Modified Helicopters Turboshaft Engines Neural Network On-board Automatic Control System Using the Adaptive Control Method

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

The work is devoted to the modification of helicopters turboshaft engines onboard automatic control system of through the introduction of an adaptive control unit into it, which consists of a reference engine model module and a signal adaptation module. The real-time identification method for helicopters turboshaft engines onboard automatic control system adaptive control subsystem has been modified, which allows you to set the desired system response to a disturbance for its current state. The implementation of the proposed solutions is carried out using the NEWFF multilayer neural network, which made it possible to significantly reduce the errors of the first and second kind in comparison with the tolerance control method. The results of the experiment – initial and secondary testing of helicopters aircraft engines automatic control system with signal tuning units and a reference model showed an improvement in the quality of transient recognition compared to the use of standard controllers.

Keywords 1

Turboshaft engines, neural network, automatic control system, signal adaptation module, reference model module

1. Introduction

The ever-increasing requirements for helicopters tactical performance, the complication of their flight conditions make it necessary to improve the characteristics of turboshaft engines (TE), to ensure the stable operation of TE in a wide range of operating modes. Distinctive features of modern helicopters TE are the need for simultaneous control of several output parameters at once, a wide range of changes in dynamic characteristics, changes in the qualitative and quantitative composition of control subsystems during operation, non-linearity and non-stationarity of engine characteristics. All this inevitably leads to a significant complication of the laws of helicopters TE automatic control, and, as a result, to the complication of their automatic control systems (ACS), with a simultaneous increase in the requirements for the quality and reliability of their operation, ease of use, etc.

One of the new promising directions in the field of complex dynamic objects automatic control is the use of intelligent control systems based on artificial neural networks (ANN). The main advantage of these control systems is the use of such properties of ANN as the ability to approximate arbitrary nonlinear dependencies (for which they are often called "universal approximators"), the ability to learn, high speed due to the parallel nature of the network itself, potentially higher noise immunity and fault tolerance.

At the same time, the analysis of modern literature on ANN and ANN control systems shows that, despite the ongoing active developments in this area, many issues related to the development of algorithms and methods for identifying nonlinear objects based on ANN models have not yet been resolved, synthesis

ITTAP'2022: 2nd International Workshop on Information Technologies: Theoretical and Applied Problems, November 22–24, 2022, Ternopil, 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)

of the structure and algorithms for adapting (training) the parameters of ANN controllers, features of their implementation in multi-mode control systems for nonlinear dynamic objects. All of the above fully applies to such a dynamically complex class of control objects as helicopters TE.

2. Related Works

2.1. Literature review

The literature describes numerous examples of the practical application of ANN for solving problems of controlling an aircraft [1], a car [2], a mining process [3], an engine shaft speed [4], an electric furnace [5], a turbogenerator [6], a welding machine [7], pneumatic cylinder [8].

In the course of the development of neurocontrol, various methods for constructing neurocontrollers using various types of neural networks were studied: linear Adalina type [9], multilayer perceptrons [10], recurrent networks (RNN) [11], radial basis functions (RBF) networks [12], etc. The best results obtained using multilayer perceptrons with delay lines [13]. Two main directions have been formed in the application of ANN inside synthesized controllers: direct methods based on direct control of an object using an ANN, and indirect methods, when a neural network is used to perform auxiliary control functions, such as noise filtering or dynamic object identification. Depending on the number of ANN that make up the neurocontroller, neurocontrol systems can be single-module or multi-module. Neurocontrol schemes that are used in conjunction with traditional controllers are called hybrid.

The key problem in solving problems of control of dynamic objects is the implementation of the model of the inverse dynamics of the controlled object. An analytical solution to this problem is not always possible, since it requires the inversion of cause-and-effect dependencies of the behavior of a real object. The use of neural networks makes it possible to find approximate solutions to this problem by ANN training on examples of controlling a real object. When using direct methods of neurocontrol, in particular, in the method of generalized inverse neurocontrol [14], this is achieved by directly ANN training using examples of the behavior of the controlled object. However, the sequences of examples used for such training, obtained by inverting the results of observing real objects, often contain contradictions that drastically reduce the quality of ANN training. A number of methods have been proposed to solve this problem. In the method of specialized inverse neurocontrol [14, 15] and some versions of adaptive criticism systems [16], the problem of training inverse dynamics is solved by approximating the analytical model of the controlled object and calculating the local values of the Jacobian for different regions of the state space. In the method of error backpropagation through a direct neuroemulator, to form a linearized model of the inverse dynamics of an object, the usual error backpropagation scheme is used, which is used to train multilayer perceptrons. In multimodule neurocontrol systems, the same problem is solved by dividing the object state space into local areas in which inverse models are represented by single-valued functions. For each such area, a separate neural module is allocated [17]. Perspective for modeling inverse dynamics may be new types of neural networks that allow modeling multivalued functions, in particular, Bishop's probabilistic networks based on mixtures of Gaussian models (Mixture Density Networks) [18].

2.2. Research problem statement

The goal of the study is to develop and improve algorithms and methods for ANN control of helicopters TE and their elements, synthesis and training of multi-mode ANN controllers of helicopters TE, as well as the implementation of the proposed neural network control algorithms in real time.

3. Proposed technique

3.1. Generalized structure of helicopters turboshaft engines control system

In neurocontrol tasks, to represent the control object (helicopters TE), a black box model (fig. 1) is used, in which the current input and output values are observable.

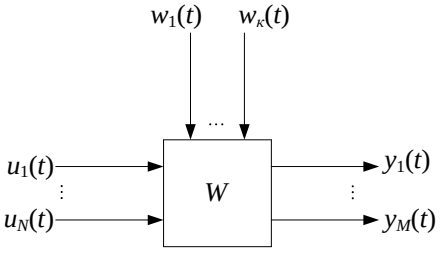


Figure 1: Aircraft engine model in the form of a black box

Helicopters TE operational mode is considered inaccessible to external observation, although the dimension of the state vector is usually considered fixed. The dynamics of helicopters TE behavior can be represented in a discrete form:

$$S(k+1)=\Phi(S(k),u(k)); \tag{1}$$

$$y(k+1)=\Psi(S(k)); \tag{2}$$

where $S(k)\in\Re^N$ – N-dimensional vector value of helicopters TE operational mode on the k-th cycle; $u(k)\in\Re^P$ – P-dimensional control vector value; $y(k+1)\in\Re^V$ – V-dimensional output value of helicopters TE at cycle k + 1. The general control diagram of helicopters TE as a dynamic object is shown in fig. 2.

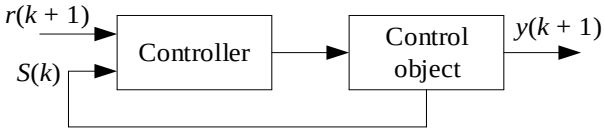


Figure 2: General feedback control diagram

To estimate the operational mode vector of helicopters TE as a dynamic object of order, the model of non-linear autoregression with additional input signals (NARX) [19] can be used:

$$S(k)=\begin{pmatrix} y(k) \\ y(k-1) \\ \dots \\ y(k-N) \\ y(k-1) \\ \dots \\ y(k-Q) \end{pmatrix}. \tag{3}$$

In practice, this ratio is usually used without retrospective control inputs:

$$S(k)=\begin{pmatrix} y(k) \\ y(k-1) \\ \dots \\ y(k-N) \end{pmatrix}. \tag{4}$$

Helicopters TE operational mode as a dynamic object can also be represented by a snapshot of its phase trajectory:

$$S(k)=\begin{pmatrix} y(k) \\ y(k)' \\ \dots \\ y(k)^{(N)} \end{pmatrix}. \tag{5}$$

In the diagrams, the TDL (Tapped Delay Line) module is used to input delayed feedback data into the controller.

3.2. Choosing of neurocontrol optimal type

The main types of neurocontrol are systematized by Artem Chernodub and Dmitry Dzyuba, researchers at the Institute of Mathematical Machines & Systems Problems, which are described in detail in [20], namely:

1. Imitative neurocontrol (Neurocontrol learning based on mimic, Controller Modeling, Supervised Learning Using an Existing Controller) [21], covering neurocontrol systems in which the neurocontroller is trained on examples of the dynamics of a conventional feedback controller, built, for example, on the basis of the usual proportional-integral-differential (PID) control diagram.

2. Inverse neurocontrol, in which the formation of an inverse model of the control object is carried out by ANN training. There are several types of such neurocontrol:

2.1. Generalized Inverse Neurocontrol (Direct Inverse Neurocontrol) [22] provides for off-line network training based on the recorded behavioral trajectories of a dynamic object.

2.2. Specialized Inverse Neurocontrol [22] makes it possible to train an inverse neurocontroller online using the deviation error of the object position from the setpoint $e = r - y$.

2.3. Backpropagation Through Time (Internal Model Control) method [23, 24] is based on the idea of using a tandem of two ANN, one of which performs the function of a controller, and the other is a direct neuroemulator that is trained to model the dynamics of the control object.

3. Predictive neurocontrol. The method of training neurocontrollers, which minimizes the deviation of the current position of the control object from the setpoint for each cycle, does not always provide the best integral quality of control. There are such types of predictive neurocontrol:

3.1. Predictive model neurocontrol (NN Predictive Control, Model Predictive Control, Neural Generalized Predictive Control) [25] minimizes the cost functional of the integral error predicted for $L = \max(L_1, L_2)$, $0 \le L_1 \le L_2$ cycles ahead $Q(k) = \sum_{0}^{L_2} e(k+i)^2 + \rho \sum_{0}^{L_2} \left(u(k+i) - u(k+i-1)\right)^2$, where e – system output error, ρ – contribution of the change in the control signal to the total cost functional Q. The remarkable thing about this method is that it does not have a trainable neurocontroller. Its place is taken by a real-time optimization module, in which the simplex method [26] or the Quasi-Newton algorithm [27] can be used.

3.2. Neurocontrol methods based on adaptive criticism (Adaptive Critics), also known as Approximate Dynamic Programming (ADP), have been very popular in recent years [28]. The criticism module performs an approximation of the values of the cost function. The popularity of adaptive criticism systems is explained by the presence of a developed theoretical base in the form of Bellman's theory of dynamic programming, as well as their ability to converge to optimal or close to optimal control [29].

4. Multi-module neurocontrol. Multi-module neurosystems, built according to the type of expert committees, have become widely used in recognition systems, and later they gave impetus to the development of multi-module neurocontrol systems. Within the framework of a multi-module approach, the original task is divided into separate subtasks, which are solved by separate modules. The final decision is made by the gateway network based on the private decisions of the expert modules.

