Methodology for Control of Helicopters Aircraft Engines Technical State in Flight Modes Using Neural Networks

Serhii Vladov[a], Yurii Shmelov[a] and Ruslan Yakovliev[a]

[a] *Kremenchuk Flight College of Kharkiv National University of Internal Affairs, vul. Peremohy, 17/6, Kremenchuk, Poltavska Oblast, Ukraine, 39605*

Abstract

In this work, the creation of a methodology for control the technical state of helicopters aircraft engines using recurrent neural networks, which, unlike other types of neural networks, have a more optimal method for training a neural network, and also after training allows you to get a lower percentage of errors in comparison with a convolutional neural network and multilayer perceptron. A mathematical description of the methodology for control the technical state of helicopters aircraft engines in flight mode using neural networks based on the numerical solution of a discrete optimal control problem has been implemented. The quality of the trained neural network is assessed, including the calculation of the testing error, which is no more than 1.2 % of the deviation from the a priori correct result on the vectors of the test set of sets. With an allowable value of 2 %, this allows us to speak about the efficiency of the method.

Keywords

Aircraft engine, recurrent neural network, testing error, optimal control, backpropagation method.

CEUR-WS.org/Vol-3137/paper10.pdf

1. Introduction

In recent decades, the development and improvement of aircraft gas turbine engines (GTE), including helicopters, has been accompanied by toughening requirements for the reliability and efficiency of their automatic control systems (ACS). Modern helicopters gas turbine engines are complex technical devices that differ in the variety of physical processes occurring in them and are characterized by multidimensionality, multi-connectivity, nonlinearity, non-stationary work processes, a significant influence of operating modes and external conditions on the characteristics of their functioning. The listed features lead to the formation of a stable trend in the development of ACS for gas turbine engines of helicopters, characterized by a constant increase in the complexity and the number of tasks solved with their help. One of the important tasks is to improve the methods and algorithms for controlling the helicopters gas turbine engine in the conditions of a helicopter flight in terms of thermogasdynamic parameters, which is due to the presence of strict requirements for ensuring the safety and efficiency of flights [1, 2].

2. Literature review

Modern approaches to control of aircraft gas turbine engines technical state are described in [3–5]. The main factors that must be taken into account in the helicopter flight mode include uncertainty

The Fifth International Workshop on Computer Modeling and Intelligent Systems (CMIS-2022), May 12-th, 2022, Zaporizhzhia, Ukraine
EMAIL: ser26101968@gmail.com (S. Vladov); nviddil.klk@gmail.com (Yu. Shmelov); ateu.nv.klk@gmail.com (R. Yakovliev)
ORCID: 0000-0001-8009-5254 (S. Vladov); 0000-0002-3942-2003 (Yu. Shmelov); 0000-0002-3788-2583 (R. Yakovliev)
© 2022 Copyright for this paper by its authors.
Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).
CEUR Workshop Proceedings (CEUR-WS.org)

factors, such as incompleteness of a priori and operational information, inaccuracy of mathematical models of GTE, errors of sensors and actuators, changes in engine characteristics during operation, occurrence of possible failures of GTE functional elements [6, 7].

In recent years, their construction in the class of intelligent control systems providing robustness, adaptability and fault tolerance of GTE control processes in conditions of uncertainty have been considered as a promising direction in solving problems of monitoring the technical state of gas turbine engines.

Despite a significant amount of research in the development of GTE control algorithms, the existing information technologies for controlling GTE parameters are not perfect for a number of reasons. On the one hand, this is a weak informational "linkage", the absence of elements of "intelligence", allowing to quickly, efficiently and effectively support decision-making in emergency situations in flight.

On the other hand, it is the complexity of the processes occurring in GTE ACS, the complexity of their mathematical description, the limited composition of the measured parameters, their technological spread, etc. These lead to the need for comprehensive automation and intellectualization of decision-making processes of GTE ACS technical state [7] under conditions of uncertainty, including using the control method according to the FDI (Fault Detection and Identification) model [8–11].

Thus, the development of intelligent algorithms for automatic control of helicopters gas turbine engine technical state, as well as the study of the features of their practical application, taking into account the limitations on the available computing resources of an onboard digital computer, is an urgent scientific and practical task.

3. The use of neural networks in the problems of control of aircraft gas turbine engines technical state

The implementation of the FDI method [8–11] allows maximum consideration of the individual characteristics of a gas turbine engine by using a mathematical model that adapts (adjusts) to the individual characteristics of the latter [12]. When using neural networks to solve the problems of monitoring the technical state of a gas turbine engine of helicopters, the available a priori information is presented to the neural network in the form of ready-made solutions (problem books), on the basis of which the process of its training (additional training) is carried out. When assessing the quality of the network, data from the test sample are fed to its input, on the basis of which it calculates the vector of deviations (the difference between the output of the neural network and the desired characteristics) [11, 12].

Let us consider the problem of control of helicopters GTE technical state in flight mode based on neural networks in the following formulation [13, 14]. We will assume that all possible states of the gas turbine engine can be divided into two classes S_0 (all serviceable states of the gas turbine engine) and $\overline{S}_0$ (all faulty states, characterized by the presence of at least one defect in the operation of the gas turbine engine), uniting related states that are close to each other according to certain integral indicators. Required based on the results of a limited number of measurements of the vector of engine output parameters $Y(t_i)$, $t_i \in T$ (where t_i – discrete moments of time; T – observation interval), make a decision on whether the GTE belongs to one of the specified classes of states. The solution of this problem in general form is reduced to finding a certain separating function (hypersurface) in the space of controlled parameters of the gas turbine engine. To solve this problem, in this work, an approach is implemented based on the construction of the specified decision rule using neural networks.

4. Development of a methodology for control of helicopters aircraft gas turbine engines technical state in flight mode using neural networks

Based on the above, within the framework of the task of control of helicopters gas turbine engine technical state, in this work, neural network algorithms are investigated, a formalized formulation of problems is given, recommendations for solving this spectrum of problems are formed, and an engineering methodology is proposed.

The general scheme of the methodology is shown in fig. 1. The main thermogasdynamic parameters of the aircraft engine (recorded by sensors and calculated using a mathematical model) are fed to the input.

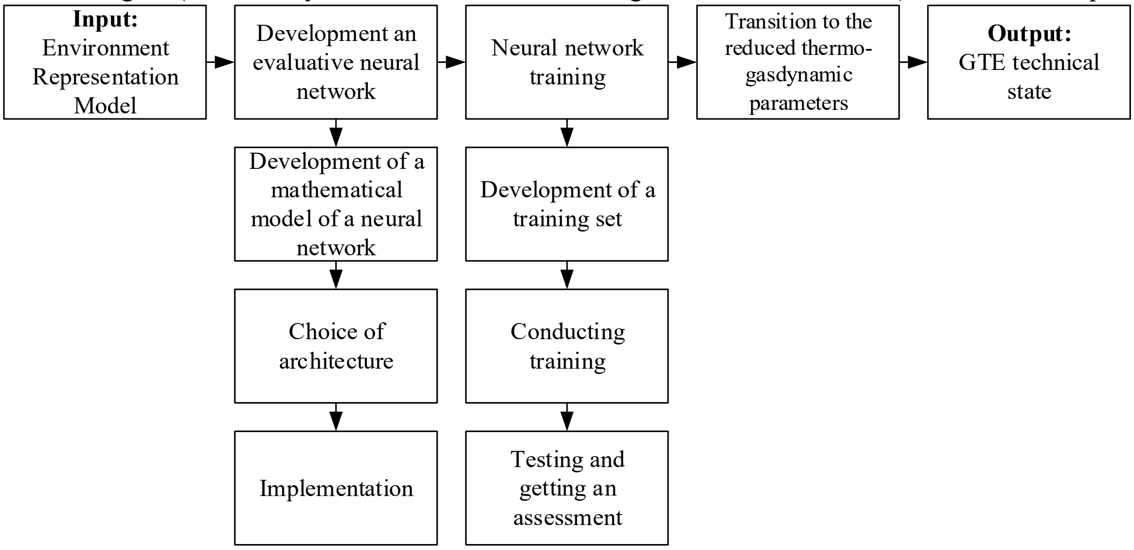


Figure 1: Block diagram of the methodology for control of helicopters aircraft gas turbine engines technical state in flight mode using neural networks

At the first stage, a model of an aircraft engine is formed by systematizing the data. Then a neural network is created – in view of the fact that neural networks have the property of non-universality, it is necessary to carry out procedures for choosing an architecture for each new task. For each task, it is necessary to select several types of architectures, conduct testing in the context of a specific task. Accordingly, it becomes necessary to train / retrain the newly created neural network. Then, for the convenience of further use and displaying the graphical representation, the transition to the reduced thermogasdynamic parameters [15, 16]. This stage is necessary for human control of the neural network, in the case when there is an additional stage of control.

