layer; c_{i1}, c_{i2}, ..., c_{in} – coordinates of the center of the i-th element; σ_i – width of the radial function of the i-th element; θ_i – output signal of the i-th element; w_i – weight coefficient of the initial connection of the i-th element; y – output signal of the RBF neural network.

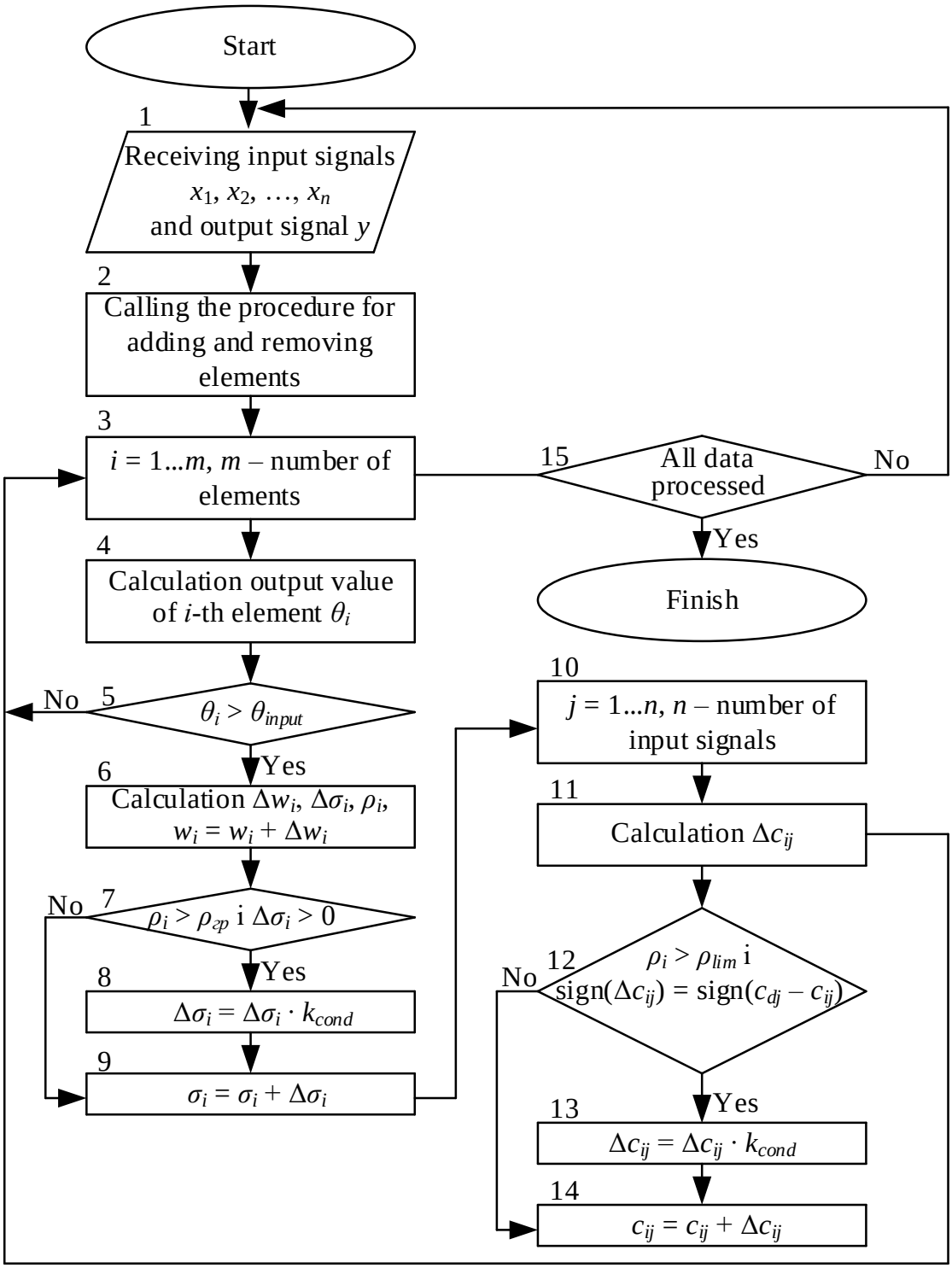


Figure 3: Diagram illustrating the gradient training algorithm for an enhanced radial basis function neural network [16]

The Gaussian function governs the output signal of each component within the radial basis function neural network [16, 21]:

$$\theta_i = e^{-\frac{\sum_{j=1}^{n}\left(x_j - c_{ij}\right)^2}{2\cdot\sigma_i^2}}. \quad (1)$$

The output signal of the radial basis function neural network is computed by aggregating the signals of its elements through a weighted sum:

$$y = \sum_{i=1}^{m} w_i \cdot \theta_i. \quad (2)$$

A gradient-based algorithm is employed to train the radial basis function (RBF) neural network by minimizing the network error objective function. This algorithm computes the adjustments for each element, including changes in the weight factor (w_i), the element width ($\Delta\sigma_i$), and the element center coordinates (c_{ij}).

Through experiments, certain drawbacks of the conventional gradient algorithm for RBF neural network training were identified:

1. The RBF neural network training algorithm lacks specific guidelines for the initial assignment of network elements and their parameters, as well as rules for adjusting the element count during training. The uniform distribution of elements across the working area may not always be optimal, and there could be instances where the initially specified number of elements proves insufficient for achieving the required training quality.

2. Throughout the training process, the parameters of all network elements undergo changes, resulting in an escalation of computational costs as the element count increases.

3. The RBF network faces challenges in reaching a stable state during training when elements with closely positioned center coordinates (c_{ij}) and radial function widths (σ_i) exist. The occurrence of such situations is highly dependent on the chosen number of elements and their initial parameters. The degradation of training quality stems from the assumption in the gradient algorithm that the output value of the RBF neural network at any given point in the working area is primarily influenced by only one element. When multiple elements are present in a specific region of the working area, adjusting their parameters according to the gradient algorithm does not consistently lead to a reduction in training error.

To identify instances where the parameters of certain elements converge, the notion of the mutual intersection coefficient of elements has been introduced. To compute this coefficient for a specific element within the RBF network, it is essential to identify a second element whose center is in closer proximity to the center of the analyzed element. The mutual intersection coefficient is determined by summing the initial value of the current element at the center of the second element and the initial value of the second element at the center of the current element:

$$\rho_i = e^{-\frac{\sum_{j=1}^{n}\left(c_{ij} - c_{dj}\right)^2}{2\cdot\sigma_i^2}} + e^{-\frac{\sum_{j=1}^{n}\left(c_{ij} - c_{dj}\right)^2}{2\cdot\sigma_d^2}}; \quad (3)$$

where i – number of the element for which the value of the coefficient of mutual intersection is calculated; d – element number, the center of which is located closer to the center of the element with number i, which is determined according to the expression:

$$d = \arg\min_k \sqrt{\sum_{j=1}^{n}\left(c_{ij} - c_{kj}\right)^2}. \quad (4)$$

The coefficient of mutual intersection resides within the range (0; 2). It achieves its maximum value when the centers of the examined elements coincide. As the coefficient of mutual intersection surpasses 1.95, it becomes imperative to cap the maximum value at 1.95 to attain optimal training quality for the RBF network.