4.1. Multimodule neurocontrol systems based on local inverse models (Incremental Clustered Control Networks) [30] consist of a set of linear neurocontrollers and a gateway module. The disadvantage of this method is the need for a large number of examples for training neurocontrollers distributed in all areas of the state space of the controlled object.

4.2. Multimodule neurocontrol method based on pairs of direct and inverse models (Multiple Paired Forward and Inverse Models, Multiple Switched Models) [31, 32]. Unlike the method of neurocontrol based on local inverse models, in which the behavior of the system is formed during training and is not corrected during control, this method provides for the correction of the behavior of neural modules at each step of neurocontrol.

5. Hybrid neurocontrol. Hybrid neurocontrol systems are called, in which neural networks work together with conventional controllers, PID-controllers or other types of controllers. Hybrid neuro-PID

control (NNPID Auto-tuning, Neuromorphic PID Self-tuning) [33, 34] allows self-tuning of the PID controller online using neural networks.

Hybrid parallel neurocontrol represents a compromise solution for the introduction of neurocontrol in the industry and the transition from conventional controllers to ANN. Thus, taking into account the analysis of existing types of neurocontrol, in the problem to be solved for controlling helicopters TE in flight modes, a hybrid neuro-PID control is applied, in which the control signal generated by the controller is a weighted sum of proportional, integral and differential parts [35]:

$$u(t) = K_1 e(t) + K_2 \int_0^t e(\tau)\, d\tau + K_3 \frac{de(t)}{dt}. \quad (6)$$

The coefficients K_1, K_2, K_3 are obtained by tuning the PID-controller, which can be performed manually according to the Ziegler-Nichols rule, the Cohen-Kuhn rule, or other methods [36], or using ANN (fig. 3).

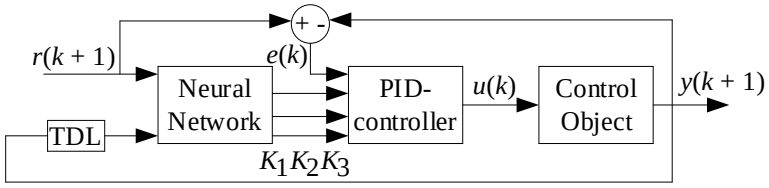


Figure 3: Hybrid neuro-PID control diagram [20]

The trained neurocontrol system operates as follows. At step k, the neural network receives the setpoint $r(k + 1)$ and generates the PID-controller control coefficients K_1(k), K_2(k), K_3(k), which are fed to the PID-controller along with the value of the current feedback error $e(k) = r\left[(k+1), r(k), ..., r(k-N+1)\right]^T$. The PID-controller calculates the control signal $u(k)$ according to the expression:

$$u(k) = u(k-1) + K_1(k)\big(e(k) - e(k-1)\big) + K_2(k)e(k) + K_3\big(e(k) - 2e(k-1) + e(k-2)\big); \quad (7)$$

used for discrete PID-controllers and feeds it to the control object.

6. Neural control with a reference model (Model Reference Adaptive Control, Neural Adaptive Control) is a variant of neurocontrol using the method of error back propagation through a direct neuroemulator, with an additional reference model (Reference Model) embedded in the circuit. This is done in order to increase the stability of the transient process: in the case when the transition of the object to the target position in one cycle is impossible, the trajectory of movement and the time of the transient process become poorly predictable values and can lead to undesirable modes of operation of the system.

3.3. Proposed neurocontrol method

Since, in the study of helicopters TE dynamic characteristics, the trajectory of movement and the time of the transition process are poorly predictable values, since they depend on many external and internal factors, which will lead to undesirable operating modes of the systems, this paper proposes a new, combined method of Hybrid neuro-PID control with a reference model (fig. 4).

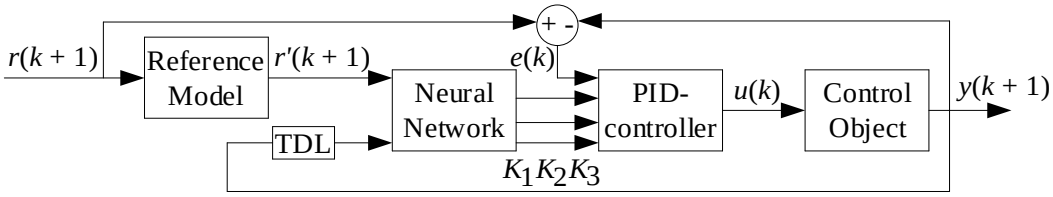


Figure 4: Hybrid neuro-PID control diagram with a reference model

3.4. Modification of helicopters turboshaft engines controlling method at flight mode

The method of complex dynamic objects adaptive control (on the example of a ground-based gas turbine plant) was developed by Ivan Bakhirev. In that work, a modification of this method and its

adaptation to helicopters TE is made. The system of differential equations, taking into account the instability of properties, representing the operation of helicopters TE in an arbitrary mode (nominal, I cruising, II cruising, emergency, idle rating) will have the form:

$$\dot{\mathbf{x}} = F\left(\mathbf{x}, \mathbf{u}, \boldsymbol{\xi}, \mathbf{f}, ...\right); \;\; \mathbf{x}\left(t_0\right) = \mathbf{x_0}; \tag{8}$$

where $\mathbf{x} = \mathbf{x}(t)$ – n-dimensional function of the state of the system; $\mathbf{u} = \mathbf{u}(t)$ – m-dimensional function of control actions; $\boldsymbol{\xi}$ – vector of limited dimension of changing parameters; $\mathbf{f} = \mathbf{f}(t)$ – n-dimensional function of external perturbations; $\mathbf{x_0}$ – initial state.

Let us represent a non-stationary nonlinear model of helicopters TE in the following form [37]:

$$\dot{\mathbf{x}} = \mathbf{A}\left(\mathbf{x}, t\right)\mathbf{x} + \mathbf{B}\left(\mathbf{x}, t\right)\mathbf{u} + \mathbf{f}\left(t\right); \tag{9}$$

where $\mathbf{A}\left(\mathbf{x}, t\right) = \mathbf{A}\left(\boldsymbol{\xi}\left(\mathbf{x}, t\right)\right)$, $\mathbf{B}\left(\mathbf{x}, t\right) = \mathbf{B}\left(\boldsymbol{\xi}\left(\mathbf{x}, t\right)\right)$ – functional matrices of appropriate sizes. The pair (**A**, **B**) has the controllability property. The description of the boundaries of changes in the elements of matrices (**A**, **B**) must accompany the expression (8).

This model consists of a linear model of helicopters TE [38] combined with nonlinear dependences obtained experimentally [39]. When operating in the rotation speed stabilization mode, regulators with the following transfer functions $W_{FT}\left(p\right) = k_p \dfrac{k_i + k_f p}{k_i + p}$, $W_G\left(p\right) = k_D \dfrac{1 + T_D p}{p}$, corresponding to the free turbine speed regulator (n_{FT}) and the gas dispenser regulator, are used, while T_D – regulator time constant, k_D – gas dispenser regulator gain. Regulator settings correspond to the TE operation mode and its rating. The gas dispenser regulator is switched on at the output of the control device, it is he who generates the signal for the fuel supply G_T. All other selective ACS controllers are connected to this controller [40]. Therefore, the sequential inclusion of these two regulators means that at the moment it is the free turbine speed circuit that is closed to the gas dispenser, and it is it that is currently active.

The control vector $\mathbf{u} = \left[u_1, u_2\right]^T$ is included in the extended state vector of equation (9): $\mathbf{x} = \left[x_1, x_2, x_3, x_4\right]^T$, where: $x_1 = n_{FT}$, $x_2 = n_{TC}$, x_3 – output of the gas dispenser regulator integrator, x_4 – output of n_{FT} regulator integrator. Then expression (9) can be written taking into account the form of functional matrices $\mathbf{A}(\mathbf{x}, t)$, $\mathbf{B}(\mathbf{x}, t)$ in the following form:

$$\dot{\mathbf{x}} = \begin{pmatrix} 0 & a_{12}\left(x_1, x_2, t\right) & 0 & 0 \\ a_{21}\left(x_2, x_3, t\right) & a_{22}\left(x_2, t\right) & a_{23}\left(x_2, x_3, t\right) & a_{24}\left(x_2, x_3, t\right) \\ a_{31} & 0 & 0 & a_{34} \\ a_{41} & 0 & 0 & a_{44} \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & b_{24}\left(x_2, x_3, t\right) \\ 0 & 0 & 0 & b_{34} \\ 0 & 0 & 0 & b_{44} \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 0 \\ 0 \\ g \end{pmatrix} + \begin{pmatrix} f_1\left(t\right) \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

where $f_1\left(t\right) = k_{CT}\left(x_1, t\right) N_G\left(t\right)$, and the maximum multiplicity of measurements of the coefficients, respectively, $a_{12} = 20…25$, $a_{22} = 1.5…3.0$, $a_{23} = 5…7$, $a_{24} = 5…7$, $b_{24} = 5…7$.

Let us single out the linear stationary part on the right side of (9) so that the description takes the following form [37]:

$$\dot{\mathbf{x}} = \mathbf{A}_0\mathbf{x} + \mathbf{B}_0\mathbf{g} + \boldsymbol{\sigma}_{\varphi}; \tag{10}$$

where $\boldsymbol{\sigma}_{\varphi} = \mathbf{F}\left(\mathbf{x}, \mathbf{u}, \boldsymbol{\xi}, \mathbf{f}, t\right) - \mathbf{A}_0\mathbf{x} - \mathbf{B}_0\mathbf{u}$ – non-linear non-stationary part; $\mathbf{x}$ – four-dimensional state vector; $\mathbf{g}$ – four-dimensional vector of setting actions; $\mathbf{A}_0$, $\mathbf{B}_0$ – (4×4)-dimensional constant matrices corresponding to the linear stationary part, which are an approximation obtained by averaging and linearizing matrix elements in time, or designate the desired behavior of the object. Then we consider the linear stationary part as the reference model: $\mathbf{A}_0 = \mathbf{A}_м$, $\mathbf{B}_0 = \mathbf{B}_м$, where $\mathbf{A}_м$ – Hurwitz matrix (stable).