5. Review and selection of neural network architecture

The architecture of neural networks most often used to solve problems of monitoring the technical state of complex dynamic objects is a feedforward network, the input neurons of which are fed with the values of the attributes of the classified object, and the output is a label or a numeric code of the class. Multilayer perceptrons are commonly used. In such networks, the elements of the feature vector arrive at the input neurons and are distributed to all neurons of the first hidden layer of the neural network, and as a result, the dimension of the problem changes [17, 18].

A convolutional neural network processes the data not entirely, but in fragments, but the data is not split into parts, but a kind of sequential run is carried out. Then the data is transferred further along the layers [19, 20]. In addition to convolutional layers, union layers are also used. Combine layers are compressed with depth (usually a power of two). Several perceptrons (feedforward network) are added to the final layers for further data processing.

In the process of implementing a convolutional neural network, technical problems often arise related to the format of the input data and the model used, which did not allow the implementation of this network and adequately assess the capabilities of this architecture for solving the problem of monitoring the technical condition of an aircraft engine.

The use of a recurrent neural network can significantly reduce the computational complexity of the backpropagation method, which will be used to train the neural network [21, 22]. To train the neural network, we will use the error backpropagation algorithm, this algorithm is optimal for the classification problem using a neural network. It is also worth noting that this neural network needs long-term training on a larger number of training sets than a multilayer perceptron.

Comparing the proposed architectures, we can conclude that for the task at hand, the optimal option would be a recurrent neural network, which, after training, makes it possible to obtain a lower percentage of errors in comparison with a convolutional neural network and a multilayer perceptron (comparison graphs are shown in fig. 2), in addition, it possesses a simpler implementation than a convolutional neural network.

This architecture assumes a more optimal method for training a neural network. As a disadvantage, there will be a larger number of training sets and a longer training time, which is compensated by the higher accuracy of calculations (fig. 3).

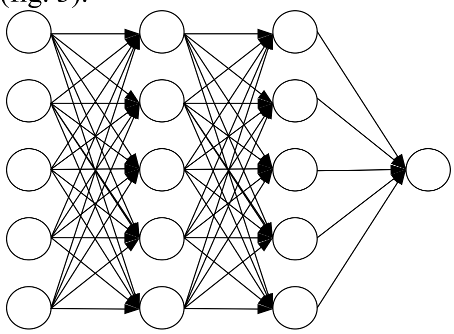


Figure 2: Final neural network architecture

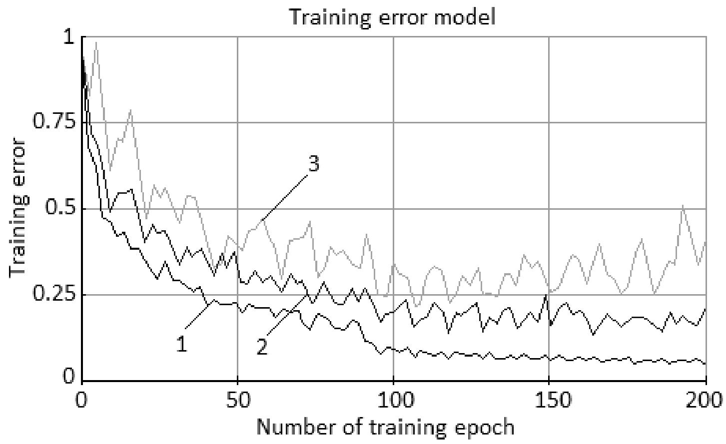


Figure 3: Comparison of the percentage of errors of classical neural network architectures designed for control complex dynamic objects: 1 – recurrent neural network; 2 – multilayer perceptron; 3 – convolutional neural network

It is also worth noting that a recurrent neural network of the LSTM structure [23] has been successfully applied to solve the problem of identification of an aviation GTE. Therefore, in this work, a recurrent neural network of the LSTM structure is used to solve the problem of monitoring the technical state of aircraft GTEs of helicopters in flight mode. At the same time, increasing the accuracy of the GTE model as a control object by modifying the architecture of the LSTM network.

6. Mathematical description of the methodology for control of helicopters gas turbine engine technical state in flight mode using neural networks

Let us solve the optimal control problem that simulates the dynamics of the considered artificial neural network in the notation given below. The dynamics of a network of *n* neurons is described by a system of differential equations with delay:

$$\dot{x}_l(t) = -\gamma_i x_i(t) + \sum_{j=1}^{n}\left[\omega_{ij}(t)\, g\left(x_j(t)\right) + u_i(t)\right];\ i, j = \overline{1, N}; \tag{1}$$

where $\dot{x}_l(t) = -\gamma_i x_i(t) + g_i(z_i(t)) + u_i(t);$ $i,j=\overline{1,N};$ $z_i(t) = \sum_{j=1}^{n} \omega_{ij}(t) x_j(t-h);$

$$g_i(z_i(t)) = \frac{1}{1+e^{-\lambda \sum_{j=1}^{n} \omega_{ij}(t) x_j(t-h)}}; \text{ thus, } \dot{x}_l(t) = -\gamma_i x_i(t) + \frac{1}{1+e^{-\lambda \sum_{j=1}^{n} \omega_{ij}(t) x_j(t-h)}} + u_i(t); \ i,j=\overline{1,N}.$$

From the point of view of a biological prototype, i.e. neural network, we can say that the equation describes the accumulated potential (electrical impulse) of a neuron at a given time, as well as its change over time. The potential of a neuron is formed and changes under the influence of many factors: $x_i(t)$ – own potential of a neuron at a given time; γ_i – intrinsic attenuation of the i-th neuron, describes the effect on the neuron of its own forces, which negatively affect the potential, as well as signal attenuation during transmission from one neuron to others. Sum $\sum_{j=1}^{n} \omega_{ij}(t) x_j(t-h)$ can be called the sum of the potentials of an ensemble of neurons. This is the sum of the impact of all neighboring neurons on the i-th neuron. Sum $\sum_{j=1}^{n} \omega_{ij}(t) x_j(t-h)$ is the main element in the formation of the potential of the i-th neuron, so this sum will be called the body of the i-th neuron $\omega_{ij}(t) x_j(t)$. Element $x_j(t-h)$ shows the lag of the neural network signal. Thus, the potential of the i-th neuron is sufficiently influenced by the residual impulse of neurons at the previous moment in time.

Control functions $\omega_{ij}(t)$ describe the axons of neurons – an electrical or chemical impulse that is transmitted from one neuron to another, thereby changing the potentials, is the most important connecting element of the neural network, since responsible for the interaction and performance of the entire network. In this case, this is the impact on the i-th neuron, j-th neuron.

The activation function $g_i(z_i(t))$ transforms the accumulated potential of the neuron according to some functional dependence. The prototype is the processes occurring in the body of a neuron, caused, for example, by signals or impulses from the peripheral nervous system of the body (due to any changes in the external environment).

The intrinsic potential of neurons should not go beyond the limits:

$$x_i(t) \le B_i; \ i=\overline{1,n}. \tag{2}$$

The characteristics of neurons at the initial moment of time are known:

$$x_i(0) = a_i; \ i=\overline{1,n}; \ x_i(t) = \varphi_i(t); \ i=\overline{-h,0}. \tag{3}$$

The control function $u_i(t)$ characterizes the external influence on the i-th neuron. These can be any changes in the environment to which the body reacts by changing the rate of transmission of electrical and chemical impulses of the nervous system. Restrictions on control functions are known:

$$|\omega_{ij}| \le b; \ |u_i| \le c. \tag{4}$$

The goal of controlling the dynamics of a neural network is to train the network, which implies the following tasks (criteria):

1) At the final moment of time, the characteristics of the neurons must coincide with the input data A_i;

2) During the execution of the process, the characteristics of neurons should not go beyond the specified range of B_i values.

3) Control u_i should strive for the minimum value for the given process.

4) Control ω_{ij} should also tend to the minimum value for this process.

Control tasks can be formalized in the form of the following target functional:

$$I([x],[\omega],[u],[t]) = S\sum_{i=1}^{n}(x_i(T)-A_i)^2 + \sum_{i=1}^{n}\int_0^T M_i(\max(0; x_i(t)-B_i))^2 dt + L\sum_{i=1}^{n}\int_0^T u_i^2(t) dt +$$
$$+K\sum_{j=1}^{n}\sum_{i=1}^{n}\int_0^T \omega_{ij}^2(t) dt \to \inf; \ i=\overline{1,n}. \tag{5}$$

The main practical goal is to obtain optimal process controls, with the help of which the minimum of the target functional is achieved.

