UDC 629.735.083.03.004.58:004.801.3(045)
DOI: 10.15587/1729-4061.2020.218137

DEVELOPMENT OF A METHOD FOR OPTIMIZING THE STRUCTURE OF STATIC NEURAL NETWORKS INTENDED FOR CATEGORIZING TECHNICAL STATE OF GAS-TURBINE ENGINES

O. Yakushenko
PhD, Associate Professor, Senior Researcher*
E-mail: yyysss@i.ua
O. Popov
PhD, Associate Professor
Department of Aircraft Continuing Airworthiness**
E-mail: popche75@gmail.com
A. Mirzoyev
PhD, Senior Researcher
Department of Flying Machines and Aviation Engines
National Academy of Aviation
Bina highway, 25, Baku, Azerbaijan, AZ1045
E-mail: azermirzoyev@gmail.com
O. Chumak
Deputy Director
TOV Aviaremontne Pidpryiemstvo URARP
Polova str., 37, Kyiv, Ukraine, 03056
E-mail: chumak113@ukr.net
V. Okhmakevych
Researcher*
E-mail: vnakuka@ukr.net
*Department of Aviation Engines**
**National Aviation University
Liubomyra Huzara ave., 1, Kyiv, Ukraine, 03058

A process of creating a static neural network intended for diagnosing bypass gas turbine aircraft engines by a method of categorizing the technical state of the engine flow path was considered. Diagnostics depth was "to the structural assembly". A variant of diagnosing single faults of the flow path was considered.

The following tasks were set:

– select the best neuron activation functions in the network layers;

– determine the number of layers;

– determine the optimal number of neurons in layers;

– determine the optimal size of the training set.

The problem was solved taking into account the influence of parameter measurement errors.

The method of structure optimization implies training the network of the selected configuration using a training data set. The training was periodically interrupted to analyze the results of the network operation according to the criterion characterizing the quality of classification of the engine technical state. The assessment was performed with training and control sets. The network that provides the best value of the classification quality parameter assessed by the test set was selected as the final network.

The PS-90A turbojet engine was selected as the object of diagnostics. Diagnostics was carried out on takeoff mode and during the initial climb.

Primary optimization was carried out according to the data with no measurement errors. It was shown that a two-layer network with the use of neurons having a hyperbolic tangent function in both layers is sufficient to solve the problem. The size of the first network layer was finally optimized according to the data containing measurement errors. A two-layer network with eight neurons in the first layer was obtained. The share of erroneous diagnoses measured 14.5 %

Keywords: static neural network, gas turbine engine, activation function, hyperbolic tangent

Received date 22.10.2020
Accepted date 26.11.2020
Published date 18.12.2020

1. Introduction

One of the ways to improve the quality and efficiency of diagnosing gas turbine engines (GTE) implies an automated analysis of functioning parameters implemented in the form of a computer diagnostic system. One of the promising methods for determining the technical state (TS) of an object using elements of artificial intelligence implies diagnosing it with the help of a neural network (NN).

In parametric diagnostics of the engine, parameters of its working process are processed using the selected method. The processing can result in the following:

– determination of integral parameters of the engine TS (flight performance, thrust in a given mode, specific fuel consumption, etc.);

– evaluation of the TS parameters of individual GTE assemblies (shift of performance characteristics and the degree of pressure increase in the compressor, the turbine throughput, etc. relative to the corresponding standard characteristics of the assemblies);

– classification of the TS engine as a whole and its individual assemblies.

The fact that the algorithm of information processing inside the NN is usually a black box even for the method developer is one of the features of parametric diagnostics using NNs. At the same time, the developer collects the data necessary for training the neural network, sets its general structure, trains the network, and monitors the results obtained. As usual, the internal network parameters obtained in the course of training are not analyzed. Thus, when creating such a diagnostic system, the developer should concentrate efforts on the issues related to obtaining data for training the network and its optimization.

The second feature of the neural network is the fact that before the network can start working, it must be trained using previously prepared examples. At the same time, the developer does not interfere in the training process itself but estimate the quality of training based on the results of the operation of the trained NN.

The third feature of the neural network is the flexibility of its structure. At the same time, the developer can easily increase the number of network layers, the number of neurons in hidden layers, and change the activation function (AF) for activating neurons in each layer [1]. The desire to obtain an NN with a minimal diagnostic error can easily lead to the creation of an unjustifiably complex network which is often accompanied by the emergence of the so-called NN overlearning effect [1–5].

Taking into account all the above features when designing a network intended for diagnosing gas turbine engines, it is necessary to determine its following characteristics:

– the nature of the processed information (ordered time series of the measured parameters or sets of parameters measured at independent points in time);

– the result of the NN operation (classification of the object TS or definition of the parameters that characterize its TS);

– the nature of training (training with a “trainer” or without a “trainer” [6, 7]);

– training method;

– the NN type (network with direct signal propagation, recurrent NN, etc.);

– the number of layers in the network;

– the type of AF for activating neurons in each layer of the NN;

– the number of neurons in each layer;

– the number of neuron connections in a layer and between layers.

A more detailed description of the NN structure and operation can be found in [8–10].

The outlined range of issues is wide enough. This article describes the process of creating a static neural network with direct signal propagation. The network under consideration provides a classification of the GTE TS according to the results of processing the parameters measured at separate, unrelated points in time. In this case, pre-classified samples are used (training with a “trainer”). When building a network, it is assumed that all neurons of the input layer are connected to all of its inputs. Similarly, all inputs of the neurons in the remaining layers are connected to the outputs of all neurons in the previous layers. Networks with delay elements, lateral and back coupling as well as adaptive neural networks are not considered there.

The developed approach makes it possible to obtain a neural network of minimum size. At the same time, the resulting network provides the best possible quality of classification of the power unit TS without the occurrence of the effects associated with network overlearning. The use of this approach in the systems for diagnosing aviation equipment will make it possible to more accurately diagnose the GTEs. This will result in the growth of the flight safety level.

2. Literature review and problem statement

A number of promising approaches to solving the problems of diagnosing and predicting the technical state of complex technical systems (GTEs, in particular, aircraft GTEs) are described in [11–19] which will make it possible to conduct more deep studies in the future in order to minimize the likelihood of an erroneous diagnosis.

The studies were carried out in [11] using a genetic algorithm to adapt available engine characteristics to the characteristics of the object considered. As a result of this procedure, new characteristics of compressors were created and a more accurate forecast of their changes was ensured. This adaptive approach provides an alternative method of creating mathematical models for analyzing TS of the GTE flow path. However, the question remains unresolved concerning individual features of characteristics of engine compressors taking into account their degradation during operation.