Thus, equation (10) can be written as:

$$\dot{\mathbf{x}} = \begin{pmatrix} 0 & a_{12} & 0 & 0 \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & 0 & 0 & a_{34} \\ a_{41} & 0 & 0 & a_{44} \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & b_{24} \\ 0 & 0 & 0 & b_{34} \\ 0 & 0 & 0 & b_{44} \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 0 \\ 0 \\ g \end{pmatrix} + \begin{pmatrix} \sigma_{\varphi_1}\left(x_1, x_2, t\right) \\ \sigma_{\varphi_2}\left(x_2, x_3, t\right) \\ 0 \\ 0 \end{pmatrix}.$$

Expression (9) is also supplemented by the adaptive controller equation [41] in the following form:

$$\mathbf{u} = U\left(\mathbf{x}, \mathbf{K}, \mathbf{z}, \mathbf{g}\right); \tag{11}$$

where $\mathbf{g} = \mathbf{g}(t)$ – m-dimensional vector of the reference signals, in this problem, the signal of the free turbine rotation speed n_{STset} recorded on board the helicopter; $\mathbf{K} = \mathbf{K}(t)$ – matrix of adjustable parameters, responsible for parametric setting (PS); $\mathbf{z} = \mathbf{z}(t)$ – m-vector of additional (signal) influences, responsible for the signal setting (SS) [37]. Let's set the reference model in the form:

$$\dot{\mathbf{x}} = \mathbf{A}_{\mathbf{M}}\mathbf{x}_{\mathbf{M}} + \mathbf{B}_{\mathbf{M}}\mathbf{g} + \mathbf{f}\left(t\right). \tag{12}$$

The set control problem is reduced to synthesizing the control law $\mathbf{u}(t)$, which should be expressed in terms of minimizing the quality functional on the solutions of system (10), (12), and also ensuring the following inequality is satisfied for any $\xi \in M, \mathbf{x}\left(t_0\right), \mathbf{x}_{\mathbf{M}}\left(t_0\right)$

$$\left\|\mathbf{x}\left(t\right) - \mathbf{x}_{\mathbf{M}}\left(t\right)\right\| = \left\|\mathbf{e}\left(t\right)\right\| \le \varepsilon_0 \tag{13}$$

for any $t \ge t_a$, $t_a = t_0 + \theta_a$, $t_0 \ge 0$, where θ_a – adaptation process time, or limiting ratio

$$\lim_{t\to\infty}\left\|\mathbf{e}\left(t\right)\right\| = 0. \tag{14}$$

We write the right side of (8) as

$$\mathbf{F}\left(\mathbf{x}, \mathbf{u}, \xi, t\right) = \left(\mathbf{A}_t + \mathbf{a}\left(\mathbf{x}, \xi\right)\mathbf{x} + \mathbf{B}_t + \mathbf{b}\left(\mathbf{x}, \xi\right)\right)\mathbf{u}; \tag{15}$$

where $\mathbf{a}(\mathbf{x}, \xi)$, $\mathbf{b}(\mathbf{x}, \xi)$ – some non-linear additions; $\mathbf{A}_t = \mathbf{A}(\xi)$, $\mathbf{B}_t = \mathbf{B}(\xi)$. When adding additives to (15), the functions $\mathbf{a}(\mathbf{x}, \xi)$, $\mathbf{b}(\mathbf{x}, \xi)$ are differentiable and continuous in their arguments.

Assuming in (10) $\mathbf{A}_0 = \mathbf{A}_{\text{м}}$; $\mathbf{B}_0 = \mathbf{B}_{\text{м}}$, taking into account (15), the corresponding expression takes the form

$$\mathbf{F}\left(\mathbf{x}, \mathbf{u}, \xi, t\right) - \mathbf{A}_{\mathbf{M}}\mathbf{x} - \mathbf{B}_{\mathbf{M}}\mathbf{u} = \left(\mathbf{A}_t - \mathbf{A}_{\mathbf{M}}\right)\mathbf{x} + \left(\mathbf{B}_t - \mathbf{B}_{\mathbf{M}}\right)\mathbf{u} + \mathbf{a}\mathbf{x} + \mathbf{b}\mathbf{u} = \boldsymbol{\sigma}_{\varphi}; \tag{16}$$

where $\boldsymbol{\sigma}_{\varphi} = \boldsymbol{\sigma} + \varphi$; $\boldsymbol{\sigma} = \left(\mathbf{A}_t - \mathbf{A}_{\mathbf{M}}\right)\mathbf{x} + \left(\mathbf{B}_t - \mathbf{B}_{\mathbf{M}}\right)\mathbf{u}$; $\varphi = \mathbf{a}\mathbf{x} + \mathbf{b}\mathbf{u}$. The adaptive controller (11) is represented as:

$$\mathbf{u}\left(t\right) = \mathbf{K}_a\mathbf{x} + \mathbf{K}_b\left(\mathbf{g} + \mathbf{z}\right); \tag{17}$$

where $\mathbf{K}_a$, $\mathbf{K}_b$ – $(m\times n)$, $(m\times m)$-dimensional matrices of adjustable coefficients. In this case, (10), (11), taking into account (18) and (19), can be represented as

$$\dot{\mathbf{x}} = \mathbf{A}_{\mathbf{M}}\mathbf{x} + \mathbf{B}_{\mathbf{M}}\mathbf{g} + \left(\mathbf{A}_t + \mathbf{B}_t\mathbf{K}_a - \mathbf{A}_{\mathbf{M}}\right)\mathbf{x} + \left(\mathbf{B}_t\mathbf{K}_b - \mathbf{B}_{\mathbf{M}}\right)\mathbf{g} + \mathbf{B}_t\mathbf{z} + \varphi; \tag{18}$$

where $\varphi = \mathbf{f} + \left(\mathbf{a} + \mathbf{b}\mathbf{K}_a\right)\mathbf{x} + \mathbf{b}\mathbf{K}_b\mathbf{g} + \mathbf{b}\mathbf{z}$.

Then the goal of adaptation (14) is achieved by fulfilling the ratio:

$$\lim_{t\to\infty}\left(\mathbf{A} + \mathbf{B}_t\mathbf{K}_a\right) = \mathbf{A}_{\mathbf{M}}; \ \lim_{t\to\infty}\mathbf{B}_t\mathbf{K}_b = \mathbf{B}_{\mathbf{M}}; \ \mathbf{B}\mathbf{z} = \varphi. \tag{19}$$

Assuming that, $\mathbf{K}_a \to \mathbf{K}_a^0$, $\mathbf{K}_b \to \mathbf{K}_b^0$, $\mathbf{A} + \mathbf{B}\mathbf{K}_a^0 = \mathbf{A}_{\mathbf{M}}$, $\mathbf{B}\mathbf{K}_b^0 = \mathbf{B}_{\mathbf{M}}$, as a result of adaptation, expression (18) can be rewritten in the form

$$\dot{\mathbf{x}} = \mathbf{A}_{\mathbf{M}}\mathbf{x} + \mathbf{B}_{\mathbf{M}}\mathbf{g} + \mathbf{B}\boldsymbol{\delta}_a\mathbf{x} + \mathbf{B}\boldsymbol{\delta}_b\mathbf{g} + \mathbf{B}\mathbf{z} + \varphi; \tag{20}$$

where $\mathbf{K}_a - \mathbf{K}_a^0 = \boldsymbol{\delta}_a$; $\mathbf{K}_b - \mathbf{K}_b^0 = \boldsymbol{\delta}_b$.

The goal of control (19) is achieved using the following equations of adaptation algorithms, presented in general form [37]:

$$\dot{\mathbf{K}} = \mathbf{A}_1\left(\mathbf{K}, \mathbf{e}, \mathbf{g}\right); \ \mathbf{K} = \left[\mathbf{K}_a; \mathbf{K}_b\right]; \tag{21}$$

$$\mathbf{z} = \mathbf{A}_2\left(\mathbf{e}, \mathbf{g}, \xi, t\right); \tag{22}$$

of these, the parametric setting (PS) corresponds to the first equation (21), and the signal setting (SS) corresponds to the second (22).

In [37], a recommendation is given and substantiated with a significant predominance of non-stationary properties to use parametric tuning for model (8). In case of predominance of non-linear properties in the system, use the signal setting for model (8). In the case of manifestation of both nonlinear and non-stationary properties of the model (8), use both types of tuning together.

When constructing a system by the Lyapunov function method [37, 42], with a reference model and with a combined setting (PN + SN), using equations (12), (18), we represent the error equation in the form

$$\dot{\mathbf{e}} = \mathbf{A}_{\mathbf{M}}\mathbf{e} + \mathbf{B}_t\boldsymbol{\delta}\mathbf{v}\left(t\right) + \mathbf{B}_t\mathbf{z}\left(t\right) + \varphi. \tag{23}$$

The Lyapunov function is chosen in the form [37]:

$$\mathbf{V}(\mathbf{e},\boldsymbol{\delta}) = \frac{1}{2}\left(\mathbf{e}^T\mathbf{P}\mathbf{e} + Tr\left(\boldsymbol{\delta}\boldsymbol{\Gamma}^{-1}\boldsymbol{\delta}^T\right)\right). \quad (24)$$

To find the matrix **P**, it is necessary to solve the matrix equation

$$\mathbf{P}\mathbf{A}^T + \mathbf{A}^T\mathbf{P} = -\mathbf{Q}; \quad (25)$$

where the matrix **Q** is chosen arbitrarily, and its determinant must be greater than zero.

When the condition of quasi-stationarity is met (limited rate of change of TE parameters), the time derivative of this function can be written as

$$\dot{\mathbf{V}}(\mathbf{e},\boldsymbol{\delta}) = \frac{1}{2}\mathbf{e}^T\left(\mathbf{A}_{\mathbf{M}}^T\mathbf{P}\mathbf{A}_{\mathbf{M}}\right)\mathbf{e} + \mathbf{e}^T\mathbf{P}\mathbf{B}_t\boldsymbol{\delta}\mathbf{v} + \mathbf{e}^T\mathbf{P}\boldsymbol{\varphi} + Tr\left(\dot{\boldsymbol{\delta}}\boldsymbol{\Gamma}^{-1}\dot{\boldsymbol{\delta}}^T\right) + \mathbf{e}^T\mathbf{P}\mathbf{B}_t\mathbf{z}. \quad (26)$$

For any $\mathbf{e} \neq 0$, $\boldsymbol{\delta} \neq 0$, derivative (26) will be negative definite if adaptive algorithms are chosen in the form:

– parametric setting:

$$\dot{\boldsymbol{\delta}} = -\mathbf{B}^T\mathbf{P}\mathbf{e}\mathbf{v}^T\boldsymbol{\Gamma}; \ \ \boldsymbol{\Gamma} = diag\{\gamma_1,...\gamma_{n+m}\}; \ \ \gamma_i > 0; \quad (27)$$

– signal setting:

$$\mathbf{z}(t) = -h\,\mathrm{sgn}\left(\mathbf{B}^T\mathbf{P}\mathbf{e}\right); \ h > 0; \ \left[\mathrm{sgn}\left(\mathbf{B}^T\mathbf{P}\mathbf{e}\right)\right]_i = \mathrm{sgn}\left(\mathbf{B}^T\mathbf{P}\mathbf{e}\right)_i. \quad (28)$$

Due to restrictions on the rate of change of parameters and the limitedness of the extended state vector **v**(*t*), the vector function **φ** is also limited with some estimate $\sup_t \|\varphi\| = M_\varphi$. The condition of negative definiteness of the function **V**(**e**, **δ**) with respect to the error **e**(*t*) is satisfied [37] if the value of *h* is chosen as

$$h \geq M_\varphi \left\|\mathbf{B}_t^+\right\|; \quad (29)$$

than guaranteed $\lim_{t\to\infty} e(t) = 0$ and $\lim_{t\to\infty} \dot{\delta}(t) = 0$.