To address the deficiencies of the conventional gradient algorithm in training the RBF network for helicopter TE identification tasks, this study employs a modified gradient algorithm [16, 22], as illustrated in Fig. 3. Blocks absent in the classic algorithm are denoted by asterisks. The key distinctions from the classical algorithm are outlined as follows:

1. Introducing regulations for altering the RBF network structure during training (block 2). At the onset of neural network training, the RBF network is devoid of elements. New elements are introduced as necessary, and unused elements are eliminated.

2. A reduction in the computational costs associated with each training cycle is achieved by modifying the parameters not for all elements, as in the classic algorithm, but exclusively for elements whose initial value at the specified point surpasses the value of θ_{zm} (blocks 4 and 5).

3. The likelihood of a scenario where the parameters of certain elements closely align is mitigated. To achieve this, the calculated values Δc_{ij} and $\Delta\sigma_i$ are diminished if the coefficient of mutual intersection among elements surpasses the threshold value ρ_{gr}, set at 1.95 (blocks 7, 8, 12, 13).

Altering the configuration of the RBF network through the addition or removal of elements results in a modification of the RBF network's output value solely in the proximity of the added or removed element's center. This effect is localized and does not impact the entire working area, unlike the alteration of a multilayer perceptron's structure. Consequently, the addition and removal of elements in the RBF network can be executed during the training process without the necessity of restarting the training process from the beginning.

3.4. Perceptron neural network algorithm training

Within the hybrid ensemble, the perceptron neural network serves as a focal field, amalgamating the outputs of six RBF-architecture neural networks, each with three outputs. In the experimental phase of this neural network, following the contrasting process, its architecture manifested as a six-layer perceptron: the initial input layer contained 18 neurons, succeeded by a second hidden layer with 15 neurons, a third hidden layer with 12 neurons, a fourth hidden layer with 9 neurons, a fifth hidden layer with 6 neurons, and a final output layer comprising three neurons. The perceptron neural network contributes to the enhancement of defect recognition quality by "fine-tuning" its weight coefficients. General equations describing the operation of the perceptron: for the input layer ($k = 1$) – $U^1 = X$ (input vector); for the first hidden layer ($k = 2$) – $U_j^2 = f\sum_{i=1}^{n_2} w_{ij}^1 \cdot U_i^1$; … ; for the sixth output layer ($k = 6$) – $Y_j = U_j^6 = f\sum_{i=1}^{n_6} w_{ij}^6 \cdot U_i^6$. Table 3 shows a comparative analysis of the perceptron training results, on the basis of which the error backpropagation algorithm is selected as the perceptron training algorithm.

Table 3

Perceptron neural network training results

Training algorithm	Perceptron recognition error	Number of training steps
Back propagation	0.00011731	700
Conjugate gradient	0.00019935	800
Quick propagation	0.00018364	750
Quasi-Newton	0.00016997	800
Lewenberg-Marquardt	0.00014133	850
Delta bar delta	0.00015702	750

3.5. Kohonen neural network algorithm training

The Kohonen neural network, functioning as a classifier with three inputs and six outputs, exhibits a high level of precision in categorizing (identifying) the operational status of helicopter TE. This includes accommodating the partial or complete uncertainty of its parameters. Simultaneously, the Kohonen neural network serves the purpose of initial sorting and clustering of incoming values, contributing to the structuring of the initial data for the hybrid neural network. As per [23], the training foundation for the Kohonen neural network involves a competitive process among neurons. In this scenario, the Kohonen network features three inputs and six outputs (R_1...R_6, corresponding to the

number of generalized state classes). The weight coefficients of synaptic connections for each *i*-th neuron in the output layer of the Kohonen neural network collectively form a vector $w_i = \left(w_{i_1}; w_{i_2}; ...; w_{i_{10}} \right)^T$ at i = 1, 2, ..., n. When the Kohonen neural network is activated by the input vector ΔY, the neuron whose weights are the least different from the corresponding components of the input vector wins the competition, i.e., for the winning neuron w_p, the relation is holds:

$$d(\Delta Y, w_p) = \min d(\Delta Y, w_i),\ 1 \le i \le n. \quad (5)$$

In equation (5), $d(\Delta Y, w_i)$ represents the distance (as per the chosen metric) between the vectors ΔY and $w = (w_1, w_2, \ldots, w_n)^T$, where n – number of outputs in the neural network's output layer (in this instance, n = 6). The victorious neuron establishes a topological neighborhood $S_p(k)$ around itself, characterized by a specific energy. All neurons situated within this neighborhood undergo adaptation, wherein their vectors of weight coefficients undergo changes in the direction of the vector ΔY, following a prescribed rule:

$$w_i(k+1) = w_i(k) + \eta_i(k)\left(\Delta Y - w_i(k)\right); \quad (6)$$

where $\eta_i(k)$ – communication coefficient of the first neuron from the neighborhood $S_p(k)$ at the k-th moment of time. The training coefficient diminishes as the separation between the *i*-th neuron and the victor expands, and the weights of neurons beyond the confines of the $S_p(k)$ neighborhood remain unchanged.

The objective of training the Kohonen neural network through neuron competition is to arrange the neurons (determine the values of their weight coefficients) in a manner that minimizes the expected distortion value. This distortion is gauged by the approximation error of the input vector ΔY relative to the weight coefficients of the winning neuron. In the context of L input vectors $(\Delta Y)_j$, where j = 1, 2, ..., L, and utilizing the Euclidean metric, this error can be formulated as:

$$E = \frac{1}{L}\sum_{j=1}^{L}\left\| \left(\Delta Y_i\right)_j - w_p(j) \right\|^2; \quad (7)$$

where $w_p(j)$ – weighting coefficients of the winning neuron when the vector network is presented $(\Delta Y)_j$.

The results of the training process of the Kohonen neural network (after 700...800 training cycles) are presented in table 4.

Table 4

Kohonen neural network training results

Neural network outputs	Number of status (winning frequencies)	Operational status
R_1	6	Compressor defect
R_2	5	Combustion chamber defect
R_3	1	Compressor turbine defect
R_4	2	Free turbine defect
R_5	3	Output device defect
R_6	4	Serviceable condition

3.6. Hybrid neural network algorithm training

The hybrid neural network enables the assessment of the membership level of the input set of indicator values to a predetermined class, indicative of the operational status of helicopter TE.

As previously mentioned, the conclusive step in diagnostics involves determining the type of failure in helicopter TE based on the analysis of the numerical vector R. Consequently, the graphical representation of the task (see Fig. 1) undergoes modification to the format depicted in Fig. 4. In this representation, the vertices of the cube align with the centers of clusters (reference engine states) as outlined in table 2, that is, S_0 – center of the cluster (precedent) corresponding to the serviceable (reference engine status); S_1 – center of the cluster corresponding to a defect in the compressor; S_2 –

cluster center corresponding to a defect in the combustion chamber; S_3 – center of the cluster corresponding to a defect in the compressor turbine; S_4 – cluster center corresponding to a defect in a free turbine; S_5 – cluster center corresponding to a defect in the exhaust unit. The actual engine status vector S can take on a value at any point inside the given cube $S = (R_1, R_2, R_3)^T$, $0 \le R_i \le 1$.