We obtain the optimal process controls by the gradient descent method (back propagation of the error) [24].

To solve the optimal control problem (2)–(5), we use the Pontryagin maximum principle [25]. We introduce the Pontryagin function:

$$H([x],[\omega],[u],[t]) = -\lambda_0 \sum_{i=1}^{n} \left(M_i \left(\max\left(0; x_i - B_i\right)\right)^2 + L u_i^2 \right) - \lambda_0 K \sum_{j=1}^{n} \sum_{i=1}^{n} \omega_{ij}^2 + \sum_{i=1}^{n} p_i(t) \times$$

$$\times \left(-\gamma_i x_i + \frac{1}{1 + e^{-\lambda \sum_{j=1}^{n} \omega_{ij}(t) x_j(t-h)}} + u_i \right); \ i, j = \overline{1, n}. \tag{6}$$

Let's write down the maximum principle:

$$-\lambda_0 \sum_{i=1}^{n} M_i \left(\max\left(0; \overline{x_l} - B_i\right)\right)^2 - \lambda_0 \sum_{i=1}^{n} L \overline{u_i}^2 - \lambda_0 K \sum_{j=1}^{n} \sum_{i=1}^{n} \overline{\omega_{ij}}^2 + \lambda_0 \sum_{i=1}^{n} L \overline{v_i}^2 + \sum_{i=1}^{n} p_i(t) \left(-\gamma_i \overline{x_l} + g_i\left(z_i(t)\right) + \overline{u_l} \right) =$$

$$= \max\left(-\lambda_0 \sum_{i=1}^{m} M_i \left(\max\left(0; \overline{x_l} - B_i\right)\right)^2 \right) - \lambda_0 \sum_{i=1}^{n} L \overline{v_i}^2 - \lambda_0 K \sum_{j=1}^{n} \sum_{i=1}^{n} \overline{\omega_{ij}}^2 + \sum_{i=1}^{n} p_i(t) \left(-\gamma_i \overline{x_l} + g_i\left(\overline{x_j} \omega_{ij}\right) + v_i \right) =$$

$$= -\lambda_0 \sum_{i=1}^{n} M_i \left(\max\left(0; \overline{x_l} - A_i\right)\right)^2 - \sum_{i=1}^{n} p_i(t) \gamma_i \overline{x_l} + \sum_{i=1}^{n} \left(\underbrace{\max}_{\omega_{ij}} \left(p_i(t) g_i\left(\overline{x_j} \omega_{ij}\right)\right) - \lambda_0 K \sum_{j=1}^{n} \omega_{ij}^2 \right) +$$

$$+ \sum_{i=1}^{n} \underbrace{\max}_{v_i} \left(p_i(t) v_i - \lambda_0 L \overline{v_i}^2 \right).$$

Conjugate vector functions are calculated according to the expressions:

$$\dot{p}_k(t) = -\frac{\partial H}{\partial y_k}(t+h);$$

where $y_k = x_k(t-h)$; thus,

$$\dot{p}_k(t) = 2\lambda_0 M_k \max\left(0; x_k - A_k\right) + p_k(t) \gamma_k - \lambda \sum_{i=1}^{n} p_i(t+h) \overline{\omega_{ik}} \frac{e^{-\lambda \sum_{j=1}^{n} \omega_{ij}(t) x_j(t-h)}}{\left(1 + e^{-\lambda \sum_{j=1}^{n} \omega_{ij}(t) x_j(t-h)}\right)^2};$$

and satisfy the transversality conditions $p_k(t) = -\lambda_0 \frac{\partial \Phi}{\partial x_k} = -2\lambda_0 \left(x_k(T) - A_k \right)$.

The maximum principle allows us to reduce the problem of optimal process control to solving a boundary value problem:

$$\dot{x}_l(t) = -\gamma_i x_i(t) + \sum_{j=1}^{n} \overline{\omega_{ij}}(t) g\left(x_j(t)\right) + \overline{u_i}(t); \ i, j = \overline{1 ... n};$$

$$\dot{x}_l(t) = -\gamma_i x_i(t) + g_i\left(z_i(t)\right) + u_i(t);$$

Wherein $x_i(0) = a_i$; $i, j = \overline{1 ... n}$; where $\overline{\omega_{ij}}$, $\overline{u_{ij}}$ – optimal controls.

7. Numerical solution of a discrete optimal control task

Let us obtain a numerical solution to the discrete optimal control problem. For this, we will reduce the original problem to a discrete form. We split the segment [0, T] into q segments. Sampling step $\Delta t = \frac{T}{q}$. Let us approximate the system of differential equations according to the Euler scheme [26]:

$$\dot{x}_l\left(t^l\right) \approx \frac{x_i^{l+1} - x_i^l}{\Delta t}.$$

We denote $x_i\left(t\right) = x_i^l$; $h = v\Delta t$; $x_i\left(t-h\right) = x_i^{l-v}$, after that we approximate the integrals $\int_0^T M_i\left(\max\left(0; x_i\left(t\right) - A_i\right)\right)^2 dt$, $\int_0^T u_i^2\left(t\right)dt$ and $\int_0^T \omega_{ij}^2\left(t\right)dt$ by the method of rectangles:

$$\int_0^T M_i\left(\max\left(0; x_i\left(t\right) - A_i\right)\right)^2 dt \approx \sum_{l=0}^{q-1}\sum_{i=1}^{n} M_i\left(\max\left(0; x_i\left(t\right) - A_i\right)\right)^2 \Delta t;$$

$$\int_0^T u_i^2\left(t\right)dt = \sum_{l=0}^{q-1}\sum_{i=1}^{n}\left(u_i^l\right)^2 \Delta t;$$

$$\int_0^T \omega_{ij}^2\left(t\right)dt = \sum_{l=0}^{q-1}\sum_{i=1}^{n}\sum_{j=1}^{n}\left(\omega_{ij}^l\right)^2 \Delta t.$$

We get the recurrence expression for calculating the phase trajectory:

$$x_i^{l+1} = x_i^l + \left(-\gamma_i x_i^l + g_i^l\left(z_i^l\right) + u_i^l\right)\Delta t;$$

where $z_i^l = \sum_{j=1}^{n}\omega_{ij}^l x_j^{l-v}$; $g_i^l\left(z_i^l\right) = \dfrac{1}{1+e^{-\lambda\sum_{j=1}^{n}\omega_{ij}^l x_j^{l-v}}}$; thus, $x_i^{l+1} = x_i^l + \left(-\gamma_i x_i^l + \dfrac{1}{1+e^{-\lambda\sum_{j=1}^{n}\omega_{ij}^l x_j^{l-v}}} + u_i^l\right)\Delta t$;

wherein $\left(g_l'\right)_{w_{ij}^l} = \lambda\dfrac{e^{-\lambda\sum_{j=1}^{n}\omega_{ij}^l x_j^{l-v}}\, x_j^{l-v}}{\left(1+e^{-\lambda\sum_{j=1}^{n}\omega_{ij}^l x_j^{l-v}}\right)^2}$; $\left(g_l'\right)_{x_k^m} = \dfrac{e^{-\lambda\sum_{j=1}^{n}\omega_{ik}^m x_k^{m-v}}\,\omega_{ik}^{m-v}}{\left(1+e^{-\lambda\sum_{j=1}^{n}\omega_{ik}^m x_k^{m-v}}\right)^2}$.

Initial values:

$$x_i\left(0\right) = a_i;\ x_i\left(t\right) = \varphi_i\left(t\right);\ t = \overline{-h, 0}. \tag{6}$$

The objective function will take the form:

$$I = S\sum_{i=1}^{n}\left(x_i^q - A_i\right)^2 + \sum_{l=0}^{q-1}\sum_{i=1}^{n} M_i\left(\max\left(0; x_i^l - B_i\right)\right)^2 \Delta t + L\sum_{l=0}^{q-1}\sum_{i=1}^{n}\left(u_i^l\right)^2 \Delta t + K\sum_{l=0}^{q-1}\sum_{i=1}^{n}\sum_{j=1}^{n}\left(\omega_{ij}^l\right)^2 \Delta t \to \inf. \tag{7}$$

Using the Lagrange multiplier method, we obtain the necessary optimality condition. Let us compose the Lagrange function [27]:

$$L = \lambda_0 S\sum_{i=1}^{n}\left(x_i^q - A_i\right)^2 + \sum_{l=0}^{q-1}\sum_{i=1}^{n} M_i\left(\max\left(0; x_i^l - B_i\right)\right)^2 + L\sum_{l=0}^{q-1}\sum_{i=1}^{n}\left(u_i^l\right)^2 \Delta t + K\sum_{l=0}^{q-1}\sum_{i=1}^{n}\sum_{j=1}^{n}\left(\omega_{ij}^l\right)^2 \Delta t +$$
$$+\sum_{l=0}^{q-1}\sum_{i=1}^{n} p_i^{l+1}\left(x_i^{l+1} - x_i^l - \Delta t\left(-\gamma_i x_i^l + \frac{1}{1+e^{-\lambda\sum_{j=1}^{n}\omega_{ij}^l x_j^{l-v}}} + u_i^l\right)\right).$$