The results of testing artificial intelligence methods in solving diagnostic problems based on parameters of the GTE working process were presented in [12]. It was shown that the presence of such factors as the number of controlled and recorded engine parameters and the accuracy of their measurement is of great importance. However, the issues related to the choice of a mathematical model of the engine’s working process taking into account its testability have remained unresolved.

The study conducted in [13] has resulted in a review of effective methods for diagnosing gas turbines based on analysis of operational characteristics. It was shown that the use of these methods makes it possible to assess the technical condition of a gas turbine, however, there are still unresolved issues related to their use in diagnosing aviation GTEs. This is determined by the peculiarities of their functioning and operation.

As a result of the studies carried out in [14], special attention was paid to the effectiveness of the application of existing methods for engine diagnosing, for example, genetic algorithms, artificial neural networks, fuzzy sets [15, 16]. It was shown that the considered methods have both advantages and disadvantages. However, the results of these studies make it possible to conclude that it is advisable to use neural network classification when recognizing TSs of an object being diagnosed.

The results of studying the use of NN for diagnosing objects in transient (unsteady) modes of engine operation were presented in [17]. A set of NNs designed to assess independent changes in parameters of the engine operating process caused by malfunctions in one or more structural units of a turbofan engine was presented. The approach includes the networks for classification and approximation. The measured motor parameters were first estimated by a trained network. If a malfunction was diagnosed, then an in-depth diagnosis was carried out using another network which makes it possible to identify both sensor malfunctions and malfunctions in structural assemblies. The disadvantage of this approach consists in that the measurement errors are not taken into account and coefficients of dynamic characteristics of the engine are not considered.

Application of artificial intelligence methods as a part of automated fault detection systems was proposed and the main advantages and disadvantages of the methods described were considered in [18, 19]. However, the possibility of using two or more methods in a complex was not considered. All this makes it possible to note the need for further studies in order to determine rules of making decisions when diagnosing GTEs based on an integrated approach using artificial intelligence methods as a part of automated systems.

In addition, the very process of preparing a neural network is a non-trivial task as well. General issues of increasing speed of preparation, the accuracy of work, and adequacy of the obtained NN were considered in [2, 3, 5, 20].

The overlearning effect is one of the problems with neural networks. It manifests itself strongly when using small-size training samples. The solution to this problem by choosing a good initial approximation of the NN parameters was described in [2].

The appearance of the overtraining effect can be avoided by increasing the network throughput but this requires the application of special methods. Regularization methods provide an easy way to prevent overlearning for large neural networks. The use of an early stop with cutoff is one of the recent recommendations regarding regularization [3, 4].

Large-size multilayer neural networks require significant computing and memory resources at the stage of preparation. To eliminate these limitations, a method was developed in [5] that makes it possible to reduce by an order of magnitude the requirements to the stored data volume and amount of computations performed to prepare the neural network. Moreover, the use of this method does not affect the result itself.

Attention was paid in [20] to determining the NN structure and the process of building a workflow model based on the NN. The model building includes system analysis, data collection and preparation, network architecture development, network training, and validation. However, there are no recommendations on the use of the presented approaches to modeling workflows for aircraft GTE turbines of a multi-shaft and bypass design.

As follows from the analysis of the data presented in the above studies, each of the methods has a series of advantages and disadvantages and does not have universality, but it allows one to evaluate TS of a gas turbine engine at the “serviceable - faulty” level. However, at the present stage of development of systems for ensuring and maintaining flight worthiness of aircraft and their components, this level of diagnosis is not enough, especially at the stage of aircraft operation. In this case, it is necessary to pay attention to the development of methods for assessing TS of the flow path in the GTE with a solution to the problem of diagnostics up to the “structural unit”. In this case, the method of pattern recognition (classification) using the NN for solving problems of diagnosing the aircraft GTEs makes it possible to assess the GTE technical condition to a given depth and is quite promising. However, it requires a more detailed study in order to increase the probability of correct diagnosis. It should also be noted that as a result of analysis of advantages and disadvantages of using artificial intelligence methods in assessing TS of a GTS, it is obvious that the development of methods of an integrated approach (synthesis of methods) in solving the problems of diagnosing the aviation GTEs is topical.

3. The aim and objectives of the study

The study objective is to devise a method for preparing a static NN intended for the classification of aircraft GTE TS according to the parameters measured in operation.

To achieve the objective, the following tasks were set:

– develop a method for determining a combination of neuron activation functions optimal for solving the problem;

– develop a method for assessing the amount of representative training and control sets which sufficiently fully characterize the object workflow with a given TS nomenclature and with an available system for measuring and recording the operation parameters;

– develop a method for optimizing the number of NN layers and the number of neurons in them.

4. The method of optimizing the structure of a neural network designed for diagnosing gas turbine engines and description of the study object

4. 1. The method of optimizing the structure of a neural network designed for diagnosing gas turbine engines

When training neural networks, one of the versions of the method of two data sets [21–23] is used. When using this method, the network is trained using the first (training) set. Training is periodically stopped and the second (control) set is fed to the input of the trained network and then the correctness of its recognition is estimated. The duration of one such iteration can range from one to several thousand training epochs and depends on the training algorithm used. In this case, the maximum number of training epochs E_{max} is set.

In the process of training such a network, two situations may arise:

– a overlearning effect appears in the trained network;

– the network being trained stops training.

If none of these situations occurs, it is necessary to increase the E_{max} value.

When the overlearning effect occurs, the trained NN begins to accurately describe the data used in its training (training set) but poorly describes the data that are not included in this set (control set). This phenomenon occurs if the network is too powerful for the task at hand and the volume of the training set is small, the training set is not a representative sample, the training and test sets are from different populations. The moment of appearance of this effect is the moment when the parameter characterizing the quality of the NN operation and obtained for the training set continues to improve and the parameter obtained for the control set starts to deteriorate steadily. If overlearning takes place, a further increase in the structure of the neural network does not make sense and its optimization ends.

If there is no overlearning effect in the network being trained, then the quality of its work reaches a certain level and then practically does not change. In this case, the number of neurons in the hidden layers increases, or the number of the NN layers increases. The process of changing the NN is completed if the complication of its structure does not lead to an improvement in the quality indicator of its work assessed by the results of processing the control set.

The main optimized characteristics of the structure of a static NN with direct signal propagation include:

– the type of AF of neurons in each layer;
– the number of neurons in each layer;
– the number of neuron layers.

Quality of work (classification of the GTE TS) of the neural network is proposed to be assessed by the percentage of obtained erroneous diagnoses (classification errors)

$$\delta = \frac{N^E}{N} 100\,\%, \tag{1}$$

where N, N^E are the volume of the data set and the number of incorrect diagnoses obtained for this data set, respectively. The parameter is assessed separately for the results of the work of the trained NN using the training (δ^T parameter) and control (δ^C parameter) sets.