Fig. 5 shows the modified structure of the control system.

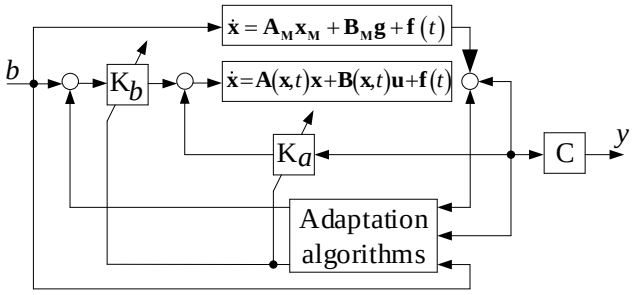


Figure 5: Modified structural diagram of the system with parametric and signal settings

The simplicity and speed of this adaptive control method is balanced by a serious drawback. High-frequency fluctuations caused by the sliding mode are unacceptable when applying the signal adaptation method to helicopters TE control. The method for eliminating oscillations of the signal branch is considered in detail in [43]. In [44], among other things, a comparison of sigmoidal (*sigma*) and relay (sign) functions is given. Replacing the relay function (*sign*) with a smooth function with saturation will solve the problem of high-frequency oscillations in the system. The sigma function is non-linear, smooth, has no singular points, and its non-linearity ensures the quality of signal estimation.

For this purpose, changes were made to equation (28):

$$\mathbf{z}(t) = -h\,\mathrm{sigma}\left(\mathbf{B}^T\mathbf{P}\mathbf{e}\right); \ h > 0; \quad (30)$$

where $\mathrm{sigma}(x) = \frac{1}{1+x^{-\frac{x}{k}}} - 0.5 = 0.5\cdot\frac{1-x^{-\frac{x}{k}}}{1+x^{-\frac{x}{k}}}$ – sigmoidal function, the coefficient *k* = const > 0 determines the slope of the tangent to the sigmoidal function at zero, otherwise, at $k \to +0$ tends to the sign function $\lim_{k\to+0} \mathrm{sigma}(x) = \mathrm{sgn}(x)$. The points $x \approx \pm 1,3k$ serve as natural boundaries that separate the gaps in the domain of definition, where the sigma function is close to either a constant or a linear function [43]. The derivative of the sigma function can be represented as:

$$\text{sigma}'\left(\frac{x}{k}\right)=\frac{1-\text{sigma}^2\left(\frac{x}{k}\right)}{2k}. \tag{31}$$

The sigma function allows you to establish a correspondence between the sets of error values of the parameters and the values of the signal branch. This makes it possible to switch to conventional control, retaining all the advantages of the sliding mode [43, 45].

Thus, the adaptive control algorithm with a reference model and a signal setting has been modified, adapted to helicopters TE control automate in real time.

3.5. Application of Hybrid neuro-PID control with a reference model for the implementation of the developed method for controlling of helicopters turboshaft engines in flight mode

In order to solve the problem, multilayer feed-forward networks are subjected to weight adjustments. This adjustment is carried out on the basis of the developed training algorithms for ANN, which are of three types [46]: training with a teacher, training with assessment, training without a teacher. According to [46], a modified neural control circuit with an emulator and a controller is shown in fig. 6, where y – emulator output, $\hat{e}$ – emulator error. In this case, the neurocontroller is trained on the inverse model of the control object, and the neuroemulator is trained on the real model of the control object (helicopters TE).

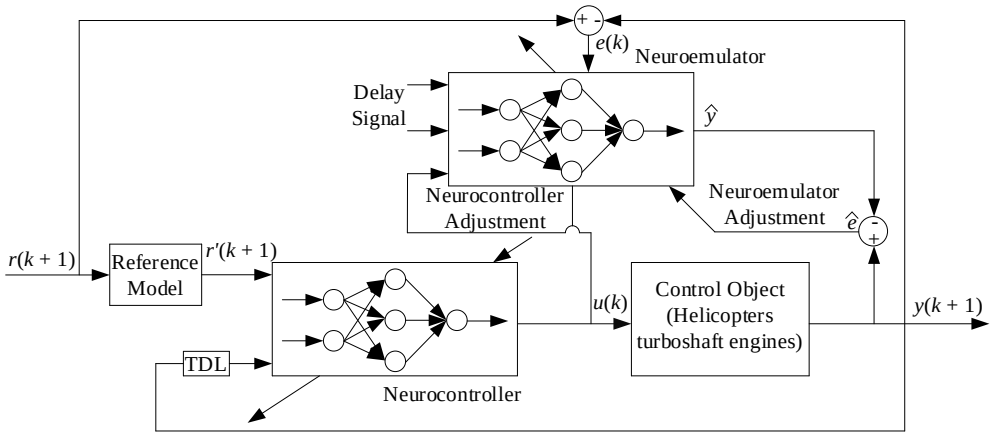


Figure 6: Hybrid neuro-PID control with a reference model with emulator and controller diagram

The neurocontroller is trained on the basis of a neuroemulator, which is trained using the backpropagation method. To train the neuroemulator, we define a multilayer feed-forward network with randomly selected weights and a training set consisting of pairs of network input – desired output {*X*, *D*}, as well as network output value *Y*. The task of neuroregulator training is to select weight coefficients to minimize some objective function – the sum of the squared errors of the network using examples from the training set, that is

$$E(w)=\sum_{j,p}\left(y_{j,p}^{(N)}-d_{j,p}\right)^2; \tag{32}$$

where $y_{j,p}^{(N)}$ – real output of the *N*-th output layer of the network for the *p*-th neuron on the *j*-th training example; $d_{j,p}$ – desired output.

To find the minimum and determine the weight coefficients included in the function $y_{j,p}^{(N)}(x)$, the steepest descent method [47] is used, in which at each training step we will change the weight coefficients according to the expression:

$$\Delta w_{ij}^{(n)}=-\eta\frac{\partial E}{\partial w_{ij}^{(n)}}; \tag{33}$$

where $w_{ij}^{(n)}$ – weight coefficient that convenes the *j*-th neuron of the *n*-th layer and the *i*-th neuron (n – 1)-th layer, η – training rate parameter.

To do this, it is necessary to determine the partial derivatives of the objective function *E* by the obtained weight coefficients of the network:

$$\frac{\partial E}{\partial w_{ij}^{(n)}} = \frac{\partial E}{\partial y_j^{(n)}} \frac{\partial y_j^{(n)}}{\partial s_j^{(n)}} \frac{\partial s_j^{(n)}}{\partial w_{ij}^{(n)}}; \tag{34}$$

where $y_j^{(n)}$ – output, $s_j^{(n)}$ – sum of the inputs of the *j*-th neuron of the *n*-th layer. Knowing the activation function, we can calculate $\frac{\partial y_j^{(n)}}{\partial s_j^{(n)}}$. For a sigmoid function $\frac{\partial y_j^{(n)}}{\partial s_j^{(n)}}$ will be equal to:

$$\frac{\partial y_j^{(n)}}{\partial s_j^{(n)}} = \alpha y_j^{(n)} \left(1 - y_j^{(n)}\right). \tag{35}$$

The output of the *i*-th neuron (n – 1)-th layer can be represented as:

$$\frac{\partial s_j^{(n)}}{\partial w_{ij}^{(n)}} = y_j^{(n-1)}. \tag{36}$$

Thus, having differentiated (34) $y_j^{(N)}$ with respect to (35) and the Kolmogorov theorem with respect to the weights of neurons in the output layer, we calculate the partial derivatives of the objective function:

$$\frac{\partial E}{\partial w_{ij}^{(n)}} = \left(y_j^{(N)} - d_j\right) \frac{\partial y_j^{(N)}}{\partial s_j^{(N)}} y_j^{(N-1)}. \tag{37}$$

Introducing the substitution $\delta_j^{(n)} = \frac{\partial E}{\partial y_j^{(n)}} \frac{\partial y_j^{(n)}}{\partial s_j^{(n)}}$ into (38), we obtain the values of neurons in the output layer:

$$\delta_j^{(N)} = \left(y_j^{(N)} - d_j\right) \frac{\partial y_j^{(n)}}{\partial s_j^{(n)}}. \tag{38}$$

To determine $\alpha y_j^{(n)}$ the weight coefficients of neurons of the inner layers, we write (34) in the following form:

$$\frac{\partial E}{\partial y_j^{(n)}} = \sum_k \frac{\partial E}{\partial y_k^{(n+1)}} \frac{\partial y_k^{(n+1)}}{\partial s_k^{(n+1)}} \frac{\partial s_k^{(n+1)}}{\partial y_j^{(n)}} = \sum_k \frac{\partial E}{\partial y_k^{(n+1)}} \frac{\partial y_k^{(n+1)}}{\partial s_k^{(n+1)}} w_{jk}^{(n+1)}. \tag{39}$$

Note that $\delta_k^{(n+1)} = \sum_k \frac{\partial E}{\partial y_k^{(n+1)}} \frac{\partial y_k^{(n+1)}}{\partial s_k^{(n+1)}}$, which allows using (32) to express the values $\delta_j^{(n)}$ of *n*-th layer neurons by means of (n + 1)-th layer neurons $\delta_k^{(n+1)}$. You can get values $\delta_j^{(n)}$ for milestone neurons of all layers through the recursive formula for the last layer $\delta_j^{(N)}$:

$$\delta_j^{(n)} = \left(\sum_k \delta_k^{(n+1)} w_{jk}^{(n+1)}\right) \frac{dy_j}{ds_j}. \tag{40}$$

Thus, (33) for the correction of weight coefficients takes the form:

$$\Delta w_{ij}^{(n)} = -\eta \delta_j^{(n)} y_j^{(n-1)}. \tag{41}$$

The neuroregulator is trained using the backpropagation algorithm in several stages:

1. Assigning arbitrary initial values to the weight coefficients of the neural network and obtaining the values of the objective function for these values.