Let us write down the stationarity conditions:

$$\frac{\partial L}{\partial x_k^m} = p_k^m - p_k^{m+1} + p_k^{m+1}\gamma_k\Delta t - \Delta t\sum_{i=1}^{n} p_i^{m+1}\frac{\partial g_k^m\left(z_k^m\right)}{\partial x_k^m} + 2M_k\lambda_0\Delta t\left(\max\left(0; x_k^m - B_m\right)\right) = 0;$$

$$k = \overline{1...n};\ m = \overline{0...q-1}.$$

$$\frac{\partial L}{\partial x_k^m} = p_k^m - p_k^{m+1} + p_k^{m+1}\gamma_k\Delta t - \Delta t\sum_{i=1}^{n} p_i^{m+v+1}\omega_{ik}^{m+v}\frac{e^{-\lambda\sum_{j=1}^{n}\omega_{ik}^m x_k^{m-v}}}{\left(1+e^{-\lambda\sum_{j=1}^{n}\omega_{ik}^m x_k^{m-v}}\right)^2} + 2M_k\lambda_0\Delta t\left(\max\left(0; x_k^m - B_m\right)\right) = 0;$$

$$k=\overline{1...n};\;\; m=\overline{0...q-1};. \tag{8}$$

$$\frac{\partial L}{\partial x_k^q}=2\lambda_0 S\left(x_k^q-A_i\right)+x_k^q=0;\;\; k=\overline{1...n}. \tag{9}$$

Let's write down the gradient values:

$$\frac{\partial L}{\partial \omega_{km}^s}=-p_k^{s+1}\frac{\partial g_k^m\left(z_k^m\right)}{\partial \omega_{km}^s}\Delta t+2K\Delta t\omega_{km}^s;\;\; k=\overline{1...n};\;\; s=\overline{1...n};\; m=\overline{0...q-1};$$

$$\frac{\partial L}{\partial \omega_{km}^s}=-\lambda\Delta t p_k^{s+1}x_m^{s-v}\frac{e^{-\lambda\sum_{m=1}^{n}\omega_{km}^s x_m^{s-v}}}{\left(1+e^{-\lambda\sum_{m=1}^{n}\omega_{km}^s x_m^{s-v}}\right)^2}+2K\Delta t\omega_{km}^s; \tag{10}$$

$$\frac{\partial L}{\partial u_k^m}=2Lu_k^m\Delta t-p_k^{m+1}\Delta t;\;\; k=\overline{1...n};\;\; m=\overline{0...q-1}. \tag{11}$$

Let us express the impulses in terms of these equations p_k^m and p_k^q:

$$p_k^m=p_k^{m+1}-p_k^{m+1}\gamma_k\Delta t+\gamma\Delta t\sum_{i=1}^{n}p_i^{m+v+1}\omega_{ik}^{m+v}\frac{e^{-\lambda\sum_{k=1}^{n}\omega_{ik}^m x_k^{m-v}}}{\left(1+e^{-\lambda\sum_{k=1}^{n}\omega_{ik}^m x_k^{m-v}}\right)^2}-2M_k\lambda_0\Delta t\left(\max\left(0;x_k^m-B_k\right)\right); \tag{12}$$

$$p_k^q=-2\lambda_0 S\left(x_k^q-A_k\right). \tag{13}$$

After completing the passage to the limit, then we check the correspondence with the boundary value problem, thereby checking the correctness of the solution:

$$\frac{p_k^{m+1}-p_k^m}{\Delta t}=p_k^{m+1}\gamma_k-\lambda\sum_{i=1}^{n}\omega_{ik}^{m+v}p_i^{m+v+1}\frac{e^{-\lambda\sum_{j=1}^{n}\omega_{ij}^m x_j^{m-v}}}{\left(1+e^{-\lambda\sum_{j=1}^{n}\omega_{ij}^m x_j^{m-v}}\right)^2}+2M_k\lambda_0\left(\max\left(0;x_k^m-B_k\right)\right);$$

passing to the limit $\lim\limits_{\Delta t\to 0}\dfrac{p_k^{m+1}-p_k^m}{\Delta t}$, we get:

$$\dot{p}_k(t)=2\lambda_0 M_k\max\left(0;x_k-B_k\right)+p_k(t)\gamma_k-\lambda\sum_{i=1}^{n}p_i(t+h)\omega_{ik}(t+h)\frac{e^{-\lambda\sum_{j=1}^{n}\omega_{ij}(t)x_j(t-h)}}{\left(1+e^{-\lambda\sum_{j=1}^{n}\omega_{ij}(t)x_j(t-h)}\right)^2};$$

$$p_k^q=-2\lambda_0 S\left(x_k^q-A_k\right);$$

$$p_k(T)=-2\lambda_0 S\left(x_k(T)-A_k\right).$$

8. Development of an algorithm for finding a numerical solution

Let's formalize the resulting algorithm:

1) We set the parameters of the model.

2) We set the parameters of the method: ε – calculation accuracy, α – gradient descent step, q – number of dividing points of the segment and the initial set of controls $[u]^{(0)}$ and $[\omega]^{(0)}$.

3) Using expressions $\mathbf{W}^{(\Sigma)} = \mathbf{W}^{1} \cdot \mathbf{W}^{2} \cdot ... \cdot \mathbf{W}^{(p)}$ and $\mathbf{Y} = \mathbf{X} \cdot \mathbf{W}^{(\Sigma)}$, we find a collection $[x]^{(0)}$. We calculate $I^{(0)}$ by the expression $s = \sum_{i=1}^{n} x_i^2 \cdot w_i$, setting $[M]^{(1)}$.

4) Using expressions (12) and (13), starting from the q-th layer, we calculate $[p]^{(k)}$, k = 0, 1, ... We calculate $\left[\frac{\partial L}{\partial \omega_{km}^{s}} : \frac{\partial L}{\partial u_{k}^{s}} \right]^{(k)}$ by expressions (10) and (11).

5) Improving control at k + 1 steps: $[u]^{(k+1)} = [u]^{(k)} - \alpha \frac{\partial L}{\partial u}$, $[w]^{(k+1)} = [w]^{(k)} - \alpha \frac{\partial L}{\partial w}$.

6) We calculate $[x]^{(k+1)}$, then $[I]^{(k+1)}$.

7) Compare the values $[I]^{(k)}$ and $[I]^{(k+1)}$. If $[I]^{(k+1)} > [I]^{(k)}$, then we decrease the step of the gradient descent $\alpha = \frac{\alpha}{N}$, where $N \in \mathbb{N}$ we go to step (5) of the same iteration, otherwise go to step (8).

8) We check the condition $\left| [I]^{(k+1)} - [I]^{(k)} \right| < \varepsilon$, if it is satisfied, then we assume that the optimal solution for a given parameter $[M]^{(1)}$ has been found and go to step (9). Otherwise, go to the next iteration $k = k + 1$ and go to step (4).

9) Checking the conditions:

$$\sum_{l=1}^{q-1} \sum_{i=1}^{n} M_i^l \left(\max\left(0; x_i^l - A_i\right) \right)^2 \le \varepsilon; \quad \sqrt{\sum_{l=0}^{q} \sum_{i=1}^{n} \left(x_i^{l(k)} - x_i^{l(k+1)} \right)^2} \le \varepsilon; \quad \sqrt{\sum_{i=0}^{q} \sum_{i=1}^{n} \sum_{j=1}^{n} \left(\omega_{ij}^{l(k)} + \omega_{ij}^{l(k+1)} \right)^2} \le \varepsilon.$$

If all conditions are met together, we assume that the solution has been found; otherwise, we determine a new penalty coefficient $[M]^{(1)} = \beta [M]^{(0)}$ and go to step (3).

9. Algorithm for training a formalized neural network

The next step in the methodology for control of helicopters aircraft GTE technical state is training an artificial neural network. It is necessary to choose a training algorithm suitable for a specific task. According to the research results, the best results were obtained using the error backpropagation algorithm. This is primarily due to the fact that the backpropagation algorithm has been created and optimized for the selected neural network architecture and activation function, which makes it possible to maximize its potential [22, 23].

Its main idea is that the change in the weights of synapses takes into account the local gradient of the error function. The difference between the real and correct responses of the neural network, determined on the output layer, propagates in the opposite direction (fig. 4) – towards the flow of signals.