The method developed for optimizing the network structure consists of four stages.

Stage 1. Select AF of activating the network neurons. The data that do not contain measurement errors are used. The data sets are relatively small and include 100–200 realizations of each class. An NN is trained for each of the considered AF combinations. The experiment is carried out with a two-layer mesh. The number of neurons in the first layer gradually increases from the minimum value. The increase in the number of neurons continues until signs of overlearning appear or until the decrease in δ^T and δ^C values stops. An NN combination is selected that provides the smallest value of the δ^C parameter.

Stage 2. Determine the volume of the representative training set if this has not been done earlier. The approach used to obtain the data sets makes it possible to obtain the data sets that are practically unlimited in volume. At the same time, with an increase in the volume of the training set, the difference between values of the δ^T and δ^C parameters decreases significantly and the stability of the estimates obtained increases. On the other hand, the use of large training sets increases the time of the neural network training and may require a computer upgrade. At this and subsequent stages, the data containing measurement errors are used. To determine the volume of a representative training set, it is necessary to generate wittingly representative data sets. The control set is used completely during the experiment and the samples having gradually increasing sizes are taken from the complete training set. With each such sampling, the NN which has the structure and size of layers determined at the first stage is trained. The $(\delta^T-\delta^C)/\delta^C$ ratio was taken as a criterion for determining the minimum volume. It was assumed that the size of the training set ceases to have a significant effect on the diagnosis process at $(\delta^T-\delta^C)/\delta^C<0.01$. If the training set has already been fully used but the specified condition was not met, it is necessary to increase the sizes of the sets and repeat the experiment.

Stage 3. Determine the minimum value of the δ^C parameter $\left(\delta^C_{\min}\right)$ achievable for the designed network when using the available control set. Assessment is carried out with a network that has excess size. Such a network should be more complex and larger than the network obtained at the first stage. Redundancy is confirmed by the appearance of the overlearning effect. If the overlearning effect does not manifest itself, then the hypothesis of redundancy of the NN size should be confirmed at Stage 4. To determine the $\delta^C_{\min}$ value, such a network is trained 5 to 10 times. Training is carried out until the overlearning effect appears or until the network training stops. The mean $\delta^C_{\min}$ value is taken as the mean δ^C value. Further, the obtained value is used as a basis for assessing the quality of all tested networks.

Stage 4. Perform actual optimization of the NN structure. The NN that has a minimum number of layers (as a rule, two layers) and a minimum number of neurons in the first layers is taken as the initial configuration. The constructed network is trained. At the end of the NN training, the achieved value of the δ^C indicator is analyzed. The $\left(\delta^C-\delta^C_{\min}\right)/\delta^C_{\min}$ relation is used as a criterion. Fulfillment of the $\left(\delta^C-\delta^C_{\min}\right)/\delta^C_{\min}>0.01$ condition indicates the need to complicate the NN. This changes the number of neurons n in the first layer. The maximum number of neurons $n_{\max}$ is taken equal to the number of neurons at which the lowest value of the δ^C parameter was provided at Stage 1. After each change of the network structure, the attempt to train the network is repeated. An enlarged diagram of the algorithm is shown in Fig. 1.

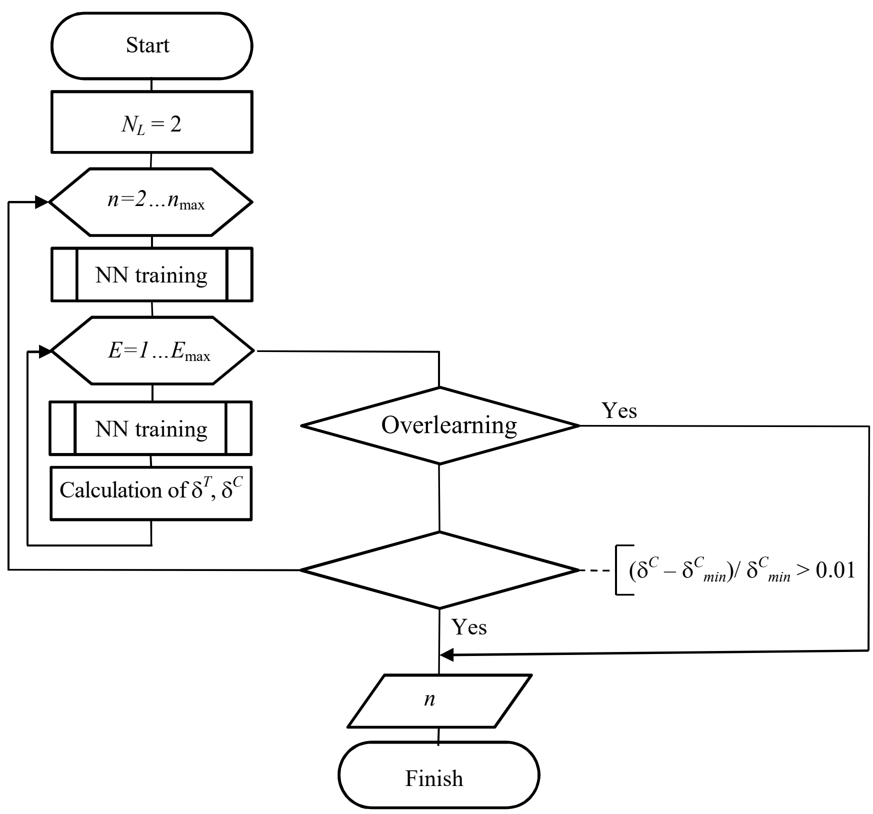


Fig. 1. An enlarged diagram of the algorithm of optimization of the NN layer size

The fact that an increase in the number of neurons in a layer does not lead to a decrease in the $\left(\delta^C - \delta^C_{\min}\right)/\delta^C_{\min}$, ratio indicates the need to increase the number of NN layers. If the number of layers is N_L>2, then the number of neurons in the intermediate layers n_2–$n_{NL\text{-}1}$ can be specified as an interpolation of their number in the first n and last n_{NL} layers. Further, after the optimal value of n is determined, the n_2–$n_{NL\text{-}1}$ values should also be sequentially optimized using the described algorithm. In this case, the $n_{\max}$ value for the optimized layer is equal to the number of neurons in the previous layer.

If the neural network which did not provide the appearance of the overlearning effect was used at Stage 3, then the hypothesis of the neural network size redundancy is confirmed if the neural network selected at Stage 4 is much simpler than that used at Stage 3 (it has a smaller number of layers and neurons in them). If the hypothesis is not confirmed, it is necessary to return to Stage3 and complicate the NN.

4. 2. The diagnosing object

PS-90A engine (USSR/Russia) [24] (2 rotors, bypass type, flow mixing, bypass ratio: 5, takeoff thrust: 155 kN) was selected as the diagnosing object.