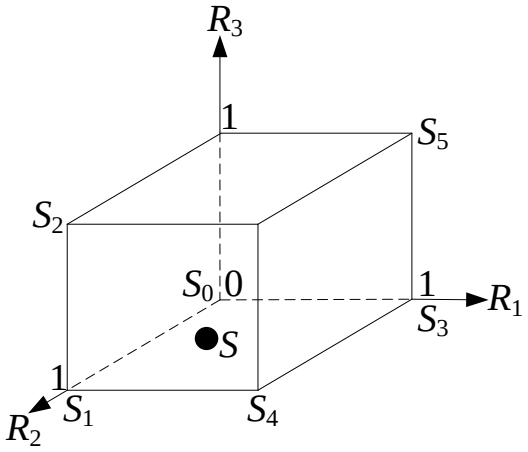


Figure 4: Modified graphical interpretation helicopters aircraft TE operational status diagnostics process

The assessment of the operational status of helicopter TE follows the "nearest neighbor" rule, wherein the engine is categorized into the class that includes its closest neighbor (or the majority of its closest neighbors). The decision rule, governing the decision-making process (diagnosis), is formulated as follows:

$$S \rightarrow S_p, \text{ if } d(S, S_p) \rightarrow \min; \tag{8}$$

where d – distance to the center of the nearest (p-th) cluster (precedent), which is calculated, for example, using the Euclidean metric.

The hybrid training algorithm devised for the hybrid neural network relies on the backpropagation algorithm as its foundation. Over a specified number of epochs, the neural network undergoes training employing a modified backpropagation algorithm. Simultaneously, the residual for neurons in the output layer is computed in a manner consistent with the backpropagation approach. The inconsistency of the hidden layers is derived from the inconsistencies across all variations of the preceding layer, as described in [16, 24]:

$$\sigma_i^{n-1} = f'\left(S_i^n\right) \cdot \sum_l \sum_k w_{l,k,i}^n \cdot \sigma_{l,k}^i. \tag{9}$$

Utilizing the residual outlined, akin to the backpropagation algorithm, the weights are incremented in the direction opposite to the gradient. However, in this instance, the gradient takes the form of a matrix rather than a vector. Consequently, the adjustments to the weights are made to minimize errors across all directions (fig. 5). The objective is to take steps that effectively diminish the disparities between the neural network outputs and the target values at various points.

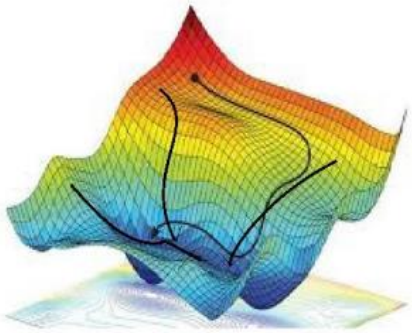


Figure 5: Diagram of weights change direction [16, 24]

The average accumulated error, which is calculated for each variant of the previous layer, is determined according to the expression:

$$\varepsilon_l = \sum_k \frac{\varepsilon_k}{n}; \tag{10}$$

where n – number of neurons in the previous layer. The offset increment is a random variable that takes any value from a series of residuals of the previous layer options with a probability equal to $1 - \varepsilon_l$, where ε_l – average accumulated error of the previous layer option, which is determined according to the expression:

$$x = \varepsilon(x);\ P_\varepsilon(B) = 1 - \varepsilon_l; \tag{11}$$

where x – offset increment.

When crossing over neurons during training steps using a genetic algorithm, the function is used as a fitness function $f(x) = 1 - \frac{\varepsilon}{m}$, where ε – accumulated error of the neuron of the output layer (in this case, after the training step using the genetic algorithm, it is reset and takes on a zero value), $m = o \cdot l$, where l – number of training sets, o – number of training epochs.

Block diagram of the developed hybrid training algorithm for hybrid neural networks is shown in fig. 6, information of the blocks of which is given in table 5.

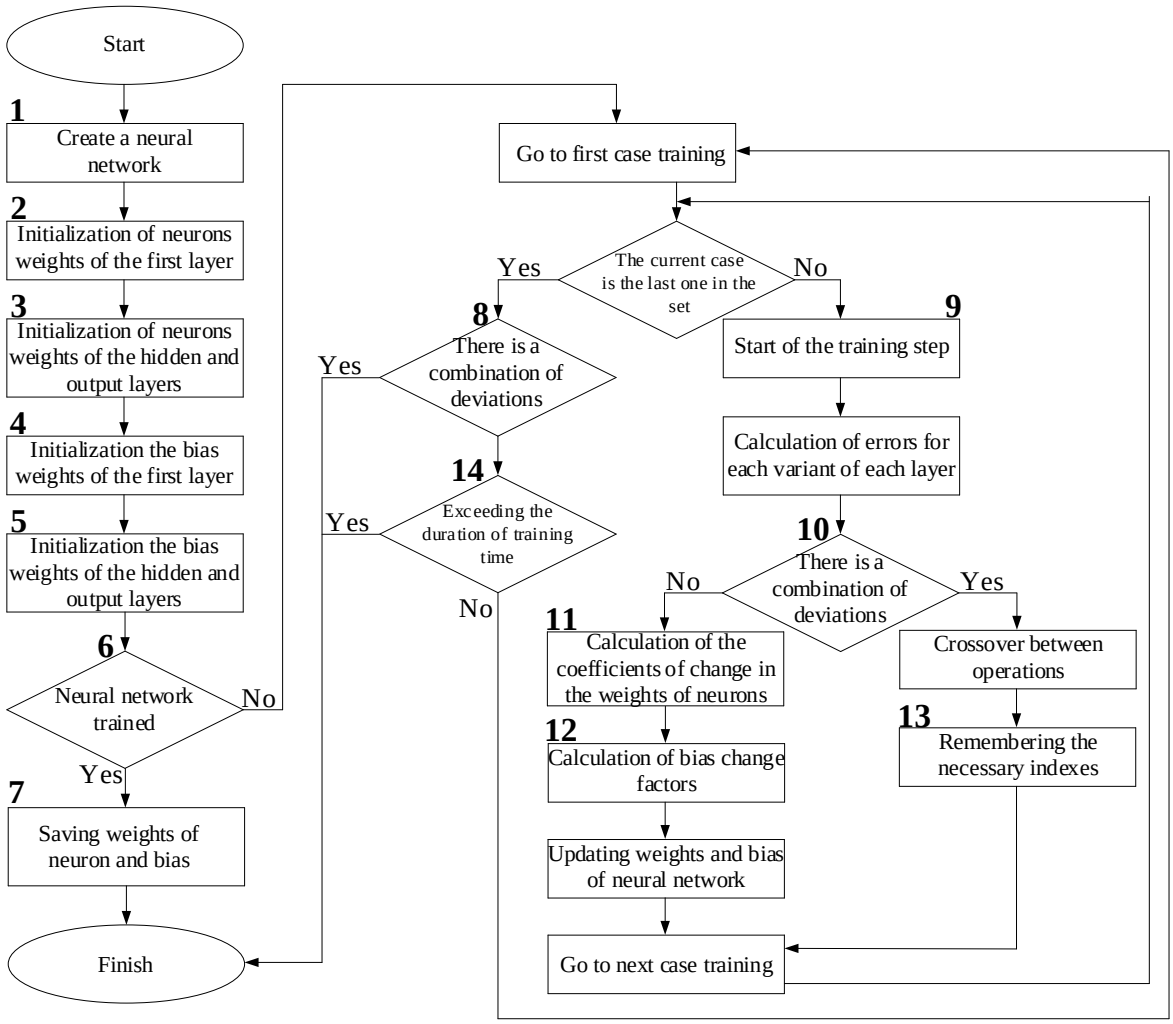


Figure 6: Block diagram of the developed hybrid training algorithm for hybrid neural networks [16, 24]