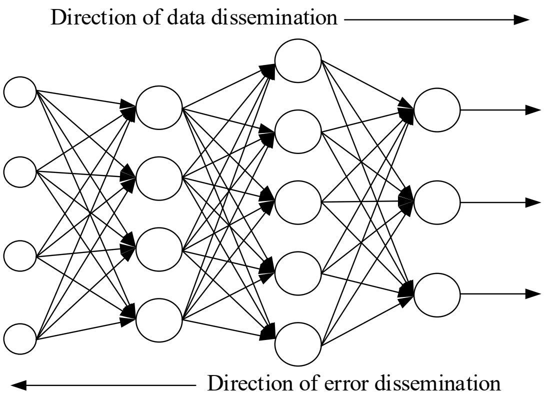


Figure 4: Backpropagation method [23] for multilayer fully connected neural network

As a result, each neuron is able to determine the contribution of each of its weights to the total network error. The simplest learning rule corresponds to the steepest descent method, that is, changes in synaptic weights are proportional to their contribution to the total error [28].

Of course, with such a training of a neural network, there is no certainty that it has trained in the best way, since there is always a possibility of the algorithm falling into a local minimum. For this, special techniques are used to "knock" the found solution out of the local extremum. If, after several such actions, the neural network converges to the same solution, then we can conclude that the solution found is most likely optimal [29].

The algorithm for training a neural network using the backpropagation procedure is constructed as follows [24, 30]:

1. Apply one of the possible images to the network inputs and, in the normal functioning of the neural network, when signals propagate from inputs to outputs, calculate the values of the latter.
2. Calculate the difference between the ideal and the obtained output values for the output layer. Calculate changes in layer weights.
3. Calculate the difference between the ideal and the obtained values of the output and change in weights for all other layers.
4. Adjust all weights to the neural network.
5. If the network error is significant, go to step 1. Otherwise, end. At step 1, all training images are alternately presented to the network in a random order so that the network, figuratively speaking, does not forget some as it memorizes others.

When the output value goes to zero, the training efficiency decreases markedly. With binary input vectors, on average, half of the weight coefficients will not be corrected; therefore, it is desirable to shift the range of possible values of neuron outputs [0, 1] within the limits [–0.5 ... 0.5], which is achieved by simple modifications of logistic functions. For example, an exponential sigmoid $\sigma(x) = \frac{1}{1+e^{-x}}$ is converted to a kind $\sigma(x) = -0.5 + \frac{1}{1+e^{-x}}$.

Now let's touch on the issue of the capacity of a neural network, that is, the number of images presented to its inputs that it is able to learn to recognize. For networks with more than two layers, it remains open. For a neural network with two layers, that is, an output and one hidden layer, the deterministic capacity of the network is estimated as follows:

$$\frac{N_w}{N_y} < C_d \frac{N_w}{N_y} < \log \frac{N_w}{N_y};$$

where C_d – deterministic capacity of the network; N_w – number of adjustable weights; N_y – number of neurons in the output layer.

It should be noted that this expression was obtained with some restrictions. First, the number of inputs N_x and neurons in the hidden layer N_h must satisfy the inequality $N_x + N_h > N_y$. Secondly, $\frac{N_w}{N_y} > 1000$.

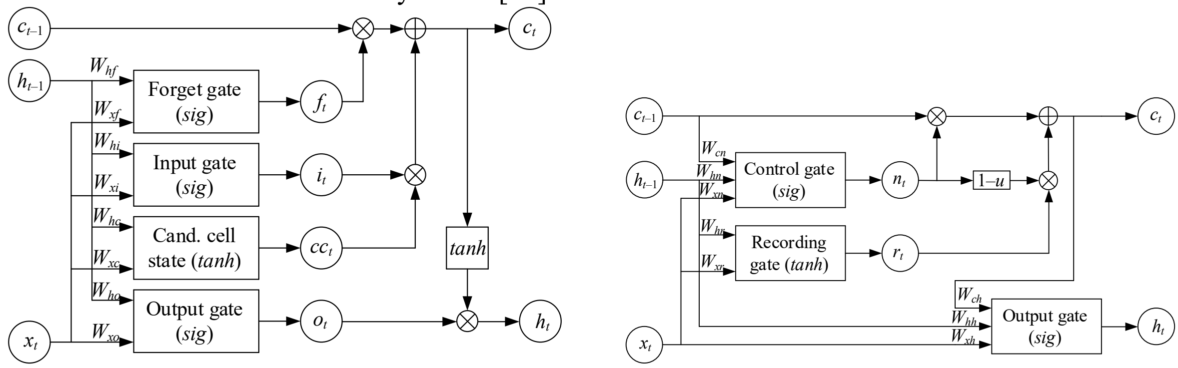
The adjective “deterministic” appearing in the name of the capacity means that the obtained capacity estimate is suitable for absolutely all possible input patterns that can be represented by N_x inputs. In reality, the distribution of input images, as a rule, has some regularity, which allows the neural network to generalize and thus increase the real capacity. Since the distribution of images, in the general case, is not known in advance, we can talk about such a capacity only conjecturally, but usually it is twice the deterministic capacity.

10.LSTM structure recurrent neural network modification

According to [23], in this work it is advisable to apply an LSTM network based on a dynamic model of a gas turbine engine based on the classical LSTM structure (fig. 5, a) or LSTM structure with variable memory (fig. 5, b) proposed by Georgy Makaryants and Alexander Kuznetsov. Each cell has one output neuron for predicting some parameter (for example, the gas generator r.p.m.). A collection of cells is a network for predicting multiple parameters. The main difference between LSTM networks and other recurrent networks is the memory tensor. A memory tensor is a variable, information into which can be record or erased during the operation of the network. The operation of the LSTM network (fig. 5) is based on the principle of memory tensor control using memorization and forgetting nodes:

$$c_t = f_t \circ c_{t-1} + i_t \circ cc_t; \tag{14}$$

where c_t – memory tensor representing a vector of weighted inputs in a step t; f_t – tensor at the exit from the forgetting node in the step t, representing the sum of the weighted inputs and outputs in the previous step $t-1$; c_{t-1} – step memory tensor $t-1$; i_t – tensor at the output of the input node at step t, which is the sum of the weighted inputs at the current step and outputs at the previous step; cc_t – candidate tensor to write to memory tensor [23].



a *b*

Figure 5: LSTM network diagram [23]: a – classical LSTM structure; b – LSTM structure with variable memory

As mentioned above, to solve the problem of dynamic monitoring of aircraft gas turbine engines, a special architecture of a recurrent neural network with long-short-term memory (LSTM) was developed, presented in [23]. When using LSTM networks, the problem of the vanishing gradient of the LSTM network arises, which the filter mechanism allows to resist. This mechanism makes it possible to regulate the flow of new information into the state vector c_t of the network, as well as the output of the state h_t of the network and updating its state c_t. The network filter vectors are determined according to the expressions [31]:

$$i_t = \sigma \cdot \left(U_i \cdot x_t + W_i \cdot h_{t-1}\right); \tag{15}$$

$$f_t = \sigma \cdot \left(U_f \cdot x_t + W_f \cdot h_{t-1}\right); \tag{16}$$

$$o_t = \sigma \cdot \left(U_o \cdot x_t + W_o \cdot h_{t-1}\right); \tag{17}$$

$$g_t = \sigma \cdot \left(U_g \cdot x_t + W_g \cdot h_{t-1}\right); \tag{18}$$

where $\sigma(z) = -0.5 + \dfrac{1}{1+e^{-z}}$ – sigmoidal function; $\tanh(z) = \dfrac{e^z - e^{-z}}{e^z + e^{-z}}$ – hyperbolic tangent function; x – input sequence; h – hidden state vector of the network cell; U_i, U_f, U_o, U_g, W_i, W_f, W_o, W_g – network filter weight matrices i, f, o, g; t – index of the element of the training sequence.

Based on the values of the network filters, its internal state (internal memory) and hidden state vectors are determined [31]:

$$c_t = tah\left(i_t \circ g_t + f_t \circ c_{t-1}\right); \tag{19}$$

$$h_t = o_t \circ c_t; \tag{20}$$

where t_c – vector of the internal state of the network cell; t_h – vector of the hidden state of the network cell; $\circ$ – elementwise product operation.