The method of obtaining initial data for training and testing the neural network was described in [25, 26]. In addition, the study [25] provides a training set that was used at Stage 1 of this study. The test set was obtained in a similar way with new initial values of the pseudo-random number generators.

Diagnostics were carried out during takeoff and initial climb. Values of the operating parameters of the engine model were in the following ranges:

– barometric flight altitude H=(–100)...2,500 m above sea level;

– Mach number M=0...0.5;

– total temperature at the engine inlet: T^*_{in} = 238...313 K;

– relative air humidity: 0.3...1;

– fan rotor speed n_{LP}=3,280...4,220 rpm (nominal and takeoff modes).

The data sets describe the behavior of the GTEs belonging to 6 Classes of TS of the main elements of the flow path:

– serviceable engine (Class 1);

– contamination, increased roughness, nicks, etc. of the fan (Class 2) and compressor (Class 3);

– warpage of the combustion chamber, contamination, or burnout of injectors (Class 4);

– burnout, melting, partial destruction of high-pressure turbine blades (Class 5);

– coking, carbon formation on low-pressure turbine blades (Class 6).

Only the case of single faults was considered.

During the operation of the gas turbine engine, the parameters that can be divided into 2 groups were recorded. The first group includes the operating parameters of the mathematical model of the GTE: T^*_{in}, M, n_{LP} and total pressure P^*_{in}. at the engine inlet. When obtaining the sets, the value of the P^*_{in} parameter was calculated according to the given values of M and H. The second group consisted of the parameters used for diagnosing:

– high-pressure rotor speed, n_{HP};

– total pressure behind the fan, P^*_F;

– total pressure P^*_F and temperature T^*_C behind the compressor;

– fuel consumption, G_F;

– temperature behind the turbine, T^*_T;

– ratio of the total pressure behind the turbine to atmospheric pressure, $P^{*\prime}_T$.

Relative diagnostic deviations of the above parameters calculated from dependence (1) in [25] (Δn_{HP}, ΔP^*_F, ΔP^*_C, ΔT^*_C, ΔG_F, ΔT^*_T, ΔP^*_T , respectively) were used for diagnostics.

In addition to the values of diagnostic deviation of parameters, each design point of the data set included a vector of expected outputs of the neural network. For the case under consideration (the use of the NN for classification of the GTE TS), this vector included markers indicating which Class this set point belongs to. The number of the vector components corresponded to the number of classes in the data set (6 Classes).

4. 3. Description of the neural network used

As mentioned above, the study considered the optimization of neural networks with direct signal propagation. In this case, the minimum number of the NN layers was taken to be two. To train the neural network, the Levenberg-Marquardt algorithm was used [27–29].

The network is intended to perform the classification of the input data. The number of neurons n_{NL} in the output layer of the NN is equal to the number of diagnosed Classes (6 Classes in this case).

Each layer of neurons has a common AF (activation function). AFs of different layers may differ. When designing the NN, combinations of the following activation functions have been tested: linear function (linear, designated as Ln), logistic (Log-sigmoid, LS), hyperbolic tangent function (hyperbolic tangent sigmoid, HS) [8–10, 20, 29].

When using these functions, the signal at the outputs of neurons of the output layer changes smoothly. Its variation range is from 0 to 1 when using the LS function, from –1 to 1 when using the HS. The Ln function is unlimited. Taking this into account, in the part of the data set describing the expected outputs of the neural network, there will be 1 in the position corresponding to the expected class. In the remaining positions, there is either –1 (when using the HS function) or 0 (functions Ln and LS) depending on the type of the neuron activation function of the output layer.

The number of the neuron of the output layer which has the highest value at its output is taken as the class number of the GTE TS.

The network is trained using a representation of training accuracy according to the criterion of the mean squared deviation of training goals from the network response (mean squared error).

5. The results of neural network optimization

5. 1. Determination of the optimal combination of neuron activation functions for solving the problem

The task of this stage of the study is the selection of AF of the neurons that make up layers of the NN and the initial estimation of the number of layers and neurons in them which are necessary for TS classification. At this stage, the data sets obtained without taking into account the influence of measurement errors are used (without using dependences (9) to (18) in [25]). The initial data for training neural networks are given in the same study. The control data set was obtained in a similar way for other initial values of the pseudo-random number generators.

In the course of the study, 5 configurations of two-layer NNs were tested: Ln–Ln, LS–LS, HS–HS, HS–LS, LS–HS. The number n of neurons in the input layer varied from 2 to 20. There were 3 attempts to train each neural network. Training quality (δ^T and δ^C parameters) was assessed following each training epoch. The maximum number of epochs E_{max} depended on the number of neurons in the first layer and varied linearly from 140 with 2 neurons to 500 with 20 neurons. The training was carried out until the maximum number of epochs was exhausted. The results are shown in Fig. 2.

As can be seen from the data presented, in contrast to other networks, neural networks having neurons with the HS activation function in the output layer (Fig. 2, *c*, *d*), show the instability of the training process. However, at the same time, if the process is completed successfully, the HS–HS network provides the best values of the δ^T and δ^C parameters. The best results for such a network were obtained with 15 neurons in the input layer. A further increase in their number did not lead to an improvement in training quality (the δ^C parameter has stopped decreasing).

Considering all of the above, the networks that included only neurons with AF HS were used for further studies.

5. 2. Determining the optimal training set size

As noted earlier, when training the neural network, data that did not contain measurement errors were used. The final stage in the creation of a diagnostic neural network implies its optimization using data close to real ones and containing measurement errors in the parameters used. Therefore, to conduct the study, it is necessary to obtain sets containing measurement errors and assess the minimum size of a representative training set using them.

5. 2. 1. Forming the training and control sets of parameters containing measurement errors

When forming data sets, the method described in [25] was used. In this case, in contrast to the data sets given in this study, dependences (9) to (13) given in the same study were additionally involved in the same study. Values of maximum errors of measurement of operating [ΔR] and diagnostic [ΔP] parameters taken in modeling are given in Table 1.

The data obtained as a result of modeling are shown in Fig. 3.