Table 5

Block comments of the developed hybrid training algorithm for hybrid neural networks [16, 24]

Block number	Comment
1	neural network is created with one input layer option and several inner and output layer options
2	initialization of neurons weights of the first layer is performed by equating them to 1
3	initialization of neurons weights of the hidden and output layers is carried out by random small values from 0 to 1, and you need to use different ranges of weights for different versions of the layers (for example, all the weights of the first version of the layer should be made in the range 0 ... 0.5, and the second version of the layer – in the range of 0.6…0.9
4	initialization of the shift weights of the first layer is carried out by equating them to 0
5	initialization of the shift weights of the hidden and output layers is carried out by random small values from 0 to 1, and you need to use different ranges of weights for different layer options (for example, all the weights of the first layer option should be in the range 0 ... 0.5, and the second layer option – in the range of 0.6…0.9
6	neural network is considered trained if there is at least one version of the original and hidden layers, in which the errors of the output layer do not exceed the maximum error
7	saving neuron weights and biasing the variants of the original and hidden layers, the deviations of the initial values of which do not exceed the maximum error for all training sets is the final solution to the neural network training problem
8	there is at least one combination of the hidden and source layer, the deviation of the source values from the target values of which does not exceed the maximum error for all training examples
9	feed a training example to the input of the neural network and get the initial values of all layers for each variant of the hidden and original layers. As a result, the output will be a multidimensional array containing the outputs of the first layer, each hidden layer option, and each set of hidden and original layer options
10	there is at least one version of the original layer, the deviation from the target values for which does not exceed the maximum error
11	calculation of the change in the coefficients of change in the weights of neurons is performed for each input, hidden and output layers
12	calculation of the change in the shift change coefficients is performed for each input, hidden and output layers
13	remember the indexes of the input and output layers with deviations not exceeding the maximum error
14	time elapsed since the start of training is greater than the maximum value of training time

4. Experiment

The examination and initial processing of the input data were conducted by the authors group and comprehensively detailed in [25, 26]. The input parameters for the mathematical model of helicopter TE encompass atmospheric values (h – flight altitude, T_N – temperature, P_N – pressure, ρ – air density). The parameters recorded aboard the helicopter (n_{TC} – gas generator rotor r.p.m., n_{FT} – free turbine rotor speed, T_G – gas temperature in front of the compressor turbine) were adjusted to absolute values according to the gas-dynamic similarity theory developed by Professor Valery Avgustinovich (refer to

table 6). In this study, we posit the constancy of atmospheric parameters (h – flight altitude, T_N – temperature, P_N – pressure, ρ – air density) [25, 26].

Table 6

Part of training set (on the example of TV3-117 TE [25, 26])

Number	T_G	n_{TC}	n_{FT}
1	0.932	0.929	0.943
2	0.964	0.933	0.982
3	0.917	0.952	0.962
4	0.908	0.988	0.987
5	0.899	0.991	0.972
6	0.915	0.997	0.963
7	0.922	0.968	0.962
8	0.989	0.962	0.969
9	0.954	0.954	0.947
10	0.977	0.961	0.953
…	…	…	…
256	0.953	0.973	0.981

The assessment of the homogeneity of the training and test samples is a crucial consideration. In this regard, the Fisher-Pearson criterion χ^2 with degrees of freedom $r - k - 1$ is employed [25, 26]:

$$\chi^2 = \min_{\theta} \sum_{i=1}^{r} \left(\frac{m_i - np_i(\theta)}{np_i(\theta)} \right); \tag{12}$$

where θ – maximum likelihood estimate found from the frequencies $m_1, \ldots, m_r$; n – number of elements in the sample; $p_i(\theta)$ – signifies the probabilities of elementary outcomes up to a certain indeterminate k-dimensional parameter θ.

The conclusive step in statistical data processing involves their normalization, a procedure that can be carried out using the following expression:

$$y_i = \frac{y_i - y_{i\min}}{y_{i\max} - y_{i\min}}; \tag{13}$$

where y_i – dimensionless quantity in the range [0; 1]; $y_{i\min}$ and $y_{i\max}$ – minimum and maximum values of the y_i variable.

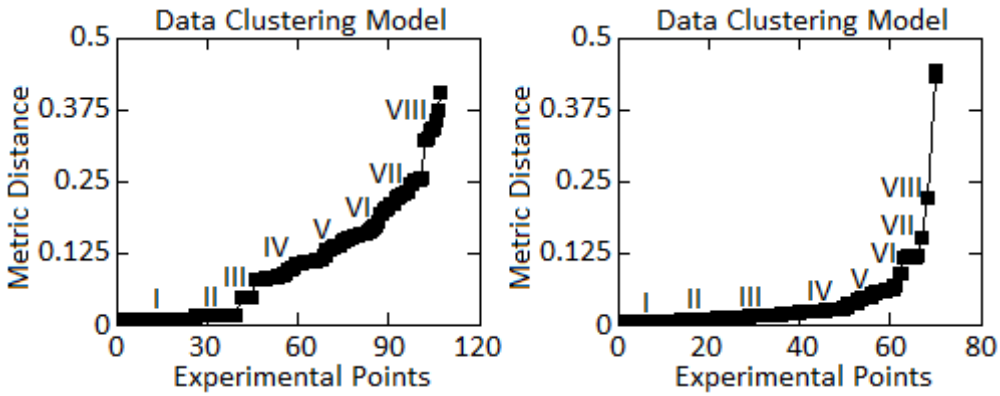
The previously mentioned χ^2 statistics, under the stated assumptions, enables the testing of the hypothesis regarding the representativeness of sample variances and the covariance of factors within the statistical model. The realm of accepting the hypothesis is denoted by $\chi^2 \le \chi_{n-m,\alpha}$, with α representing the significance level of the criterion. The outcomes of the computations based on equation (12) are provided in table 7 [25, 26].

Table 7

Part of the training sample during the operation of helicopters TE (on the example of TV3-117 TE) (author's development, described in [25, 26])

Number	$P(T_G)$	$P(n_{TC})$	$P(n_{FT})$
1	0.561	0.109	0.652
2	0.588	0.155	0.574
3	0.542	0.128	0.515
4	0.612	0.147	0.655
5	0.644	0.121	0.612
…	…	…	…
256	0.537	0.098	0.651

To verify the representativeness of the training and test samples, an initial data cluster analysis was conducted (refer to table 7), resulting in the identification of eight classes (fig. 7, *a*). Subsequent to the randomization procedure, the effective training (control) and test samples were chosen in a 2:1 ratio, specifically 67 % and 33 %, respectively. The clustering process applied to the training (fig. 7, *b*) and test samples reveals that, akin to the original sample, each of them comprises eight classes. The distances between the clusters align nearly perfectly across all examined samples, indicating the representativeness of both the training and test samples [25, 26].



a b

Figure 7: Clustering results: a – initial experimental sample (I…VIII – classes); b – training sample (author's development, described in [25, 26])

To form the training and test subsets in the work, cross-validation was used [27] to estimate the values of the parameters of TV3-117 engine, the results obtained in [27] of which are shown in fig. 8.