Expressions (15) – (20) correspond to the stage of direct signal propagation through the network. This mechanism allows the LSTM network to deal with the vanishing gradient problem when working with long sequences. By training filter parameters (U_i, U_f, U_o, U_g, W_i, W_f, W_o, W_g) LSTM network "tunes" its "memory". In this paper, to solve the problem of saturation of the activation function, it is proposed to use an activation function that does not saturate. This approach will allow make network training faster and more accurate. The activation function based on the logarithm, in contrast to [32], avoids saturation when processing large values and is defined by the expression:

$$f(x)=\begin{cases}-0.5+\ln(x+1),\ x\geq 0\\ -0.5-\ln(x-1),\ x<0\end{cases} \tag{21}$$

The advantage of the proposed activation function compared to the hyperbolic tangent function is its "unsaturation" and therefore its application will improve the efficiency of training the LSTM network. The sigmoid activation function is also subject to the saturation problem, but it is not possible to replace it with the proposed one, since it cannot serve as a gateway due to the fact that it does not scale the input values in the range from 0 to 0.5.

11.Formation of training and test subsets

The training set consists of a sufficient number of vectors (10000 vectors are enough for the task at hand), which are input data sets with the correct result. The sets included the main thermodynamic parameters of TV3-117 aircraft engine according to table 1 [33, 34]. Each of these parameters has been assigned a corresponding priority required for the functioning of the neural network. To form the training and test subsets in the work, cross-validation was used [35] to estimate the values of the parameters of TV3-117 aircraft GTE, the results of which are shown in fig. 6.

Table 1

Fragment of the training set

Engine unit	Parameter	1	2	3	4
Input device	ΔP^*_{in}	0.984	0.878	0.812	0.766
	ΔT^*_{in}	0.935	0.864	0.744	0.707
Compressor	ΔP^*_{comp}	0.961	0.902	0.853	0.781
	ΔT^*_{comp}	0.977	0.912	0.834	0.762
Combustion chamber	ΔP^*_{gas}	0.942	0.865	0.776	0.693
	ΔT^*_{gas}	0.954	0.875	0.762	0.689
Compressor turbine	$\Delta P^*_{comp.turb}$	0.988	0.931	0.902	0.894
	$\Delta T^*_{comp.turb}$	0.975	0.917	0.899	0.884
Free turbine	$\Delta P^*_{freeturb}$	0.933	0.898	0.835	0.808
	$\Delta T^*_{freeturb}$	0.946	0.909	0.844	0.801
Output device	ΔP^*_{out}	0.942	0.907	0.891	0.855
	ΔT^*_{out}	0.944	0.908	0.983	0.861

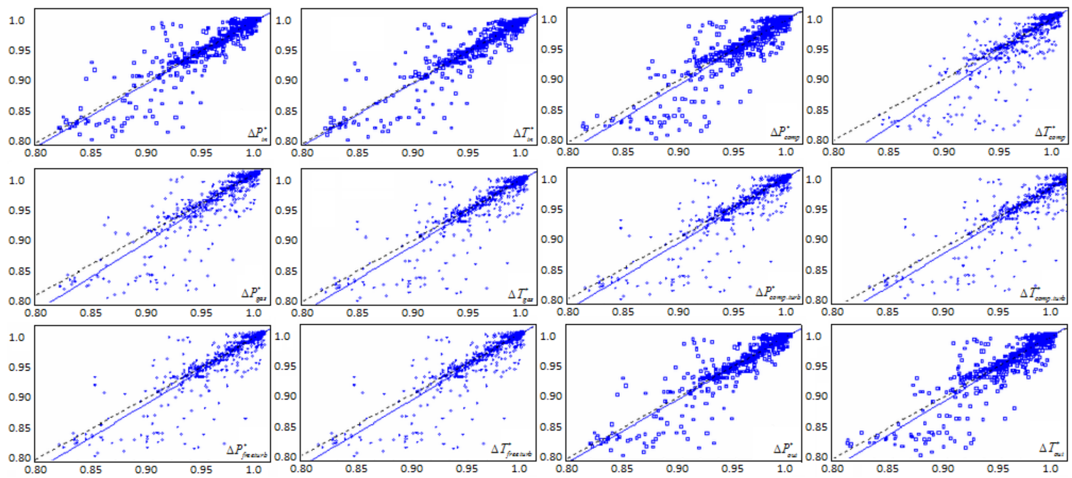


Figure 6: TV3-117 aircraft engine input parameter scatter diagram

12. Quality assessment of a trained neural network

As an example, a test version of a recurrent neural network was developed in the Matlab environment, illustrating the solution to the problem of TV3-117 aircraft GTE technical state control. The neural network was trained according to the following rule [22, 23]. First, all 10000 vectors of the training set were sequentially fed to the network input, with the help of which training was performed using the backpropagation algorithm. Each time, after training, a training test was performed for 2000 sets – the error was checked on 100 arbitrary sets of those already passed (test error control). Then, after completing the entire training, the error was checked for 1000 (10 % of the training set) – the stage of final control. At this stage, the erroneous estimate did not exceed the predetermined threshold of 1.2 % (fig. 7), which is a good result.

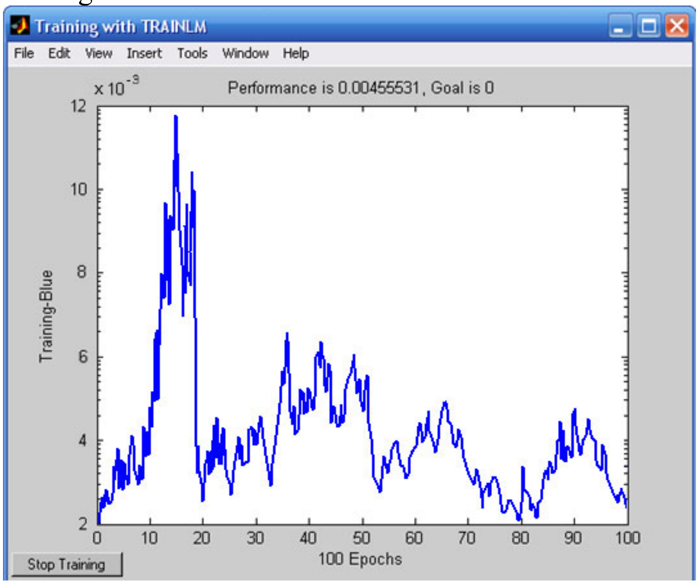


Figure 7: Neural network training results graph

In this case, [36] were used:

– Adam's algorithm as an optimization algorithm [37];

– Accuracy indicator as an objective function, where P – number of objects for which the neural network made the correct decision; N – number of objects in the training set.

Is the binary-crossentropy function that returns the classification error as the logistic loss function *Loss*:

$$Loss = -\frac{1}{N} y_i \log\left(\hat{y}_i\right) + \left(1 - \hat{y}_i\right) \log\left(1 - \hat{y}_i\right); \tag{22}$$

where y_i – true class label; $\hat{y}_i$ – response of the classifier (calculated class label) on the i-th object; N – number of classes.

To assess the quality of neural network training, various quality indicators can be used, in particular, such indicators as *Accuracy*, *Precision*, *Recall*, *F*-measure.

In the case of a binary classification based on the errors matrix of inaccuracies (errors), a 2×2 table can be drawn up (table 2), in which the following designations are used: *TP* – true-positive solution; *TN* – true negative decision; *FP* – false positive decision; *FN* – false negative decision.

Table 2

Errors matrix

Classification		Class labels in the dataset	
		Positive grade label	Negative grade label
Class labels exposed by the neural network	Positive grade label	*TP*	*TN*
	Negative grade label	*FP*	*FN*

The *Accuracy* indicator determines the proportion of objects for which the classifier made the right decision:

$$Accuracy = \frac{P}{N} = \frac{TP + TN}{FP + FN + TP + TN}; \tag{23}$$

The *Precision* indicator within a class determines the proportion of objects correctly assigned by the classifier to the class, to the total number of objects assigned by the classifier to the class in the training (test) sample. The higher the *Precision* score, the fewer false positives.

The *Recall* indicator within a class determines the proportion of objects correctly assigned by the classifier to the class to the number of objects of this class in the training (test) sample. The higher the value of the *Recall* indicator, the less false-negative decisions.

In the case of binary classification, *Precision* and *Recall* are defined as:

$$Precision = \frac{TP}{TP + FP}; \tag{24}$$

$$Recall = \frac{TP}{TP + FN}. \tag{25}$$

In the simplest case, the *F*-measure is defined as the harmonic average between *Precision* and *Recall*:

$$F = 2\frac{Precision \cdot Recall}{Precision + Recall}. \tag{26}$$

The results of training the neural network according to the *Accuracy* and *Loss* indicators are shown in fig. 9 and 10 respectively. As can be seen from fig. 8 and 9, the *Accuracy* indicator approaches one, and *Loss* indicator – tends to zero, which indicates the high accuracy of the model and its minimal error.