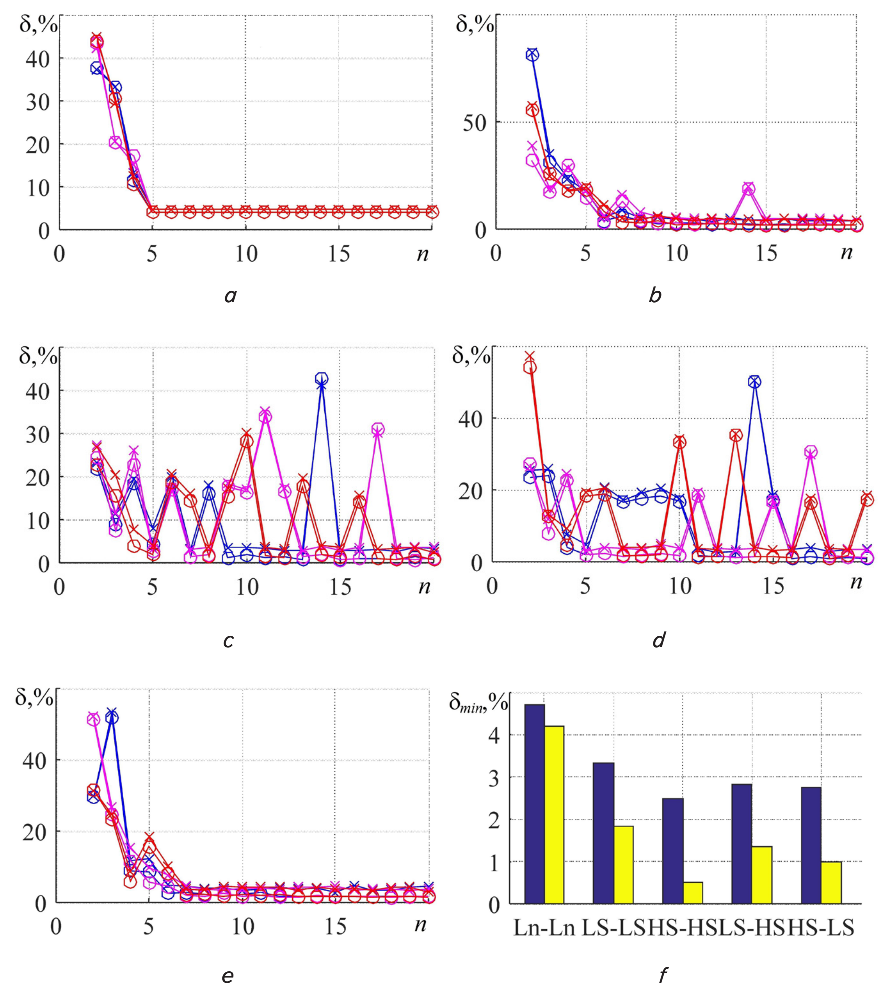


Fig. 2. Dependence of values of the TS, δ, classification error on the number of neurons n in the first layer of the two-layer NN: *a* – The NN configurations: Ln–Ln ; *b* – LS–LS ; *c* – HS–HS ; *d* – LS–HS; *e* – HS–LS (–⊖– is for quality of NN training δ^T estimated for the training sample; –×– is for quality of NN training δ^C estimated for the control sample; ——— is for the first attempt of the NN training; ——— is for the second attempt; ——— is for the third attempt); the best δ_{min} values of δ^T ( ) and δ ( ) parameters achieved at the moment the parameter δ^C reaches the minimum value (*e*)

Table 1

Values of the [ΔR] and [ΔP] parameters for the measured parameters of the PS-90A engine operating process

Parameter	[ΔR]				[ΔP]						
	P^*_{in}	T^*_{in}	M	n_{LP}	n_{HP}	P^*_F	P^*_C	T^*_C	T^*_T	$P^{*\prime}_T$	G_F
Units	kPa	K	–	rpm	rpm	kPa	kPa	K	K	–	kg/h
Measurement error	1.1	0.11	0.02	7.5	20	5	83	5.5	4.5	0.035	200

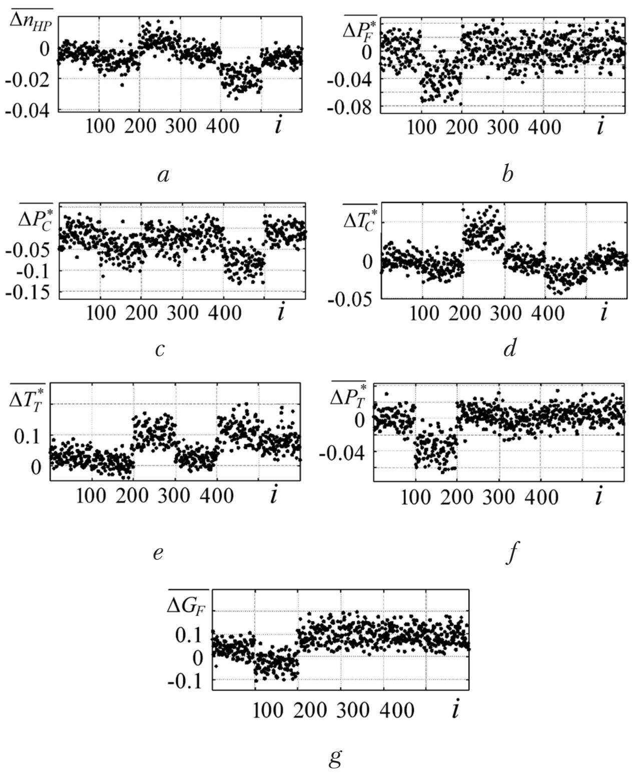


Fig. 3. Values of relative diagnostic deviations in the set intended for training the network to recognize 6 classes of TS of aircraft GTEs. Each class is represented by 100 points. i is the number of the point in the set. Relative deviations of the following parameters are shown: a — high-pressure rotor speed $\overline{\Delta n_{HP}}$; b — total pressure downstream of the fan $\overline{\Delta P^*_F}$; c — total pressure downstream of the compressor $\overline{\Delta P^*_C}$; d — total temperature downstream of the compressor $\overline{\Delta T^*_C}$; e — fuel consumption $\overline{\Delta G_F}$; f — temperature downstream of the turbine $\overline{\Delta T^*_T}$; g — the ratio of the total pressure downstream of the turbine to atmospheric pressure $\overline{P^*_T}$

As can be seen, when comparing these data and the data given in [25], the introduction of measurement errors into numerical experiment leads to a significant increase in the overlap of classes in the zone of their delimitation.

5. 2. 2. Determining the optimal size of the training set

At the next stage of the study, it is necessary to establish the optimal size of the training set. In the process of optimizing the sample size, a two-layer HS–HS network with 15 neurons in the first layer was used. Training and control sets were obtained in which each class was represented by 10,000 points. The control kit was used completely in each experiment. Samples of size V from 100 to 5,000 points per class were made from the obtained training set. The samples were trained three times each. The results of the best attempts are shown in Fig. 4, *a*, *b*.

As indicated earlier, it was assumed that the size of the training set ceases to have a significant impact on the diagnostic process at $(\delta^T-\delta^C)/\delta^C<0.01$.