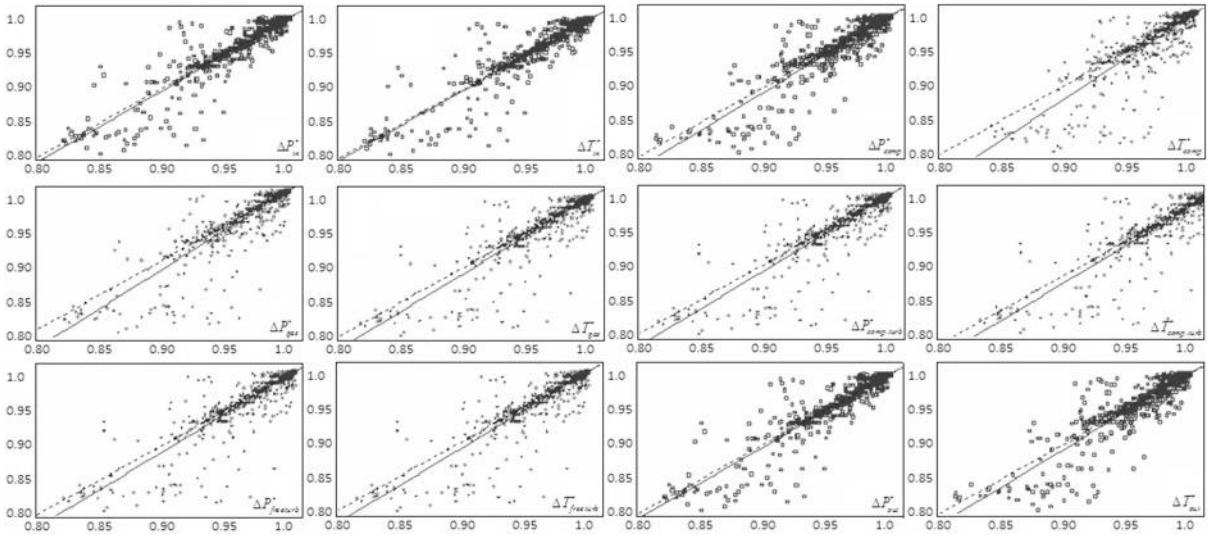


Figure 8: TV3-117 aircraft engine input parameter scatter diagram [27]

5. Results

The outcomes of assessing the unknown condition of helicopter TE, demonstrated through the TV3-117 aircraft engine (integral to the power plant of the Mi-8MTV helicopter), using the developed neural network classifier, are outlined in table 6. Similar to the earlier discussed examples, the performance of the developed neural network classifier was examined on a test sample (the third and fourth rows of table 6) in noise-free settings (the first and third rows of table 6), as well as under the influence of the additive component of white noise ($\sigma = 0.01$; $M = 0$), considering a 1% and 3% alteration in compressor

efficiency (second and fourth rows of table 8). The final line (table 8) corresponds to a dual defect, involving simultaneous changes in the efficiency of both the compressor and compressor turbine.

Table 8

Test results of the developed neural network classifier

Line number	Classifier output values						Operational status
	R_1	R_2	R_3	R_4	R_5	R_6	
1	0.609	0.084	0.009	0.011	0.009	0.007	Compressor defect
2 (with noice)	0.717	0.143	0.204	0.135	0.252	0.004	(–1 % η_C)
3	0.668	0.121	0.233	0.032	0.177	0.005	Compressor defect
4 (with noice)	0.745	0.308	0.362	0.180	0.316	0.002	(–3 % η_C)
5	0.732	0.711	0.009	0.018	0.016	0.001	Compressor and compressor turbine defect (–3 % η_C and (–3 % η_{TC})

The examination of the test outcomes from the developed neural network classifier reveals that the prevailing neuron (for rows 1…4) is the one with the output R_1, signifying a compressor defect, as indicated in table 4. As previously, the severity of the defect can be inferred from the numerical value at each neural network output. In the fifth row (table 8), there are two prevailing neurons, with the maximum signal values observed for outputs R_1 and R_2. According to table 6, the developed neural network classifier successfully identifies the mentioned dual defect involving a reduction in the efficiency of both the compressor and compressor turbine.

Let us consider this procedure using the example of the problem of diagnosing a double defect associated with a decrease in the efficiency of the compressor and compressor turbine by 3 %. The value of the vector at the output of the neural network classifier in this case is $R = (0.732; 0.711; 0.009; 0.018; 0.016; 0.001)^T$ (table 6) of the corresponding precedent to the centers of clusters: $d(S, S_0) = 0.653$; $d(S, S_1) = 0.724$; $d(S, S_2) = 0.631$; $d(S, S_3) = 1.168$; $d(S, S_4) = 1.241$; $d(S, S_5) = 1.326$.

The effectiveness of defect recognition was assessed by the total "discrepancy" based on the parameters of the state of engine components [28]:

$$\Delta\Sigma = \sum_{i=1}^{6} \delta x_i = |\delta x_1| + |\delta x_2| + ... + |\delta x_6|; \quad (14)$$

where $\delta x_i = x_i^* - x_i$; δx_1 – operational status deviation (defect size) of the air inlet section; δx_2 – operational status deviation (defect size) of the compressor unit; δx_3 – operational status deviation (defect size) of the combustion chamber unit; δx_4 – operational status deviation (defect size) of the compressor turbine unit; δx_5 – operational status deviation (defect size) of the free turbine unit; δx_6 – operational status deviation (defect size) of the exhaust unit (see table 1).

Fig. 9 shows the results of the efficiency of the neural network classifier for recognizing defects in helicopter TE (using the example of the TV3-117 engine): G_T – fuel consumption; T_C – air temperature behind the compressor (calculated using [15]); T_G – gas temperature in front of the compressor turbine; n_{TC}, n_{FT} – gas generator rotor r.p.m. and free turbine rotors, respectively; T_{TC}, T_{FT} – gas temperature behind the compressor turbine and free turbine, respectively (calculated using [15]).