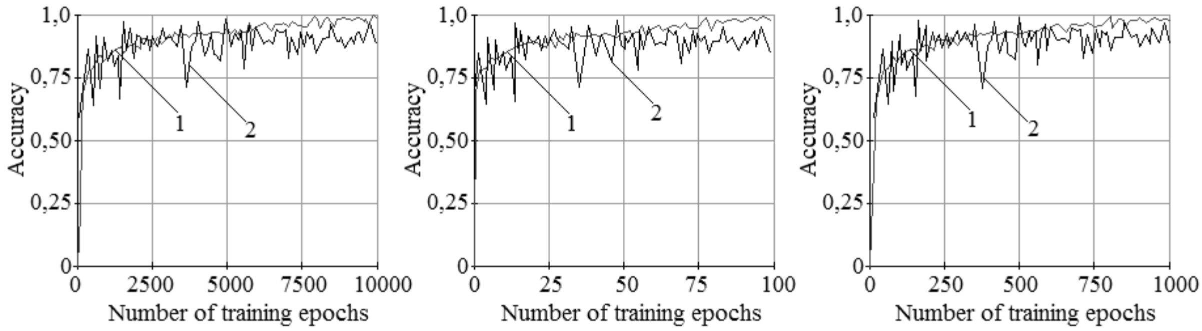


Figure 8: Model training results in terms of the *Accuracy* indicator: 1 – test; 2 – train

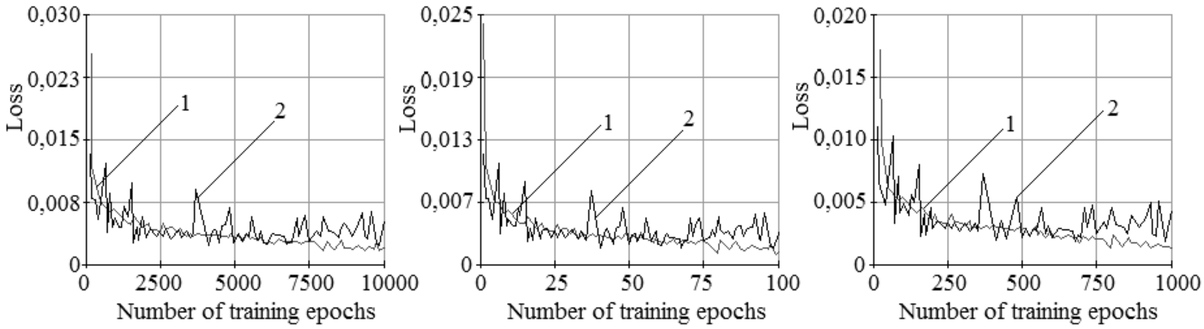


Figure 9: Model training results in terms of the *Loss* indicator: 1 – test; 2 – train

Table 3 shows the averaged values of the model learning outcomes, as well as the mean and variance values for the *Accuracy* indicator.

Table 3

Average values of neural network testing indicators

Accuracy	F-measure	Precision	Recall	Average time, s	Average Accuracy	Dispersion Accuracy
0.98867	0.97234	0.94617	1.0	1204.12	0.99011	0.0000085

13. Results and discussion

In fig. 10 shows a graph of the hypersurface in the space of the controlled parameters of helicopters gas turbine engine in flight modes (for example, the TV3-117 aircraft engine). As a result of its work, the trained neural network divided all the values of the degree of increase in the total pressure in the compressor fed to its inputs into 3 regions (fig. 10), corresponding to the serviceable (blue), faulty (red) and indefinite states where it is difficult to perform separation of parameter values due to their mutual overlap (green). The proposed neural network, integrated into the GTE control system, is capable of real-time correlating the value of the monitored parameter (the degree of increase in the total pressure in the compressor) with one of the areas and, if it enters the area of a faulty state, to give a signal about an incipient malfunction. This information (depending on the degree of danger) can be issued to the crew for timely adoption of the correct decision or to GTE automatic control system the power plant as a whole.

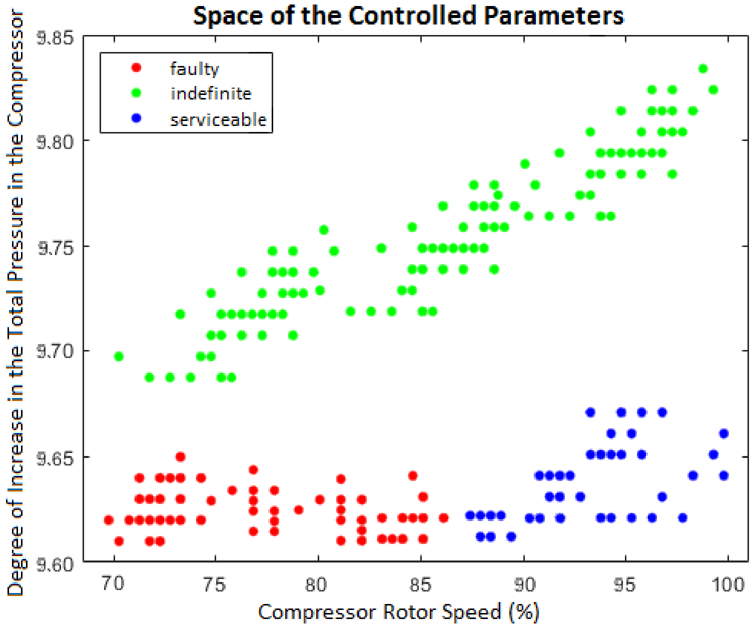


Figure 10: Results of TV3-117 aircraft engine technical state control in flight modes

Analogous research has been carried out using other neural network architectures (multilayer perceptron, convolutional neural network, Hopfield network, Kosko network, Jordan network, Elman network). It should be noted that the use of these neural network architectures did not allow to adequately classify the states of the TV3-117 aircraft engine, namely, the areas of serviceable, faulty and undefined states were not clearly distinguished. The results of a comparative analysis of the accuracy of solving the problem of control of helicopters aircraft GTE technical state in flight modes using the aforementioned neural network architectures are given in table 4.

Table 4

Results of comparative analysis of control accuracy

Neural network architecture	Absolute error (%)		
	Serviceable state	Faulty state	Indefinite state
Modified LSTM network	0.64	0.64	0.65
LSTM network	1.32	1.32	1.35
Multilayer perceptron	2.08	2.12	2.14
Convolutional neural network	16.35	17.48	19.24
Hopfield network	5.66	6.84	6.93
Kosko network	8.32	8.78	8.81
Jordan network	3.76	3.77	3.89
Elman network	3.35	3.35	3.46

Analysis of the table 4 shows that the use of the modified LSTM network makes it possible to solve the problem of control of helicopters aircraft GTE technical state in flight modes with maximum accuracy (the smallest error).

14. Conclusions

In this work, a methodology for control of helicopters aircraft engines technical state in flight modes has been developed, based on the operation of artificial neural networks and the transition to the given indicators of engine thermogasdynamic parameters. The scientific novelty of the result lies in the use of a neural network specially designed and trained on the originally formed a priori sets of engine thermogasdynamic parameters by a neural network, which makes it possible to increase the speed of systems and reduce the load on hardware resources, as well as to topologize the results of evaluating of engines technical state, which makes it possible to more clearly display problem engine nodes (input device, compressor, combustion chamber, compressor turbine, free turbine, output device) and simplify the optimization processes of solutions at a specific node (by reducing the number of engine thermogasdynamic parameters).

The proposed neural network was trained and tested on the practical task of control of TV3-117 aircraft engine technical state, based on the flight data of the Mi-8MTV helicopter. The testing error is calculated, which is no more than 1.2 % of the deviation from the a priori correct result on the vectors of the test set of sets. With an allowable value of 2 %, this allows us to speak about the efficiency of the method.

The use of such on helicopter board system, in contrast to the existing system for control the parameters of a gas turbine engine, will allow control of helicopters gas turbine engine technical state in flight in real time and recognize the failure at an early stage and inform the crew or the engineering staff about it. Becoming, which will be the guarantee of the correctness of the decision on the possibility of using all the potential of the gas turbine engine and will lead to an increase in the level of flight safety.

Thus, as can be seen from the results of experimental studies, recurrent neural networks demonstrate their high efficiency in solving the problem of monitoring the probable class of errors in the operation of equipment in complex dynamic systems (control of helicopters aircraft engines technical state in flight mode).

15. References

[1] B. Li, Y.-P. Zhap, 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
URL: https://www.sciencedirect.com/science/article/abs/pii/S1270963821005411

[2] S. N. Vassilyev, A. Yu. Kelina, Y. I. Kudinov, F. F. Pashchenko, Intelligent Control Systems, Procedia Computer Science, vol. 103 (2017) 623–628. 10.1016/j.procs.2017.01.088

[3] E. L. Ntantis, 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

[4] I. A. Krivosheev, K. E. Rozhkov, N. B. 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

[5] S. V. Zhernakov, A. T. Gilmanshin, New algorithms for on-board diagnostics of aircraft gas turbine engine based on fuzzy networks, Bulletin of USATU, vol. 19, no. 2 (68) (2015) 63–68.