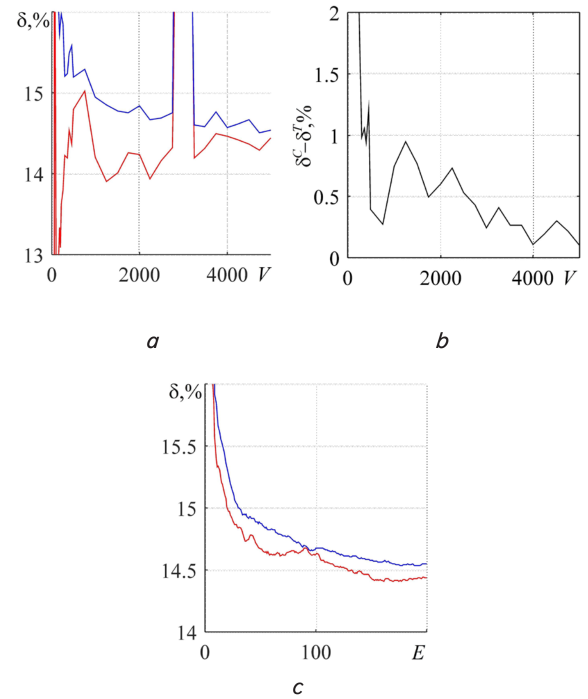


Fig. 4. Results of optimization of the size of the training set: a — dependence of TS classification error values, δ, estimated by training (δ^T, ———) and control (δ^C, ———) samples on size of the training sample V representing each TS class; b — dependence of the value difference ($\delta^T-\delta^C$) on the sample size V; c — change in values of parameters (δ^T (———) and δ^C (———) in the process of training the neural network (E is number of the training epochs)

As seen from Fig. 4, *a*, the difference between values of δ^T and δ^C parameters is in the range of 1–3 % (the ratio $(\delta^T-\delta^C)/\delta^C$ is 0.07–0.2) with small sizes of the training sample (up to 1,000–2,000 points per class). With an increase in the volume of more than 3000 points, this difference steadily decreases to about 0.1–0.14 %. The ratio $(\delta^T-\delta^C)/\delta^C$ is 0.007–0.0096.

It can be concluded from all that has been said that with the adopted characteristics of the TS classes (Table 2 in [25]) and metrological characteristics of the parameter measurement system given in Table 1, the training set can be considered representative if each class is represented by 4,000 or more points.

Fig. 4, *c* shows the process of training neural networks for 200 epochs using a training set in which each class is represented by 4,000 points. Analysis of these data shows that there are no clear signs of overlearning. In this case, values of the δ and δ^C parameters practically cease to change after 150 training epochs.

5. 3. The results obtained by optimizing the structure of the neural network

5. 3. 1. Determination of the $\delta^C_{\min}$ parameter value

Before starting the actual optimization, it is necessary to determine the minimum value of δ^C which can be reached by NN of the type used $\left(\delta^C_{\min}\right)$ when analyzing the data described in Section 5. 2. 1. To this end, nine attempts were made to train a three-layer HS–HS neural network which has 25 neurons in the first layer and 15 in the second layer during 1,000 epochs. The training was unsuccessful in 3 attempts (values of δ^T and δ^C parameters were above 80 %). In other attempts, the NN stopped training after completing 600–800 epochs. Fig. 5 shows the change in δ^C and δ^C parameters during the best training attempt (δ^C=14.51 %).

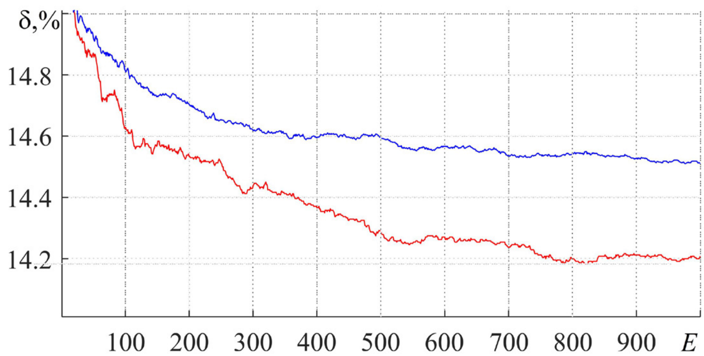


Fig. 5. Change in parameters δ^C (———) and δ^C (———) in the course of the best attempt to train a three-layer HS–HS–HS neuron network (*E* is the number of training epochs)

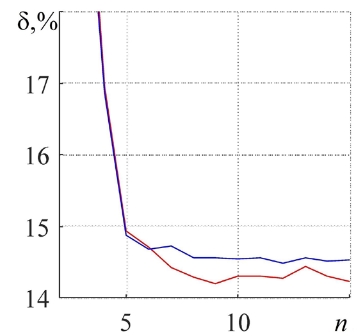
The value of the δ^C parameter obtained in the course of successful training attempts was at a level of 14.51–14.7 % and the value of $\delta^C_{\min}$, calculated from them was 14.55 %.

5. 3. 2. The results of neural network optimization

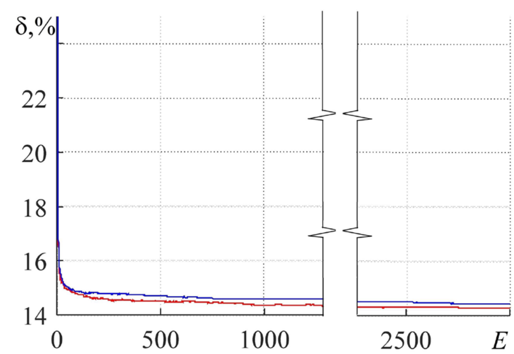
When optimizing, a two-layer HS–HS network containing from 2 to 15 neurons in the first layer was considered. Duration of network training increased in proportion to the number of neurons in the input layer. The network with 2 neurons was trained during 500 epochs and the network with 15 neurons was trained during 930 epochs. The training was carried out without taking into account the $\delta^C_{\min}$. value. For each network, 3 training attempts were made. The results of the best attempts are shown in Fig. 6, *a*.

As can be seen, the δ^C parameter decreases quite steadily with an increase in the number of neurons up to 8. With a further increase in the network size, the quality of its work practically did not change remaining at a level of 14.48–14.55 %.