Wherein:

fig. 9, a: curve 1 – G_T; curve 2 – T_C; curve 3 – T_G;

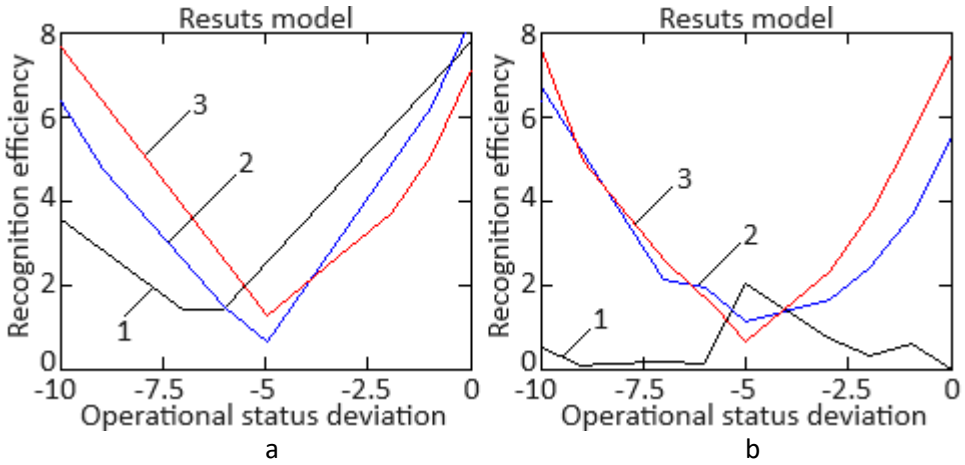
fig. 9, b: curve 1 – T_C, T_G; curve 2 – G_T, T_C; curve 3 – G_T, T_G;

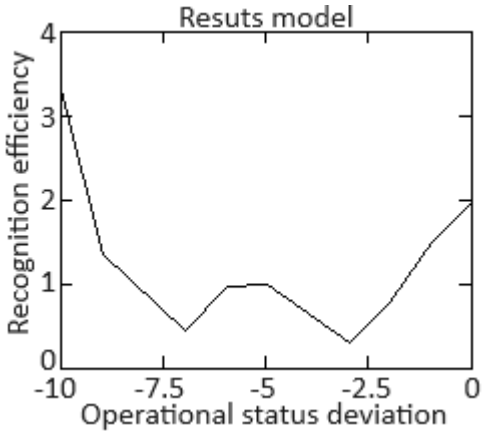
fig. 9, c: curve 1 – G_T, T_C, T_G;

fig. 9, d: curve 1 – T_{TC}; curve 2 – G_T; curve 3 – n_{FT}; curve 4 – T_{FT}; curve 5 – n_{TC};

fig. 9, e: curve 1 – G_T, n_{TC}, n_{FT}, T_{TC}; curve 2 – G_T, n_{TC}, T_{TC}, T_{FT}; curve 3 – G_T, n_{FT}, T_{TC}, T_{FT}; curve 4 – G_T, n_{TC}, n_{FT}, T_{FT}; curve 5 – n_{TC}, n_{FT}, T_{TC}, T_{FT};

fig. 9, f: curve 1 – G_T, n_{TC}, n_{FT}, T_{TC}, T_{FT}.





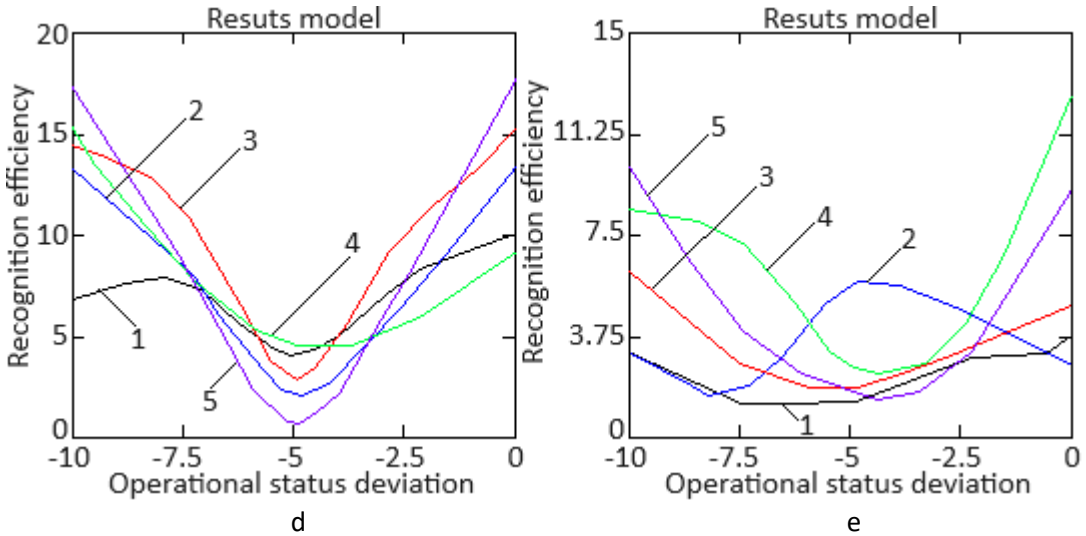
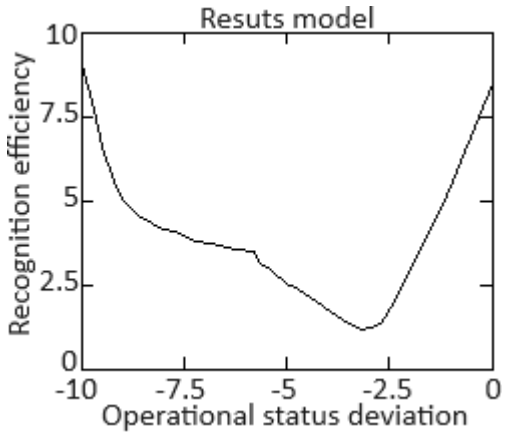


Figure 9: Results of the effectiveness of the neural network classifier for recognizing defects in helicopter turboshaft engines



f

Figure 9: Results of the effectiveness of the neural network classifier for recognizing defects in helicopter turboshaft engines (continuation)

It was found (fig. 7) that as the depth of node defects δx_i increases in the training interval 0 ... 10 %, the recognition efficiency first increases (the value of $\Delta\Sigma$ decreases), reaching its maximum (the minimum value of $\Delta\Sigma$), and then decreases. This feature of the dependence $\Delta\Sigma = f(\delta x_i)$ can be used for practical purposes, for example, the boundaries of the training interval should be selected from the condition of obtaining the greatest efficiency in recognizing the defective state of the engine (for an engine compressor unit this may be the boundary value of the reduced efficiency of the compressor at which it is required engine flushing, etc.).

6. Discussions

The results of a comparative analysis of the accuracy and error in diagnosing defects in helicopters TE units (on the example of defects in the compressor and compressor turbine) are given in table 9.

Table 9
Comparative analysis results

Neural network architecture	Accuracy	Loss
Developed neural network classifier	0.998	0.0011
Fully Connected Deep Networks RTRN	0.991	0.0045
Kohonen Neural Network (proposed by Professor Serhi Zhernakov)	0.970	0.022
Three-layer perceptron (proposed by Professor Serhi Zhernakov)	0.962	0.014
Cosco Neural Network	0.921	0.019
Elman Neural Network	0.899	0.032
Hamming Neural Network	0.873	0.044
Hopfield Neural Network	0.854	0.056
Jordan Neural Network	0.836	0.069
Radial basis function network	0.817	0.091

Currently, several methods are known for parametric diagnostics of the GTE operational status, which can be divided into methods A, B, C, D and E [29]: A – method of diagnostic matrices; B – method based on solving a system of normal equations; C – method based on nonlinear optimization of a criterion characterizing the state of the engine; D – method of adjustment using a square objective function; E – method of adjustment using a modular objective function. The results of comparing the effectiveness of the neural network approach with methods A, B, C, D and E show (fig. 10) that the advantage of the developed neural network classifier over other methods increases as information about the controlled engine parameters decreases. Wherein: fig. 10, a: diagnostics by parameter T_C; fig. 10, b: diagnostics by

parameter T_G; fig. 10, c: diagnostics by parameter G_T; fig. 10, d: diagnostics by parameter G_T and T_G; fig. 10, e: diagnostics by parameter G_T and T_C; fig. 11, f: diagnostics by parameter T_C and T_G.