2. The vector of the training set is fed to the input of the neural network, and then the output values of the neural network are calculated, which form the memory vector from the values of each neuron.

3. The value $\delta_j^{(N)}$ of the neurons in the output layer is calculated according to (38), and $\delta_j^{(n)}$ according to the recursive formula (40), the values are calculated using the neurons $\delta_k^{(n+1)}$ of the (n + 1)-th layer, and then the weights of the neural network are changed according to (42).

4. Correction of network weights: $w_{ij}^{(n)} = -w_{ij}^{(n)} + \Delta w_{ij}^{(n)}$.

5. The objective function is calculated according to (32) and, if it is relatively small, we can assume that the neural network has successfully passed the training procedure. Otherwise, go to step 2.

Consider as a control object TV3-117 TE, which is part of the power plant of the Mi-8MTV helicopter. The simplified model of TV3-117 TE is described by the following equations:

– gas dispenser angle equation:

$$\dot{A}_{DI} = a_{11} \cdot A_{DI} + a_{12} \cdot G_T + a_{13} \cdot n_{TC}; \quad (44)$$

– fuel consumption equation:

$$\dot{G}_T = a_{21} \cdot A_{DI} + a_{22} \cdot G_T + a_{23} \cdot n_{TK}; \quad (45)$$

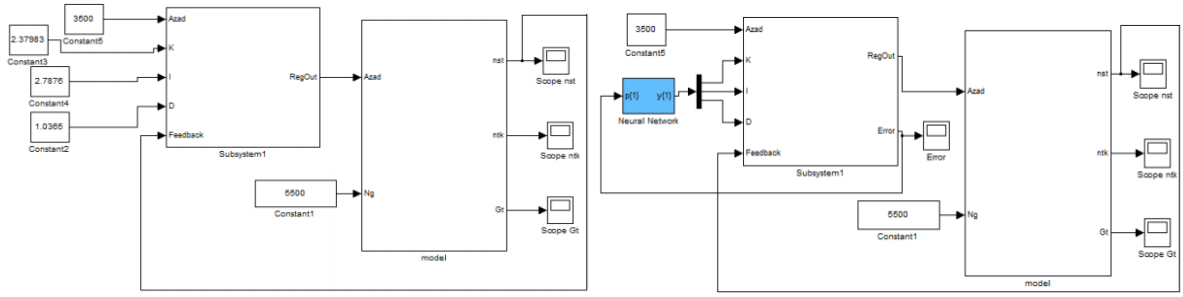
– rotor r.p.m. equation:

$$\dot{n}_{TC} = a_{31} \cdot A_{DI} + a_{32} \cdot G_T + a_{33} \cdot n_{TC}; \quad (46)$$

– free turbine rotor speed equation:

$$\dot{n}_{FT} = a_{41} \cdot G_T + a_{42} \cdot n_{TC} + a_{43} \cdot n_{FT} + a_{44} \cdot A_{DI} + a_{45} \cdot M_{KR}. \quad (47)$$

The gas metering angle A_{DI} regulates the amount of incoming G_T fuel as a result of rotor r.p.m. n_{TC}, the rotation is transferred to the free turbine n_{FT}, which is loaded with M_{KR}. Thus, the general structure of the control model is shown in fig. 7.



a b

Figure 7: General model structure: a – without neural network; b – using a neural network [48]

As input data, the thermogas-dynamic parameters of an aircraft engine n_{TC} – rotor r.p.m.; n_{FT} – free turbine rotor rotational speed; T_G – gas temperature in front of the compressor turbine, recorded on board of the helicopter, reduced to absolute parameters, according to the theory of gas-dynamic similarity, developed by Professor Valery Avgustinovich (table 1).

Table 1

Fragment of the training sample during the operation of helicopters TE (on the example of TV3-117 TE)

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
…	…	…	…
156	0.953	0.973	0.981

4. Helicopters turboshaft engines automatic control system modification

Helicopters TE ACS was developed in [49] (fig. 8), where TE – helicopter TE, TE Model – model of helicopter TE, LB – logical block, FMU – fuel metering unit, FMU model – model of fuel metering unit.

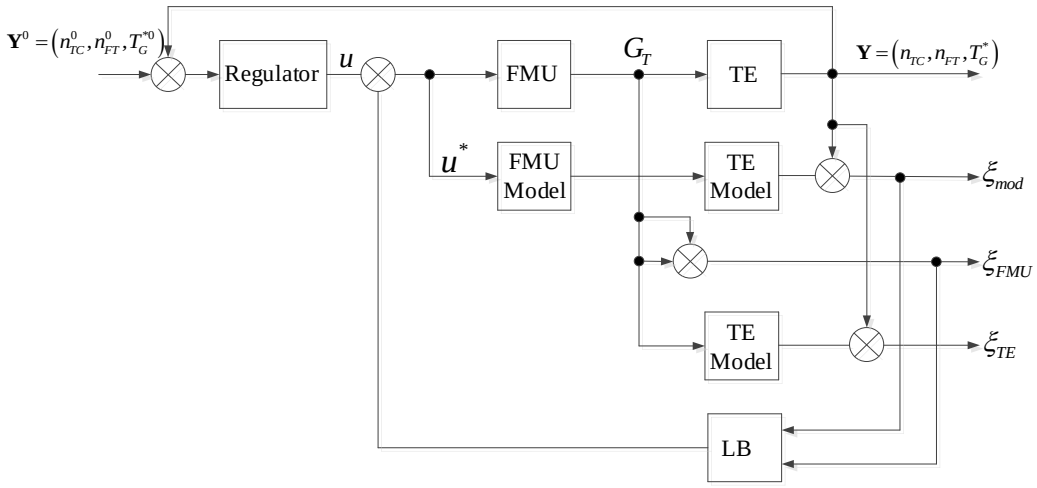


Figure 8: Helicopters TE automatic control system [49]

Modification of developed helicopters TE ACS (fig. 8) consists in adding software modules modified compared to [49] that implement adaptive control methods:

– signal adaptation module with submodules of reference and adjustable models;

– parametric adaptation module.

In this paper, we consider the addition of developed helicopters TE ACS with a reference model module (fig. 9, a), a signal adaptation module (fig. 9, b). The vector **x** is presented in the following form: $x_1 = n_{FT}$ – free turbine rotor rotational speed, $x_2 = n_{TC}$ – rotor r.p.m., x_3 – gas metering regulator integrator, $x_4 – n_{FT}$ regulator integrator, that is, the input data vector $\mathbf{Y}^0$ is supplemented with the free speed parameter. turbines n_{FT} and, accordingly, is converted to the form $\mathbf{Y}^0 = (n_{FT}^0, n_{TC}^0, T_G^{*0})$.

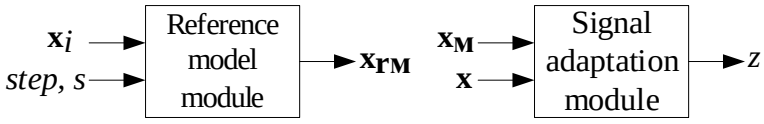


Figure 9: Additional modules structural diagram: a – signal adaptation module; b – reference model module

Reference model module operation principle. The input of the module receives engine's thermogas-dynamic parameters values and the step of solving differential equations. Based on the obtained data, the model state variables are sequentially calculated. The first order method is used to solve differential equations. The reference model state vector $\mathbf{x_{RM}}$ is the output variable of the module.

Signal adaptation module operation principle. The input of the module receives: $\mathbf{x_M}$ – state vector of the custom model and **x** – reduced state vector of helicopters TE. Based on the obtained data, the mismatch vector is calculated. After that, the weighted sum of the mismatch vector is calculated. Then the signal action *z* is calculated. To create a signal adaptive helicopters TE ACS, reference model and signal adaptation modules are additionally included in the standard regulator. The adaptation subsystem will work in accordance with the algorithm shown in fig. 10.

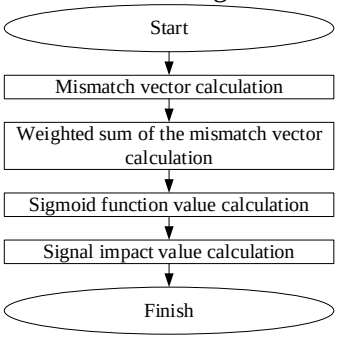


Figure 10: Block diagram of the algorithm of the adaptation module with signal adaptation

5. Results and discussion

5.1. Neural network training results

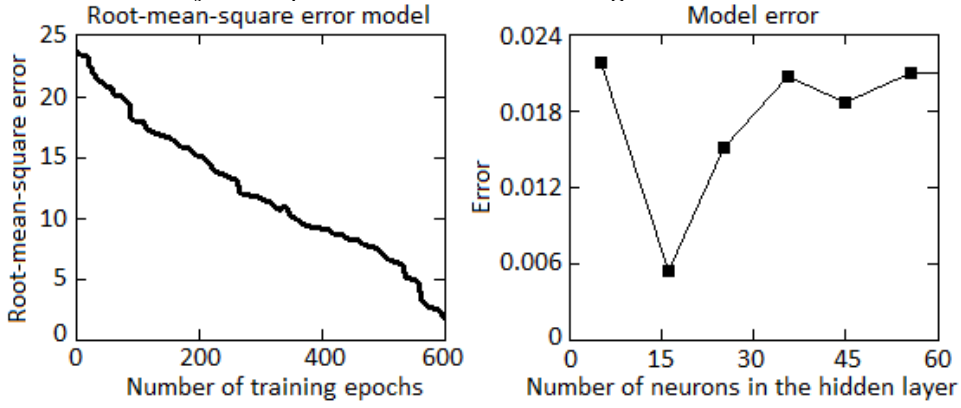
In the course of the experiments, the following parameters of ANN were chosen: 1) type – NEWFF multilayer ANN; 2) number of hidden layers – two; 3) the number neurons in the first layer – 16; 4) number of neurons in the second layer – 20; 5) number of neurons in the output layer – 3; ANN training method – training with a teacher using the error backpropagation algorithm (training a neurocontroller using a neuroemulator) [46]. Table 2 presents a comparative analysis of NEWFF multilayer ANN training results, the results of which gave grounds for choosing an error backpropagation algorithm.