To check the influence of duration of the network training on the result of its operation, 8 attempts were additionally made to train the HS–HS network with 8 neurons of the first layer. The training was conducted during over 3,000 epochs. As a result, the values of the δ^C parameter were obtained in the range of 14.43–14.51 %. At the same time, 3 out of 8 attempts ended in failure when the δ^C parameter level was above 29 %. The average value of the parameter calculated from successful attempts was 14.47 %. Fig. 6, *b*, *c* shows a successful training process for such a network. As can be seen, the network reached the level of 14.58 % for δ^C values after 1000 training epochs, and then the training process did not lead to a significant change in this parameter. At the same time, there were no unambiguous signs of overlearning. The data obtained indicate that a neural network of this size actually stopped training after 1,000 epochs when the value of the δ^C parameter was in the range of 14.43–14.51 %.



a



b

Fig. 6. Results of neural network training: *a* – dependence of parameters δ^T (———) and δ^C (———) on the number of neurons *n* in the first layer of a two-layer neural network; *b* – change in values of parameters δ^T (———) and δ^C (———) in the process of training a two-layer neural network with 8 neurons in the first layer (*E* is the number of training epochs)

The value of $\delta^C_{\min}$, obtained at Stage 3 was greater than the average value of the δ^C parameter obtained in this experiment (14.47 %). This confirms the hypothesis about the redundancy of size of the NN used at the third stage and indicates the inexpediency of transition to a NN with a three-layer structure.

An attempt was made at the final stage of NN optimization to reduce the obtained value of the δ^C parameter. Therefore, an attempt was made to train a two-layer HS–HS network with 8 neurons in the first layer using a vector of training goals varying within –0.5–0.5 and 0–1. Such a change in the target vector did not provide a decrease in the value of the δ^C parameter.

6. Discussion of the results obtained during the optimization of the neural network

The developed method is based on the use of a mathematical model of an engine for the formation of training and control samples of any size. This circumstance has made it possible to determine the minimum volume of the training set which can be considered representative. The results obtained show (Fig. 4) that each class in such a set should be represented by at least 4,000 points. This large volume of training data is explained by the need for a fairly complete presentation of various combinations of engine assemblies having various functional characteristics within the same TS class.

When choosing the optimal combination of neuron activation functions, it was shown that it is best to use a hyperbolic tangent function in both layers (Fig. 2). It should be borne in mind that the use of such a combination leads quite often (in about 30 % of cases) to the impossibility of NN training with an acceptable level of accuracy (Fig. 2, *d*). A possible solution to this problem may imply changing the training algorithm used.

A rather high proportion of erroneous diagnoses (about 14.5 %, Fig. 6) which in turn is a consequence of the presence of errors in measuring the parameters which are used in diagnostics is one of the main problems of using the proposed approach. Various averaging and filtering methods can be used to minimize the impact of errors.

Static neural networks with direct signal propagation and fixed interneuron connections were used in the study. In the future, in order to reduce the proportion of incorrect diagnoses, it is planned to test dynamic, recurrent, radial-basis, and adaptive neuro-fuzzy neural networks [30].

The neuron number which has the highest value at the output is also taken in the considered method as the diagnosis number. In this case, outputs of the remaining neurons are not checked for the proximity of their output signals to the signal of the winning neuron, and proximity to unity of the value at its output is not analyzed. Such an analysis would make it possible to single out doubtful diagnoses into a separate group for in-depth analysis.

The approach used provides for a diagnosis in the form of assigning an engine to a specific TS class. This approach greatly simplifies work for an expert in the express analysis of the results obtained but does not provide information on the degree of defect development. In addition, when the characteristics of the classes change, the network must be retrained for correct TS classification. A transition from direct TS classification to the definition of parameters that numerically characterize the TS nodes can be a possible solution to this problem. For example, values of displacement of functional characteristics of a particular node relative to the reference can be used as such parameters.

Another problem of the considered method is the use of diagnostic deviations (dependences (1) and (2) in [25]) as input information of the NN. The use of this representation of diagnostic parameters makes it possible to significantly simplify and speed up the process of preparing the NN due to the fact that there is no need to include the GTE operating parameters in the data sets. However, in its turn, this leads to the fact that in order to calculate diagnostic deviations, it is necessary to include a block for calculating parameters of a standard engine into the diagnostic algorithm. Therefore, it is necessary to consider the option of diagnosing directly by the measured parameters as a prospect for the development of the proposed approach.

To conclude, it should be pointed out that the considered approach has been tested on the sets representing single defects. Here, the case of the simultaneous appearance of several defects of the flow path is quite frequent. According to the data given in [31], more than one defective unit was found in approximately 9–12 % of cases of detecting defects in the flow path. In its turn, this indicates that the classes describing the operation of an engine having significant defects simultaneously in two or more assemblies should be included in the data sets.

7. Conclusions

1. A method of assessing the minimum size of the training and control samples has been developed. The method implies training a neural network of large size which ensures a guaranteed manifestation of the overlearning effect for solving the task at hand. Training is carried out up to a stable appearance of the overlearning effect and determination of the difference between δ^T and δ^C. The training is carried out using a sufficiently large control set. The size of the training set is considered representative if the value of the $\delta^T-\delta^C$ difference obtained with its use is significantly less than the minimum value of δ^C obtained at the moment the parameter δ^C reaches its minimum value. When checking the method, it was found that the minimum size of the training sample should be 4,000 points per class for a given TS nomenclature and with an available system of measuring and recording the engine operating parameters. In this case, the ratio $(\delta^T-\delta^C)/\delta^C$ is about 0.01.

2. A method of determining the optimal combination of neuron activation functions has been developed. The method implies testing the networks that have various neuron activation functions. The best is the network that provides the lowest value of the δ^C parameter. Network training is carried out with the sets having parameters with no measurement errors. It was shown for the considered case that the HS–HS combination is the best combination of neurons.

3. A method of optimizing the neural network size has been developed. The method implies testing the networks of various configurations and assessing the value of the δ^C parameter. In this case, such a limit of network complexity is sought that the δ^C parameter value ceases to decrease after exceeding this limit. The search is carried out with the sets having parameters with measurement errors. For the considered case, a two-layer network with 8 neurons in the input layer is the optimal choice.

References

1. Sutskever, I., Martens, J., Dahl, G., Hinton, G. (2013). On the importance of initialization and momentum in deep learning. Proceedings of the 30th International Conference on Machine Learning, 28 (3), 1139–1147. Available at: http://proceedings.mlr.press/v28/sutskever13.html
2. Krutikov, V. N., Samoylenko, N. S. (2018). On the convergence rate of the subgradient method with metric variation and its applications in neural network approximation schemes. Vestnik Tomskogo Gosudarstvennogo Universiteta. Matematika i Mekhanika, 55, 22–37. doi: https://doi.org/10.17223/19988621/55/3