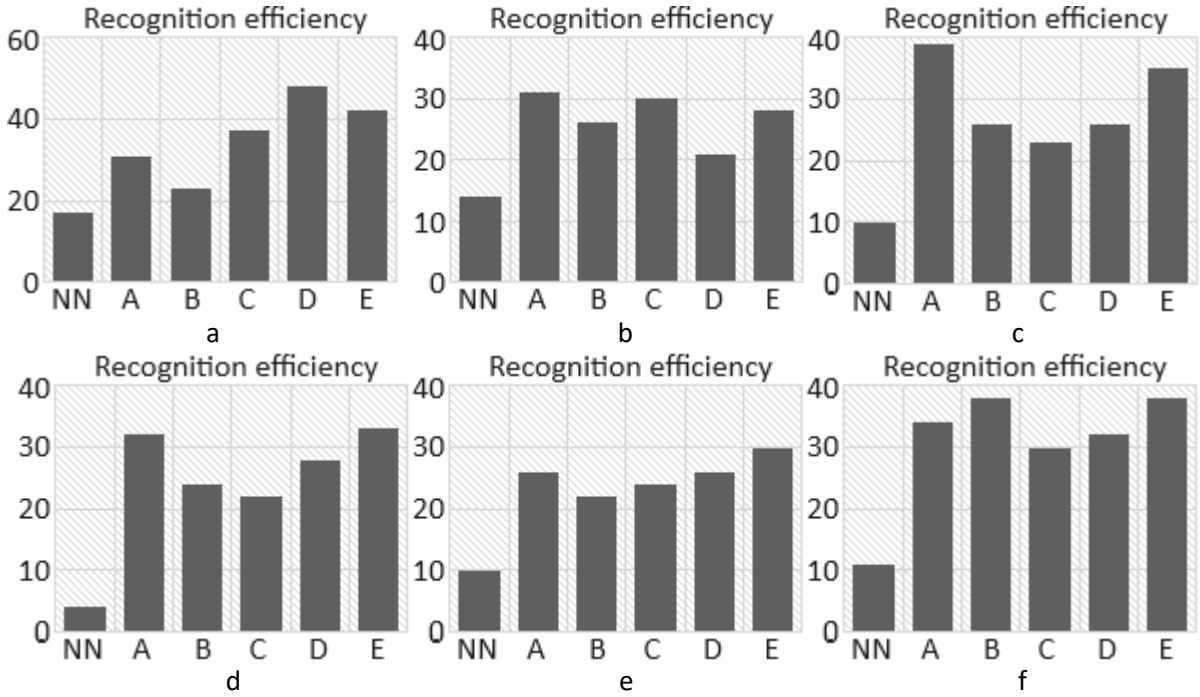


Figure 10: Comparative assessment of the effectiveness of methods for diagnosing the condition using the developed neural network classifier and methods A, B, C, D and E

Since the effectiveness of the considered methods for diagnostics the operational status of helicopter TE varies in different situations, it is obvious that the combined method is optimal.

7. Conclusions

The enhancement of the neural network diagnostics method involves the utilization of an ensemble comprising six radial-basis neural networks, a perceptron, a Kohonen neural network, and a hybrid neural network. This approach, applied to experimental data recorded during helicopter operation or data obtained through a mathematical model, proves effective in addressing the challenge of diagnosing the operational status of helicopter engines during flight.

Unlike diagnostic methods reliant on calculating the thermogas-dynamic parameters of helicopters turboshaft engines using nonlinear element-by-element engine models, the neural network diagnostics method is refined by training the neural network based on a small training sample. The quality of the resulting neural network model is then evaluated on a meticulously organized test sample.

The hybrid neural network training algorithm is enhanced by combining the error backpropagation algorithm with a genetic algorithm. This involves altering the range of initialization for neuron weights and biases in the input, hidden, and output layers. This refinement enables the training of the hybrid neural network to determine the degree of membership of indicator values at their inputs to a specified class characterizing the operational status of helicopters turboshaft engines.

The merit of incorporating the Kohonen network in a neural network classifier (part of a neural networks ensemble) for diagnosing helicopters turboshaft engines operational status in flight modes lies in its capacity for automatic classification (clustering) without expert instructions. This involves processing a training sample composed of real or calculated (reference) data across various engine operating modes.

For decision-making regarding the location and nature of defects in helicopters turboshaft engines, the estimation of neuroclassifier outputs using the nearest neighbor rule with precedents (codes of reference condition) can be employed. The metric value (distance to the nearest precedent) provides insight into defect intensity or multiplicity (i.e., the number of simultaneously manifesting defects).

The developed neural network classifier, initially designed for helicopters turboshaft engines, has the potential for diagnosing other engine types, such as turbojet and turboprop engines used in aircraft power plants. However, when extending the use of the classifier to other aircraft engines, considerations must be made for their structural blocks (e.g., fan, low-pressure compressor, high-pressure compressor, etc.) and, consequently, the applied thermogas-dynamic parameters.

8. References

[1] E. L. Ntantis, P. Botsaris, Diagnostic methods for an aircraft engine performance, Journal of Engineering Science and Technology, vol. 8, no. 4 (2015) 64–72. doi: 10.25103/jestr.084.10

[2] B. Li. Y.-P. Zhao, Group reduced kernel extreme learning machine for fault diagnosis of aircraft engine, Engineering Applications of Artificial Intelligence, vol. 96 (2020) 103968. doi: 10.1016/j.engappai.2020.103968.

[3] H. Hanachi, J. Liu, C. Mechefske, Multi-mode diagnosis of a gas turbine engine using an adaptive neuro-fuzzy system, Chinese Journal of Aeronautics, vol. 31, issue 1 (2018) 1–9. doi: 10.1016/j.cja.2017.11.017

[4] S. Pang, Q. Li, B. Ni, Improved nonlinear MPC for aircraft gas turbine engine based on semi-alternative optimization strategy, Aerospace Science and Technology, vol. 118 (2021) 106983. doi: 10.1016/j.ast.2021.106983

[5] S. Herzog, C. Tetzlaff, F. Worgotter, Evolving artificial neural networks with feedback, Neural Networks, vol. 123 (2020) 153–162. doi: 10.1016/j.neunet.2019.12.004

[6] M. Lungu, R. Lungu, Automatic control of aircraft lateral-directional motion during landing using neural networks and radio-technical subsystems, Neurocomputing, vol. 171 (2016) 471–481. doi: 10.1016/j.neucom.2015.06.084

[7] F. D. Amare, S. I. Gilani, B. T. Aklilu, A. Mojahid, Two-shaft stationary gas turbine engine gas path diagnostics using fuzzy logic, Journal of Mechanical Science and Technology, vol. 31 (2017) 5593–5602. doi: 10.1007/s12206-017-1053-9