Table 2

Results of neural network training by various algorithms

Training Algorithm	Root-mean-square error	Number of training epochs	Number of neurons in the hidden layer
Proposed algorithm	1.99794	600	16
Back propagation	2.38061	650	18
Conjugate gradient	4.35773	830	36
Quick propagation	4.14182	790	32
Quasi-Newton	3.14325	750	20
Lewenberg-Marquardt	3.07164	720	20
Delta bar delta	3.23218	770	26
Resilient propagation	3.43016	850	24
Genetic Algorithm	2.19735	630	18

The ANN was trained for 600 epochs, the training accuracy characteristic is shown in fig. 11, a, while the steady-state root-mean-square error (RMS) is ~1.99794. According to fig. 11, b, the number of neurons in the hidden layer that provide the smallest training error is 16 neurons.



a b

Figure 11: Neural network training results: a – characteristic of the accuracy of neural network training; b – dependence of training error on the complexity of the neural network

An important issue is the assessment of the homogeneity of the training and test samples. To do this, we use the Fisher-Pearson criterion $\chi 2$ [50] with $r - k - 1$ degrees of freedom:

$$\chi^2 = \min_{\theta} \sum_{i=1}^{r} \left(\frac{m_i - np_i(\theta)}{np_i(\theta)} \right); \quad (48)$$

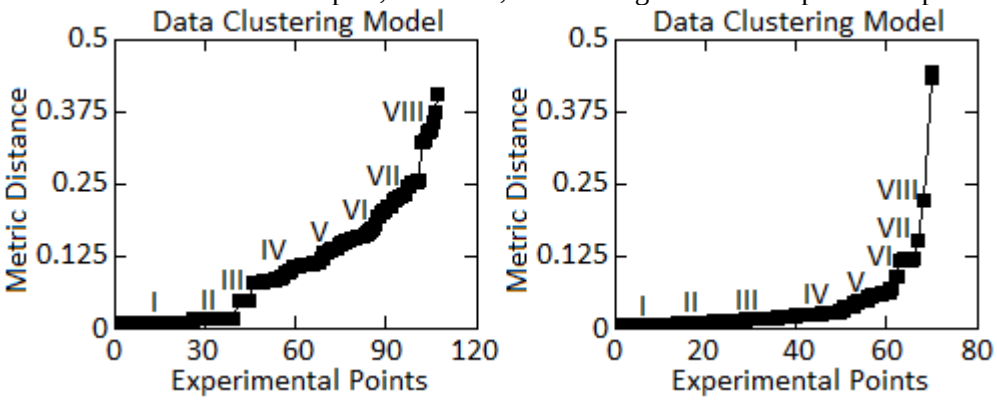
where θ – maximum likelihood estimate found from the frequencies $m_1, \ldots, m_r$; n – number of elements in the sample; $p_i(\theta)$ – probabilities of elementary outcomes up to some indeterminate k-dimensional parameter θ.

The final stage of statistical data processing is their normalization, which can be performed according to the expression:

$$y_i = \frac{y_i - y_{i\min}}{y_{i\max} - y_{i\min}}; \quad (49)$$

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.

In order to establish the representativeness of the training and test samples, a cluster analysis of the initial data was carried out (table 1), during which eight classes were identified (fig. 12, a). After the randomization procedure, the actual training (control) and test samples were selected (in a ratio of 2:1, that is, 67% and 33%). The process of clustering the training (fig. 12, b) and test samples shows that they, like the original sample, contain eight classes each. The distances between the clusters practically coincide in each of the considered samples, therefore, the training and test samples are representative.



a b

Figure 12: Clustering results: a – initial experimental sample (I…VIII – classes); b – training sample

The specified statistics $\chi 2$ allows, under the above assumptions, to test the hypothesis about the representability of sample variances and covariances of factors contained in the statistical model. The area of acceptance of the hypothesis is $\chi^2 \le \chi_{n-m,\alpha}$, where α – significance level of the criterion. The results of calculations according to (48) are given in table 3.

Table 3

Fragment of the training sample during the operation of helicopters TE (on the example of TV3-117 TE)

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
…	…	…	…
156	0.537	0.098	0.651

Calculating the value of $\chi 2$ from the observed frequencies m_1, …, m_r (summing line by line the probabilities of the outcomes of each measured value) and comparing it with the critical values of the distribution $\chi 2$ with the number of degrees of freedom $r - k - 1$. In this work, with the number of degrees of freedom $r - k - 1 = 13$ and $\alpha = 0.05$, the random variable $\chi 2 = 3.588$ did not exceed the critical value from table 3 is 22.362, which means that the hypothesis of the normal distribution law can be accepted and the samples are homogeneous.

5.2. Initial verification of the signal setting with the reference model

In the case of using the signal branch with the sign function sign (29), high-frequency oscillations occur due to the sliding mode. In view of the design of the gas dispenser, these fluctuations are unacceptable. To eliminate this lack of signal adaptation, the signal branch equation was transformed into

(31). Here the sign function sign has been replaced by a smooth sigma function. As a result, high-frequency oscillations were eliminated. Thus, a distinctive feature of the obtained adaptive control method is an additional signal action, which at each moment of time corresponds to a weighted sum of mismatch signals. High-frequency oscillations that occur when using the signal branch with the sign function are shown in fig. 13, a, and an example of the absence of high-frequency oscillations is shown in fig. 13, b.

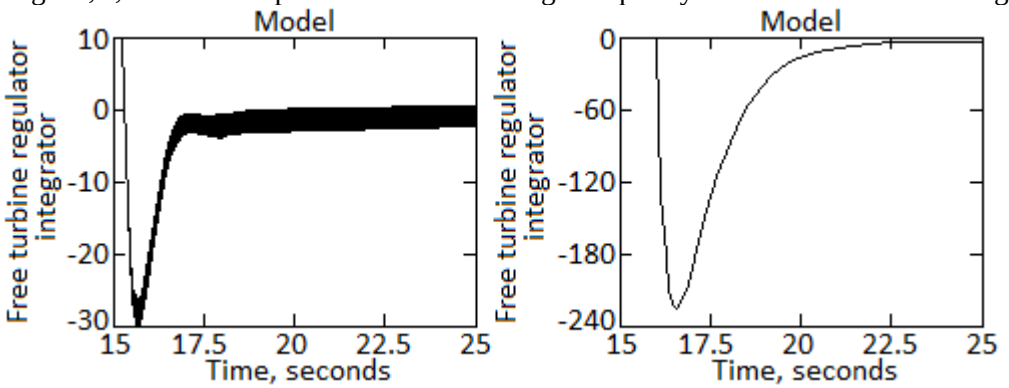


Figure 13: Diagrams of the study of high-frequency oscillations

In the case of using a linear reference model with signal setting (31), a static error occurs. This is due to the fact that fuel consumption and rotor r.p.m. non-linearly, which leads to a non-zero value of the signal branch in static mode. Thus, the reference model must be supplemented with static characteristics.

The results of modeling a neural network system with a signal setting and a reference model are shown in fig. 14, where 1 – reference model, 2 – system with the signal setting, 3 – system with the factory regulator.

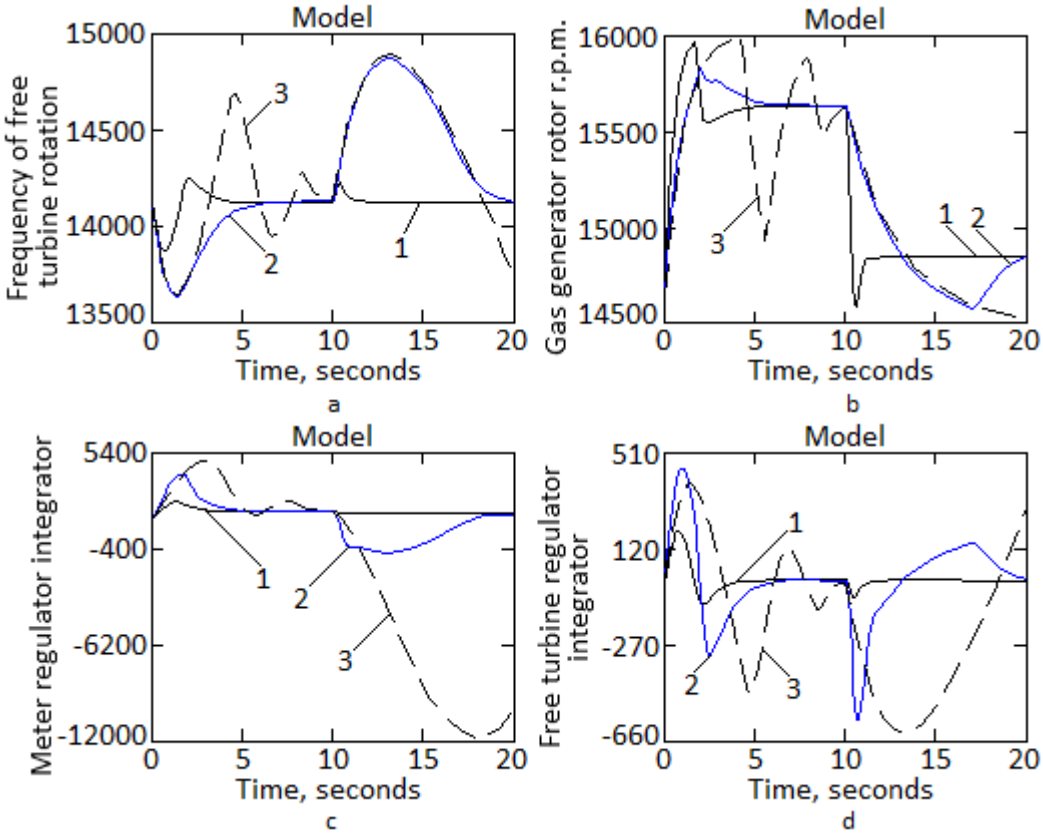


Figure 14: Diagrams of change: a – free turbine speed; b – rotor r.p.m.; c – dispenser controller integrator; d – free turbine regulator integrator

Thanks to the use of signal setting, the transient time has been significantly reduced and the stability of the system has increased. Improvement in quality indicators during transients are shown in tables 4 and 5.