3. Brownlee, J. (2018). How to Avoid Overfitting in Deep Learning Neural Networks. Available at: https://machinelearningmastery.com/introduction-to-regularization-to-reduce-overfitting-and-improve-generalization-error/
4. Denil, M., Shakibi, B., Dinh, L., Ranzato, M., De Freitas, N. (2014). Predicting Parameters in Deep Learning. arXiv.org. Available at: https://arxiv.org/pdf/1306.0543.pdf
5. Han, S., Pool, J., Tran, J., Dally, W. J. (2015). Learning both Weights and Connections for Efficient Neural Networks. arXiv.org. Available at: https://arxiv.org/pdf/1506.02626v3.pdf
6. Patrick, E. A. (1972). Fundamentals of pattern recognition. Prentice-Hall, Inc., 504.
7. Patrik, E.; Levin, B. (Ed.) (1980). Osnovy teorii raspoznavaniya obrazov. Moscow: Sov. radio, 408. Available at: http://padaread.com/?book=19276
8. Rashid, T. (2016). Make your own neural network. CreateSpace, 222.
9. Medvedev, V. S., Potemkin, V. G.; Potemkin, V. G. (Ed.) (2002). Neyronnye seti. MATLAB 6. Moscow: DIALOG-MIFI, 496.
10. Medvedev, V. S., Potemkin, V. G. (2001). Neyronnye seti. MATLAB 6. Pakety prikladnyh programm. Kn. 4. Moscow: DIALOG-MIFI, 630.
11. Lo Gatto, E., Li, Y. G., Pilidis, P. (2006). Gas Turbine Off-Design Performance Adaptation Using a Genetic Algorithm. Volume 2: Aircraft Engine; Ceramics; Coal, Biomass and Alternative Fuels; Controls, Diagnostics and Instrumentation; Environmental and Regulatory Affairs. doi: https://doi.org/10.1115/gt2006-90299
12. Sampath, S., Ogaji, S., Singh, R., Probert, D. (2002). Engine-fault diagnostics: an optimisation procedure. Applied Energy, 73 (1), 47–70. doi: https://doi.org/10.1016/s0306-2619(02)00051-x
13. Li, Y. G. (2002). Performance-analysis-based gas turbine diagnostics: A review. Proceedings of the Institution of Mechanical Engineers, Part A: Journal of Power and Energy, 216 (5), 363–377. doi: https://doi.org/10.1243/095765002320877856
14. Ntantis, E. L., Botsaris, P. N. (2015). Diagnostic Methods for an Aircraft Engine Performance. Journal of Engineering Science and Technology Review, 8 (4), 64–72. doi: https://doi.org/10.25103/jestr.084.10
15. Ross, T. J. (2010). Fuzzy logic with engineering applications. John Wiley & Sons. doi: https://doi.org/10.1002/9781119994374
16. Yang, M., Shen, Q. (2008). Reinforcing fuzzy rule-based diagnosis of turbomachines with case-based reasoning. International Journal of Knowledge-Based and Intelligent Engineering Systems, 12 (2), 173–181. doi: https://doi.org/10.3233/kes-2008-12208
17. Ogaji, S. O. T., Li, Y. G., Sampath, S., Singh, R. (2003). Gas Path Fault Diagnosis of a Turbofan Engine From Transient Data Using Artificial Neural Networks. Volume 1: Turbo Expo 2003. doi: https://doi.org/10.1115/gt2003-38423
18. Angeli, C., Chatzinikolaou, A. (2004). On-Line Fault Detection Techniques for Technical Systems: A Survey. International Journal of Computer Science & Applications, I (1), 12–30. Available at: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.100.6189&rep=rep1&type=pdf
19. Agneli, C. (2010). Chap. 4. Diagnostic Expert Systems: From Expert's Knowledge to Real-Time Systems. Vol. 1. TMRF e-Book, Advanced Knowledge Based Systems: Model, Applications & Research, 1, 50–73. Available at: http://www.tmrfindia.org/eseries/ebookv1-c4.pdf
20. Asgari, H., Chen, X. (2015). Gas Turbines Modeling, Simulation, and Control. CRC Press, 206. doi: https://doi.org/10.1201/b18956
21. Osigwe, E., Li, Y.-G., Suresh, S., Jombo, G., Indarti, D. (2017). Integrated Gas Turbine System Diagnostics: Components and Sensor Faults Quantification using Artificial Neural Network. 23rd International Society of Air Breathing Engines (ISABE) Conference – ISABE 2017. Available at: https://www.researchgate.net/profile/Emmanuel_Osigwe/publication/319645027_Integrated_Gas_Turbine_System_Diagnostics_Components_and_Sensor_Faults_Quantification_using_Artificial_Neural_Network/links/59e52ae90f7e9b0e1aa888f0/Integrated-Gas-Turbine-System-Diagnostics-Components-and-Sensor-Faults-Quantification-using-Artificial-Neural-Network.pdf?origin=publication_detail
22. Loboda, I. (2010). Gas Turbine Condition Monitoring and Diagnostics. Gas Turbines, 119–144. doi: https://doi.org/10.5772/10210
23. Loboda, I., Feldshteyn, Y., Ponomaryov, V. (2012). Neural Networks for Gas Turbine Fault Identification: Multilayer Perceptron or Radial Basis Network? International Journal of Turbo & Jet-Engines, 29 (1). doi: https://doi.org/10.1515/tjj-2012-0005
24. Nihmakin, M. A., Zal'tsman, M. M. (1997). Konstruktsiya osnovnyh uzlov dvigatelya PS-90A. Perm', 92.
25. Kulyk, M., Abdullayev, P., Yakushenko, O., Popov, O., Mirzoyev, A., Chumak, O., Okhmakevych, V. (2018). Development of a data acquisition method to train neural networks to diagnose gas turbine engines and gas pumping units. Eastern-European Journal of Enterprise Technologies, 6 (9 (96)), 55–63. doi: https://doi.org/10.15587/1729-4061.2018.147720
26. Kulyk, M., Dmitriev, S., Yakushenko, O., Popov, O. (2013). Method of formulating input parameters of neural network for diagnosing gas-turbine engines. Aviation, 17 (2), 52–56. doi: https://doi.org/10.3846/16487788.2013.805868
27. Hagan, M. T., Menhaj, M. B. (1994). Training feedforward networks with the Marquardt algorithm. IEEE Transactions on Neural Networks, 5 (6), 989–993. doi: https://doi.org/10.1109/72.329697
28. Wilamowski, B. M., Yu, H. (2010). Improved Computation for Levenberg-Marquardt Training. IEEE Transactions on Neural Networks, 21 (6), 930–937. doi: https://doi.org/10.1109/tnn.2010.2045657
29. Wilamowski, B. M., Irwin, J. D. (Eds.) (2018). Intelligent Systems. CRC Press, 610. doi: https://doi.org/10.1201/9781315218427
30. ANFIS. Available at: https://ru.m.wikipedia.org/wiki/ANFIS
31. Popov, A. V. (2007). Issledovanie dinamicheskih harakteristik TRDD s peremezhayushchimisya neispravnostyami protochnoy chasti na ustanovivshihsya rezhimah ego raboty. Aviatsionno-kosmicheskaya tehnika i tehnologiya, 2 (38), 63–67.