[6] S. V. Zhernakov, Algorithms for control and diagnostics of an aviation gas turbine engine under conditions of onboard implementation based on neural network technology, Bulletin of USATU, vol. 14, no. 3 (38) (2010) 42–56.

[7] S. Kiakojoori, K. Khorasani, Dynamic neural networks for gas turbine engine degradation prediction, health monitoring and prognosis, Neural Computing & Applications, vol. 27, no. 8 (2015) 2151–2192.

[8] S. V. Enchev, S. S. Tovkach, Diagnostic of the Technical Condition Aviation Engines is Based on Fuzzy logic, Scientific Bulletin Kherson State Maritime Academy, no 1 (8) (2013) 216–224.

[9] V. I. Vasiliev, S. V. Zhernakov, Control and diagnostics of aircraft engine technical state based on expert systems, Bulletin of USATU, vol. 9, no 4 (22) (2007) 11–23.
[10] S. V. Zhernakov, R. F. Ravilov, Control and diagnostics of aircraft engine technical state based on the C-Priz expert system, Bulletin of USATU, vol. 16, no 6 (51) (2012) 3–11.
[11] Yu. M. Shmelov, S. I. Vladov, A. S. Khebda, K. G. Kotliarov, Application of the rules of urban logic in the problem of identification of the technical state of the aviation engine TV3-117, Scientific notes of Taurida National V.I. Vernadsky University. Series: Technical Sciences, vol. 30 (69), no 3 (2018) 34–40.
[12] A. M. Pashayev, D. D. Askerov, C. Ardil, R. A. Sadiqov, P. S. Abdullayev, Condition monitoring system of aircraft gas turbine engine complex, International Journal of Aerospace and Mechanical Engineering, vol. 1, no. 11 (2007) 689–695.
[13] S. Vladov, K. Kotliarov, S. Hrybanova, O. Husarova, I. Derevyanko, S. Gvozdik, Neuro-mechanical methods of control and diagnostics of the technical state of aircraft engine TV3-117 in film regions, Visnyk of Kherson National Technical University, no. 1 (72), part 1 (2020) 141–154.
[14] Y. Shmelov, S. Vladov, A. Kryshan, S. Gvozdik, Application of neural network technologies in the technical state control system of the aircraft engine TV3-117 in flight modes, Radiotekhnika, no. 194 (2018) 147–154.
[15] S. Vladov, Y. Shmelov, R. Yakovliev Control and Diagnostics of TV3-117 Aircraft Engine Technical State in Flight Modes Using the Matrix Method for Calculating Dynamic Recurrent Neural Networks. CMIS-2021: The Fourth International Workshop on Computer Modeling and Intelligent Systems, April 27, 2021, Zaporizhzhia, Ukraine. CEUR Workshop Proceedings (ISSN 1613-0073), vol. 2864 (2021) 97–109.
[16] S. Vladov, Yu. Shmelov, K. Kotliarov, S. Hrybanova, O. Husarova, I. Derevyanko, L. Chyzhova, Onboard parameter identification method of the TV3-117 aircraft engine of the neural network technologies, Transactions of Kremenchuk Mykhailo Ostrohradskyi National University, issue 5/2019 (118) (2019) 90–96. Doi: 10.30929/1995-0519.2019.5.90-96
[17] V. I. Vasiliev, S. V. Zhernakov, I. I. Muslukhov, Onboard algorithms for monitoring GTE parameters based on neural network technology, Bulletin of USATU, vol. 12, no. 1 (30) (2009) 61–74.
[18] F. A. Del Campo, M. C. Guevara Neri, O. O. V. Villegas, V. G. C. Sanchez, H. J. O. Domínguez, V. G. Jimenez, Auto-adaptive multilayer perceptron for univariate time series classification, Expert Systems with Applications, vol. 181 (2021) 115147.
[19] W. Guo, J. Zhao, J. Zhang, H. Wei, A. Song, K. Zhang, Stability analysis of opposite singularity in multilayer perceptrons, Neurocomputing, vol. 282 (2018) 192–201. Doi: 10.1016/j.neucom.2017.12.018
[20] D. Shi, B. Lam, K. Ooi, X. Shen, W.-S. Gan, Selective fixed-filter active noise control based on convolutional neural network, Signal Processing, vol. 190 (2021) 108317. Doi: 10.1016/j.sigpro.2021.108317
URL: https://www.sciencedirect.com/science/article/pii/S0165168421003546
[21] R. Shang, Y. Meng, W. Zhang, F. Shang, L. Jiao, S. Yang, Graph Convolutional Neural Networks with Geometric and Discrimination information, Engineering Applications of Artificial Intelligence, vol. 104 (2021) 104364. Doi: 10.1016/j.engappai.2021.104364
[22] Zgurovsky M., Sineglazov V. and Chumachenko E. (2021) Artificial Intelligence Systems Based on Hybrid Neural Networks: Theory and Applications, Publisher: Springer, 734 p.
[23] A. V. Kuznetsov, G. M. Makaryants, Gas turbine engine dynamic model based on variable-memory LSTM architecture, Vestnik of Samara University. Aerospace and Mechanical Engineering, vol. 19, no. 2 (2020) 38–52. Doi: https://doi.org/10.18287/2541-7533-2020-19-2-38-52
[24] Vladov S., Shmelov Yu. And Shmelova T. (2020) Control and diagnostics of TV3-117 aircraft engine technical state in flight modes using neural network technologies, Kremenchuk Novabook, 200 p.
[25] F. Cardin, A. Spiro, Pontryagin maximum principle and Stokes theorem, Journal of Geometry and Physics, vol. 142 (2019) 274–286. Doi: 10.48550/arXiv.1812.07875
[26] Z. Chen, S. Gan, X. Wang, A full-discrete exponential Euler approximation of the invariant measure for parabolic stochastic partial differential equations, Applied Numerical Mathematics, vol. 157 (2020) 135–158. Doi: 10.48550/arXiv.1811.01759

[27] S. Giuffre, Lagrange multipliers and non-constant gradient constrained problem, Journal of Differential Equations, vol. 269, issue 1 (2020) 542–562. Doi: 10.1016/j.jde.2019.12.013
[28] M. Meyer, D. Baldwin, Statistical learning of action: The role of conditional probability, Learning & Behavior, vol. 39 (2011) 383–398. Doi: 10.3758/s13420-011-0033-7
[29] Bodyansky E. V. and Rudenko O. G. (2004) Artificial neural networks: architectures, training, applications. Kharkiv, Teletech, 369 p.
[30] A. Glushchenko, V. Petrov, K. Lastochkin, Backpropagation method modification using Taylor series to improve accuracy of offline neural network training, Procedia Computer Science, vol. 186 (2021) 202–209. Doi: 10.1016/j.procs.2021.04.139
[31] I. Chernobaev, A. Surkova, A. Pankratova, Modelling of texts using recurrent neural networks, Proceedings of Nizhny Novgorod State Technical University n.a. R.E. Alekseev, no 1 (120) (2018) 65–73.
[32] S. Gluge, R. Bock, G. Palm, A. Wendemuth, Learning long-term dependencies in segmented-memory recurrent neural networks with backpropagation of error, Neurocomputing, vol. 141 (2014) 54–64. Doi: 10.1016/j.neucom.2013.11.043
[33] S. I. Vladov, N. V. Podgornykh, V. Ya. Teleshun, Mathematical model of TV3-117 aircraft engine compressor for its control and diagnostics of a technical state in the conditions of onboard operation of the aircraft, The path to success and prospects for development (to the 26th anniversary of the Kharkiv National University of Internal Affairs) : proceedings of the international scientific-practical conference, November 20, 2020, Kharkiv 112–116.
[34] R. R. Mukhamedov, GTE mathematical models, Youth Bulletin of USATU, no 1 (10) (2014) 35–43.
[35] J. Wainer, G. Cawley, Nested cross-validation when selecting classifiers is overzealous for most practical applications, Expert Systems with Applications, vol. 182 (2021) 115222. doi: 10.1016/j.eswa.2021.115222
[36] L. Demidova, D. Marchev, Application of recurrent neural networks in the classification problem of failures in the complex technical systems within the framework of proactive maintenance, Vestnik of RSREU, no 69 (2019) 135–148. doi: 10.21667/1995-4565-2019-69-135-148
[37] D. P. Kingma, J. L. Ba, Adam: A Method for Stochastic Optimization, 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 07–09, 2015. 15 p. URL: https://arxiv.org/pdf/1412.6980.pdf