Table 4

Quality indicators for n_{FT} of the reference model with a signal regulator

Regulator type	Maximum deviation, rpm	Transient process time, s	Number of vibrations
Regular	450	8.5	2
Adaptive	320	3.7	0

Table 5

Improvement of quality indicators for n_{FT} of the reference model with a signal regulator

Improvement, %	30.18	58.37	100
Section of the transition process, s	0…10	0…10	0…10

5.3. Secondary verification of the signal setting with the reference model

The next step is the study of adaptive controllers on complex element-by-element models of aviation gas turbine engines of helicopters. Let us consider the operation of an adaptive controller with a signal branch (30) and a reference model [43]. The check is carried out in a full-time engine regulator, which includes various limitation and control circuits. The free turbine speed stabilization circuit is adaptive. At the first stage, the load was set by an instantaneous change in the active power, further checks were made for a variety of operational situations of the engine. Comparison of a standard regulator and a standard regulator with a signal setting is shown in fig. 15, where 1 – standard regulator with a signal setting, 2 – standard regulator.

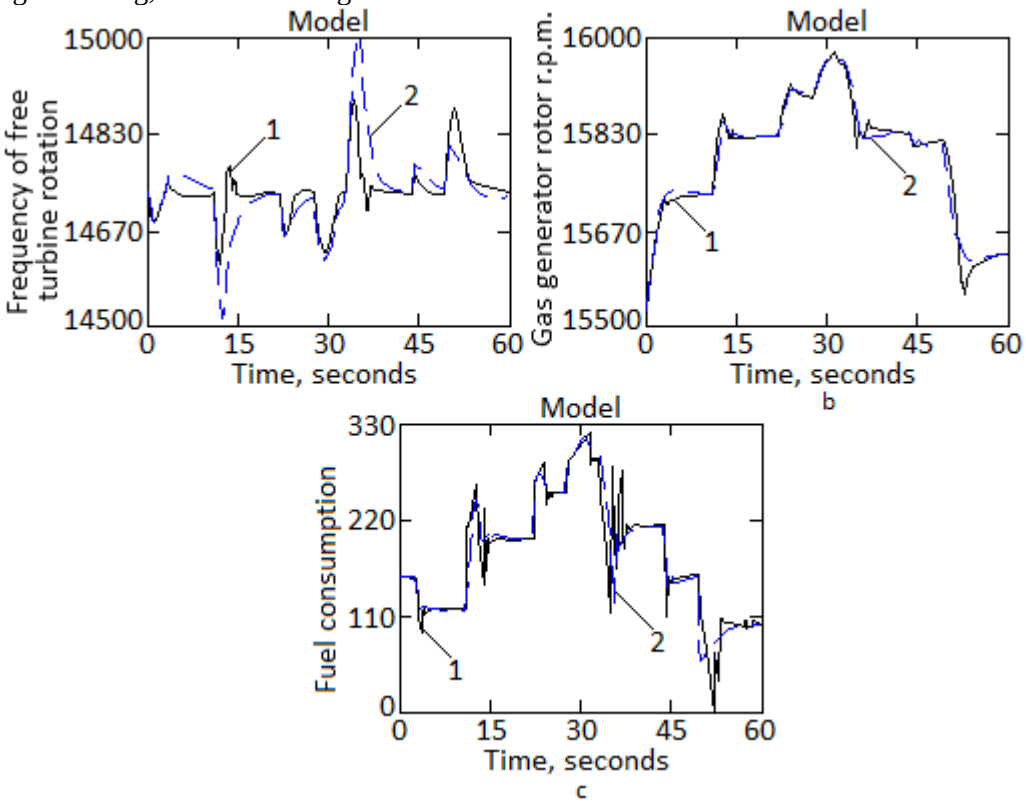


Figure 15: Diagrams of change: a – free turbine speed; b – rotor r.p.m.; c – fuel consumption

From fig. 15 it follows that the engine operation mode with a load change affects the improvement of the quality of free turbine speed transient processes, which is associated with the setting of the reference model, while from the 45th second the reaction of the reference model is deliberately idealized, as a result of which the mismatch of the parameters with the element-by-element model engine gets too big. As a result, the fuel consumption generated by the adaptive loop becomes unacceptable, therefore, the control priority is transferred to another control loop. From this it follows that the adaptive ACS does not pose a danger, thanks to the selection scheme. In table. 6 and 7 show the improvement in quality indicators.

Table 6

Quality indicators for n_{FT} of the reference model with a signal regulator

Regulator type	Maximum deviation, rpm	Transient process time, s	Number of vibrations
Regular	1080	4.8	0
Adaptive	580	3.2	1

Table 7

Improvement of quality indicators for n_{FT} of the reference model with a signal regulator

Improvement, %	43.24	60.96	–
Section of the transition process, s	30…40	30…40	10…20

5.4. Transient process section simulation

In the process of modeling the transient process in the time interval from 0 to 10 s, we consider that the control variable is the angle of the gas dispenser, and the control variable is the free turbine rotor speed. The paper compares the output graph of the transient process using PID control with the graph of the transient process using hybrid neuro-PID control, the resulting diagram is shown in fig. 16.

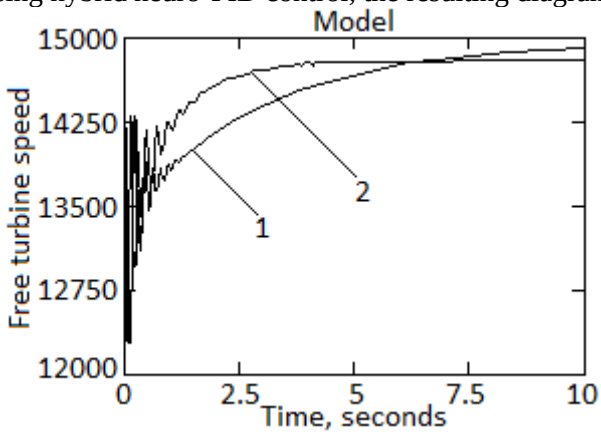


Figure 16: Transient process diagram: 1 – initial diagram using PID control (fig. 7, a); 2 – original diagram using hybrid neuro-PID control (fig. 7, b)

From fig. 16 shows that when using hybrid neuro-PID control, the following indicators of control quality are improved: error in the steady state, overshoot and transient time, the number of oscillations during the transient process. The system comparison is justified because the PID controller tuning in the first experiment is the starting point for neural network tuning.

5.5. Performance evaluation

A comparative analysis of the accuracy of the classical and neural network methods for controlling helicopters TE (on the example of the TV3-117 engine) is given in table 8, which shows the probabilities of errors of the 1st and 2nd kind when determining the optimal parameters n_{TC} and n_{FT}.

Table 8

Comparative characteristics of methods

Method of determination	Probability of error in determining the optimal parameters n_{TC} and n_{FT}, %			
	Determination of the optimal parameter n_{TC}		Determination of the optimal parameter n_{FT}	
	Type 1st error	Type 2nd error	Type 1st error	Type 2nd error
Classic (method of tolerance control)	1.85	1.12	2.38	1.76
Neural Network	0.63	0.24	0.74	0.24

The result obtained confirms the possibility of using hybrid neuro-PID control in the framework of the problem under consideration.

6. Conclusions

1. The method of adaptive control with a reference model and signal setting has been further developed, which makes it possible to helicopters turboshaft engines controlling process automate at flight modes.

2. The neural network method for monitoring the operational status of helicopters turboshaft engines at flight modes was further developed, which, through the use of a hybrid neuro-PID control, including neural control with an emulator and controller, made it possible to reduce errors of the first and second kind in determining the optimal engine parameters.

3. It has been proven that the use of signal regulator units with a reference model in helicopter aircraft engines automatic control system improves the quality of transient recognition by an average of 60 % compared to the use of standard controllers.

7. References

[1] W. A. Khan, S.-H. Chung, H.-L. Ma, S. Q. Liu, C. Y. Chan, A novel self-organizing constructive neural network for estimating aircraft trip fuel consumption, Transportation Research Part E: Logistics and Transportation Review, vol. 132 (2019) 72–96. doi: 10.1016/j.tre.2019.10.005

[2] S. Zhou, J. Wang, B. Xu, Innovative coupling and coordination: Automobile and digital industries, Technological Forecasting and Social Change, vol. 176 (2022) 121497. doi: 10.1016/j.techfore.2022.121497

[3] Y. Ahn, Y. Kim, Data mining in sloshing experiment database and application of neural network for extreme load prediction, Marine Structures, vol. 80 (2021) 103074. doi: 10.1016/j.marstruc.2021.103074

[4] Y. Xu, X. Yan, B. Sun, Z. Liu, Global contextual residual convolutional neural networks for motor fault diagnosis under variable-speed conditions, Reliability Engineering & System Safety, vol. 225 (2022) 108618. doi: 10.1016/j.ress.2022.108618

[5] D. Gajic, I. Savic-Gajic, I. Savic, O. Georgieva, S. Di Gennaro, Modelling of electrical energy consumption in an electric arc furnace using artificial neural networks, Energy, vol. 108 (2016) 132–139. doi: 10.1016/j.energy.2015.07.068

[6] H. M. El Zoghby, H. S. Ramadan, Enhanced dynamic performance of steam turbine driving synchronous generator emulator via adaptive fuzzy control, Computers & Electrical Engineering, vol. 97 (2022) 107666. doi: 10.1016/j.compeleceng.2021.107666

[7] Y. Chen, B. Chen, Y. Yao, C. Tan, J. Feng, A spectroscopic method based on support vector machine and artificial neural network for fiber laser welding defects detection and classification, NDT & E International, vol. 108 (2019) 102176. doi: 10.1016/j.ndteint.2019.102176

[8] H.-P. Ren, S.-S. Jiao, X. Wang, J. Li, Adaptive RBF Neural Network Control Method for Pneumatic Position Servo System, IFAC-PapersOnLine, vol. 53, issue 2 (2020) 8826–8831. doi: 10.1016/j.ifacol.2020.12.1394

[9] O. Rudenko, O. Bezsonov, Robust training of ADALINA based on the criterion of the maximum correntropy in the presence of outliers and correlated noise, 5th International Conference on Computational Linguistics and Intelligent Systems (COLINS 2021). Volume I: Main Conference Lviv, Ukraine, April 22–23, 2021, CEUR Workshop Proceedings, vol. 2870 (2021) 1694–1705.

[10] R. Vang-Mata, Multilayer Perceptrons: Theory and Applications, New York, Nova Science Publishers, 2020, 143 p.