[8] J. Zeng, Y. Cheng, An Ensemble Learning-Based Remaining Useful Life Prediction Method for Aircraft Turbine Engine, IFAC-PapersOnLine, vol. 53, issue 3 (2020) 48–53. doi: 10.1016/j.ifacol.2020.11.009

[9] D. El-Masri, F. Petrillo, Y.-G. Gueheneuc, A. Hamou-Lhadj, A. Bouziane, A systematic literature review on automated log abstraction techniques, Information and Software Technology, vol. 122 (2020) 106276. doi: 10.1016/j.infsof.2020.106276

[10] J. Friederich, G. Lugaresi, S. Lazarova-Molnar, A. Matta, Process Mining for Dynamic Modeling of Smart Manufacturing Systems: Data Requirements, Procedia CIRP, vol. 107 (2022) 546–551. doi: 10.1016/j.procir.2022.05.023

[11] S. Kiakojoori, K. Khorasani, Dynamic neural networks for gas turbine engine degradation prediction, health monitoring and prognosis, Neural Computing & Applications, vol. 27, no. 8 (2016) 2151–2192. doi: 10.1007/s00521-015-1990-0

[12] Y. Shen, K. Khorasani, Hybrid multi-mode machine learning-based fault diagnosis strategies with application to aircraft gas turbine engines, Neural Networks, vol. 130 (2020) 126–142. doi: 10.1016/j.neunet.2020.07.001

[13] B. Li, Y.-P. Zhao, Y.-B. Chen, Unilateral alignment transfer neural network for fault diagnosis of aircraft engine, Aerospace Science and Technology, vol. 118 (2021) 107031. doi: 10.1016/j.ast.2021.107031

[14] Z. Wu, J. Li, A Framework of Dynamic Data Driven Digital Twin for Complex Engineering Products: the Example of Aircraft Engine Health Management, Procedia Manufacturing, vol. 55 (2021) 139–146. doi: 10.1016/j.promfg.2021.10.020

[15] O. Avrunin, S. Vladov, M. Petchenko, V. Semenets, V. Tatarinov, H. Telnova, V. Filatov, Y. Shmelov and N. Shushlyapina. Intelligent automation systems, Kremenchuk, Novabook, 2021. 322 p. doi: 10.30837/978-617-639-347-4

[16] S. Vladov, I. Dieriabina, O. Husarova, L. Pylypenko, A. Ponomarenko, Multi-mode model identification of helicopters aircraft engines in flight modes using a modified gradient algorithms

for training radial-basic neural networks, Visnyk of Kherson National Technical University, no. 4 (79) (2021) 52–63. doi: 10.35546/kntu2078-4481.2021.4.7
[17] O. Khrebtova, O. Shapoval, O. Markov, V. Kukhar, N. Hrudkina and M. Rudych, Control Systems for the Temperature Field During Drawing, Taking into Account the Dynamic Modes of the Technological Installation, in: Proceedings of the 2022 IEEE 4th International Conference on Modern Electrical and Energy System (MEES), Kremenchuk, Ukraine, 2022, pp. 521–525. doi: 10.1109/MEES58014.2022.10005724
[18] S. Vladov, Y. Shmelov, R. Yakovliev, Modified Neural Network Method for Diagnostics the Helicopters Turboshaft Engines Operational Status at Flight Modes, in: Proceedings of the IEEE International Conference on System Analysis & Intelligent Computing (SAIC), Kyiv, Ukraine, October 04–07, 2022, pp. 224–229. doi: 10.1109/SAIC57818.2022.9923025
[19] I. Krivosheev, K. Rozhkov, N. Simonov, Complex Diagnostic Index for Technical Condition Assessment for GTE, Procedia Engineering, vol. 206 (2017) 176–181. doi: 10.1016/j.proeng.2017.10.456
[20] J. Rabcan, V. Levashenko, E. Zaitseva, M. Kvassay, S. Subbotin, Non-destructive diagnostic of aircraft engine blades by Fuzzy Decision Tree, Engineering Structures, vol. 197 (2019) 109396. doi: 10.1016/j.engstruct.2019.109396
[21] Z. Wang, M. Chen, J. Chen, Solving multiscale elliptic problems by sparse radial basis function neural networks, Journal of Computational Physics, vol. 492 (2023) 112452. doi: 10.1016/j.jcp.2023.112452
[22] J. Bai, G.-R. Liu, A. Gupta, L. Alzubaidi, X.-Q. Feng, Y. T. Gu, Physics-informed radial basis network (PIRBN): A local approximating neural network for solving nonlinear partial differential equations, Computer Methods in Applied Mechanics and Engineering, vol. 415 (2023) 116290. doi: 10.1016/j.cma.2023.116290
[23] X. Li, W. Yao, Q. Zhou, J. Wang, Fault diagnosis of Wind Turbine Pitch Systems based on Kohonen network, in: Proceedings of the 3rd International Conference on Mechatronics, Robotics and Automation (ICMRA 2015), Shenzhen, China, 2015, pp. 1168–1174.
[24] E. Nushtaeva, I. Sannikov, Research of neural network training using a hybrid algorithm that includes elements of backpropagation algorithm and genetic algorithm, International scientific journal "Bulletin of Science", vol. 3, no. 12 (33) (2020) pp. 153–160.
[25] S. Vladov, Y. Shmelov, R. Yakovliev, Modified Helicopters Turboshaft Engines Neural Network On-board Automatic Control System Using the Adaptive Control Method, ITTAP'2022: 2nd International Workshop on Information Technologies: Theoretical and Applied Problems, November 22–24, 2022, Ternopil, Ukraine, CEUR Workshop Proceedings, vol. 3309 (2022) 205–224.
[26] S. Vladov, Y. Shmelov, R. Yakovliev, M. Petchenko, Modified Neural Network Fault-Tolerant Closed Onboard Helicopters Turboshaft Engines Automatic Control System, COLINS-2023: 7th International Conference on Computational Linguistics and Intelligent Systems, Volume I: Machine Learning Workshop, April, 20–21, 2023, Kharkiv, Ukraine, CEUR Workshop Proceedings, vol. 3387 (2023) 160–179.
[27] S. Vladov, Y. Shmelov, R. Yakovliev, Methodology for Control of Helicopters Aircraft Engines Technical State in Flight Modes Using Neural Networks, The Fifth International Workshop on Computer Modeling and Intelligent Systems (CMIS-2022), May, 12, 2022, Zaporizhzhia, Ukraine, CEUR Workshop Proceedings (ISSN 1613-0073) vol. 3137 (2022) 108–125. doi: 10.32782/cmis/3137-10
[28] F. M. Tu, M. H. Wei, J. Liu, A coupling model of multi-feature fusion and multi-machine learning model integration for defect recognition, Journal of Magnetism and Magnetic Materials, vol. 568 (2023) 170395. doi: 10.1016/j.jmmm.2023.170395
[29] K. Mammadova, A. Abbasova, Empirical examples based on neuro-fuzzy systems, in: Proceedings of the Current Issues and Prospects for The Development of Scientific Research: Proceedings of the 7th International Scientific and Practical Conference, 2023, Orleans, France, pp. 635–642. doi: 10.51582/interconf.19-20.04.2023.068