[11] F. M. Salem, Recurrent Neural Networks: From Simple to Gated Architectures, Switzerland, Springer Nature Switzerland AG, 2022, 200 p.

[12] H.-G. Han, M.-L. Ma, H.-Y. Yang, J.-F. Qiao, Self-organizing radial basis function neural network using accelerated second-order learning algorithm, Neurocomputing, vol. 469 (2022) 1–12. doi: 10.1016/j.neucom.2021.10.065

[13] D. Plonis, A. Katkevicius, V. Urbanavicius, D. Miniotas, A. Serackis, A. Gurska, Delay systems synthesis using multi-layer perceptron network, Acta Physica Polonica A, vol. 133, no 5 (2018) 1281–1286. doi: 10.12693/APhysPolA.133.1281
[14] X. Zhang, R. Jing, Z. Li, Z. Li, X. Chen, C.-Y. Su, Adaptive pseudo inverse control for a class of nonlinear asymmetric and saturated nonlinear hysteretic systems, IEEE/CAA J. Autom. Sinica, vol. 8, no. 4 (2021) 916–928. doi: 10.1109/JAS.2020.1003435
[15] B. Perez-Sanchez, O. Fontenla-Romero, B. Guijarro-Berdinas, A review of adaptive online learning for artificial neural networks, Artificial Intelligence Review, vol. 49 (2018) 281–299. doi: 10.1007/s10462-016-9526-2
[16] R. T. Thibault, A. MacPherson, M. Lifshitz, R. R. Roth, A. Raz, Neurofeedback with fMRI: A critical systematic review, NeuroImage, vol. 172 (2018) 786–807. doi: 10.1016/j.neuroimage.2017.12.071
[17] C. J. Vega, L. Djilali, E. N. Sanchez, Secondary control of microgrids via neural inverse optimal distributed cooperative control, IFAC-PapersOnLine, vol. 53, issue 2 (2020) 7891–7896. doi: 10.1016/j.ifacol.2020.12.1973
[18] F. Errica, D. Bacciu, A. Micheli, Graph Mixture Density Networks, Proceedings of the 38th International Conference on Machine Learning, PMLR 139, 2021. URL: http://proceedings.mlr.press/v139/errica21a/errica21a.pdf doi: 10.48550/arXiv.2012.03085
[19] F. Bonassi, M. Farina, R. Scattolini, Stability of discrete-time feed-forward neural networks in NARX configuration, IFAC-PapersOnLine, vol. 54, issue 7 (2021) 547–552. doi: 10.1016/j.ifacol.2021.08.417
[20] A. Chernodub, A. Dzuba, Review of neurocontrol methods, Programming problems, no. 2 (2017) 79–94.
[21] F. Baghbani, M.-R. Akbarzadeh-T, M.-B. Naghibi Sistani, Stable robust adaptive radial basis emotional neurocontrol for a class of uncertain nonlinear systems, Neurocomputing, vol. 309 (2018) 11–26. doi: 10.1016/j.neucom.2018.03.051
[22] G. Hernandez-Mejia, A. Y. Alanis, E. A. Hernandez-Vargas, Neural inverse optimal control for discrete-time impulsive systems, Neurocomputing, vol. 314 (2018) 101–108. doi: 10.1016/j.neucom.2018.06.034
[23] A. Mesbah, J. A. Paulson, R. D. Braatz, An internal model control design method for failure-tolerant control with multiple objectives, Computers & Chemical Engineering, vol. 140 (2020) 106955. doi: 10.1016/j.compchemeng.2020.106955
[24] Yong Li, J. Han, Y. Cao, Yunxuan Li, J. Xiong, D. Sidorov, D. Panasetsky, A modular multilevel converter type solid state transformer with internal model control method, International Journal of Electrical Power & Energy Systems, vol. 85 (2017) 153–163. doi: 10.1016/j.ijepes.2016.09.001
[25] S. H. Son, J. W. Kim, T. H. Oh, D. H. Jeong, J. M. Lee, Learning of model-plant mismatch map via neural network modeling and its application to offset-free model predictive control, Journal of Process Control, vol. 115 (2022) 112–122. doi: 10.1016/j.jprocont.2022.04.014
[26] M. Oliver, Practical guide to the simplex method of linear programming, 2020 URL: http://math.jacobs-university.de/oliver/teaching/iub/spring2007/cps102/handouts/linear-programming.pdf
[27] H. Zhang, Q. Ni, A new regularized quasi-Newton algorithm for unconstrained optimization, Applied Mathematics and Computation, vol. 259 (2015) 460–469. doi: 10.1016/j.amc.2015.02.032
[28] Q. Deng, B. F. Santos, Lookahead approximate dynamic programming for stochastic aircraft maintenance check scheduling optimization, European Journal of Operational Research, vol. 299, issue 3 (2022) 814–833. doi: 10.1016/j.ejor.2021.09.019
[29] H. Zwart, K. A. Morris, O. V. Iftime, Optimal linear–quadratic control of asymptotically stabilizable systems using approximations, Systems & Control Letters, vol. 146 (2020) 104802. doi: 10.1016/j.sysconle.2020.104802
[30] Y. Chen, R.-H. Li, Q. Dai, Z. Li, S. Qiao, R. Mao, Incremental structural clustering for dynamic networks, Web Information Systems Engineering – WISE 2017, part I (2017) 123–134.
[31] G.-C. Hao, Y.-L. Zhang, S.-X. Wen, X. Du, B. Yang, X.-M. Sun, A softly switching multiple model predictive control for aero-engines, IFAC-PapersOnLine, vol. 54, issue 10 (2021) 477–482. doi: 10.1016/j.ifacol.2021.10.208
[32] E. Uchibe, K. Doya, Forward and inverse reinforcement learning sharing network weights and hyperparameters, Neural Networks, vol. 144 (2021) 138–153. doi: 10.1016/j.neunet.2021.08.017
[33] F. Highland, C. Hart, Unsupervised Learning of Patterns Using Multilayer Reverberating Configurations of Polychronous Wavefront Computation, Procedia Computer Science, vol. 95 (2016) 175–184. doi: doi.org/10.1016/j.procs.2016.09.310

[34] Y.-P. Huang, S.-Y. Wen, W.-C. Xiang, Y.-S. Jin, PID parameters self-tuning based on genetic algorithm and neural network, Artificial Intelligence Science and Technology: Proceedings of the 2016 International Conference (AIST2016), (2017) 14–20. doi: 10.1142/9789813206823_0003
[35] S. Saadatmand, P. Shamsi, M. Ferdowsi, Adaptive critic design-based reinforcement learning approach in controlling virtual inertia-based grid-connected inverters, International Journal of Electrical Power & Energy Systems, vol. 12 (2021) 106657. doi: doi.org/10.1016/j.ijepes.2020.106657
[36] S. B. Joseph, E. G. Dada, A. Abidemi, D. O. Oyewola, B. M. Khammas, Metaheuristic algorithms for PID controller parameters tuning: review, approaches and open problems, Heliyon, vol. 8, issue 5 (2022) e09399. doi: 10.1016/j.heliyon.2022.e09399
[37] L. Morales, J. Aguilar, A. Rosales, D. Chavez, P. Leica, Modeling and control of nonlinear systems using an Adaptive LAMDA approach, Applied Soft Computing, vol. 95 (2020) 106571 doi: 10.1016/j.asoc.2020.106571
[38] 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/kntu2078-4481.2021.4.6
[39] J. Zheng, J. Chang, J. Ma, D. Yu, Performance uncertainty propagation analysis for control-oriented model of a turbine-based combined cycle engine, Acta Astronautica, vol. 153 (2018) 39–49. doi: 10.1016/j.actaastro.2018.10.009
[40] E. Hedrick, K. Hedrick, D. Bhattacharyya, S. E. Zitney, B. Omell, Reinforcement learning for online adaptation of model predictive controllers: Application to a selective catalytic reduction unit, Computers & Chemical Engineering, vol. 160 (2022) 107727 doi: 10.1016/j.compchemeng.2022.107727
[41] V. Veerasamy, N. I. Abdul Wahab, R. Ramachandran, M. Lutfi Othman, H. Hizam, J. S. Kumar, A. X. Raj Irudayaraj, Design of single- and multi-loop self-adaptive PID controller using heuristic based recurrent neural network for ALFC of hybrid power system, Expert Systems with Applications, vol. 192 (2022) 116402 doi: 10.1016/j.eswa.2021.116402
[42] S. M. Hosseinimaab, A. M. Tousi, A new approach to off-design performance analysis of gas turbine engines and its application, Energy Conversion and Management, vol. 243 (2021) 114411. doi: 10.1016/j.enconman.2021.114411
[43] I. Bakhirev, B. Kavalerov, Adaptive control of a gas turbine plant with a reference model and a sigmoid function, Control systems and information technologies, no 3.1 (61) (2015) 118–123.
[44] S. Krasnova, N. Mysik, Cascade synthesis of a state observer with nonlinear corrective actions, Automation and telemechanics, no 2 (2014) 106–128.
[45] I. Bakhirev, B. Kavalerov, On the adaptive control of a gas turbine power plant with a reference model, Automation in the electric power industry and electrical engineering: materials of the I International Scientific and Technical Conference, 2015 31–37.
[46] Y. Shmelov, S. Vladov, Y. Klimova, M. Kirukhina, Expert system for identification of the technical state of the aircraft engine TV3-117 in flight modes, System Analysis & Intelligent Computing: IEEE First International Conference on System Analysis & Intelligent Computing (SAIC), 08–12 October 2018. 77–82. doi: 10.1109/SAIC.2018.8516864
[47] 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
[48] S. Vladov, N. Yankevych, D. Khodin, Application of neural network technologies in the tasks of controlling helicopters aircraft gas turbine engines in flight modes, Management of high-speed moving objects and professional training of operators of complex systems (on the occasion of the 70th anniversary of the academy): materials of the X International Scientific and Practical Conference, 2021 41–43.
[49] S. Vladov, Y. Shmelov, R. Yakovliev, Helicopters Aircraft Engines Self-Organizing Neural Network Automatic Control System. The Fifth International Workshop on Computer Modeling and Intelligent Systems (CMIS-2022), May, 12, 2022, Zaporizhzhia, Ukraine, CEUR Workshop Proceedings, vol. 3137 (2022) 28–47. doi: 10.32782/cmis/3137-3
[50] H.-Y. Kim, Statistical notes for clinical researchers: Chi-squared test and Fisher's exact test, Restor Dent Endod, vol. 42, no. 2 (2017) 152–155. doi: 10.5395/rde.2017.42.2.152