Optimization of Helicopters Aircraft Engine Working Process Using Neural Networks Technologies

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

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

Abstract

In this work, the multicriteria optimization method according to the NSGA-II algorithm was further developed on the basis of approximate models of the object under study – helicopters turboshaft engine, which made it possible to optimize the parameters of its working process in the helicopter flight mode. An approximate model of the helicopter turboshaft engine is developed on the basis of a radial basis neural network (RBF networks), the parameters of which are determined using an evolutionary algorithm. The practical significance of the obtained results of the work lies in the use of a neural network unit for optimizing the parameters of the working process of helicopter turboshaft engine as part of the on-board system for monitoring its operational status.

Keywords

Helicopter aircraft engine, neural network, radial basis neural network (RBF-network), multiobjective optimization method, evolutionary algorithm, thermogasodynamic parameters.

1. Introduction

The development and operation of complex technical systems at the modern level involves the mandatory use of their mathematical models, which can be defined as a mathematical "image of the essential aspects of a real system or its design in a convenient form that reflects information about the system" [1].

Existing optimization methods [2] make it possible by calculation to find the most effective combination of product parameters before starting to manufacture prototypes. It is especially important to use multi-criteria optimization [2], which leads, however, to a significant increase in the number of calculations performed. Very often, the connections between objective functions and independent variables are described by systems of partial differential equations or integro-differential equations, the solution of which can only be obtained by numerical methods, as well as empirical dependencies in the form of tables, and by their nature these dependencies are multidisciplinary in nature.

With regard to helicopters turboshaft engines (TE) and its modifications, which is a complex technical system, during its creation and operation a large number of mathematical models of different types of the engine as a whole and its individual components are developed. These are models of stress-strain, thermal status of blades, disks, rotors and other elements of the compressor, compressor turbine, combustion chamber, free turbine, etc.; thermogasdynamic model describing the working process in the engine elements, i.e., the relation between pressure, temperature, air and gas flow at different points of the engine duct, and other models.

COLINS-2022: 6th International Conference on Computational Linguistics and Intelligent Systems, May 12–13, 2022, Gliwice, Poland
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)

The importance of mathematical models of helicopter gas turbine engines as an object of regulation in the processes of developing, creating and adjusting engines is constantly growing. This is determined by a number of objective factors, the main of which are the following:

– complication of schemes, designs and fabrication methods of engines, increase in cost of design materials and, as a result, very high cost of field tests. At the same time, it is practically impossible to carry out full-scale tests in all operating conditions characteristic of multimode engines;

– ability to create high-precision and fairly fast-acting mathematical models of engines that adequately describe their working process at different flight conditions.

At the stage of flight operation of helicopters, the methods of direct numerical optimization require a significant number of algorithmic calculations, which in the presence of even modern computers does not allow to perform all necessary calculations in a given time. The method of analytical optimization of objective functions is devoid of this drawback. Based on the set number of calculations, it is possible to form a transfer model with a given accuracy for a complete assessment.

Helicopters engines (including TV3-117) – GTE with a free turbine, is a subsystem of a more complex aircraft system. The working process of turboshaft engines with a free turbine is determined by several dozen parameters. Although this complex is quite large, the choice of a significant part of the parameters from it (σ_{in}, σ_{cc}, η_T^*, η_C^*, φ_{out} etc.) for the calculation mode is carried out within such a narrow range that the assessment of their most probable values is usually not particularly difficult. The values of such parameters required for the calculation are not optimized, but predicted [1]. Therefore, only those operating process parameters that determine a closed system of equations of thermogasodynamic calculation of the engine and can vary in a wide range of values are selected for optimization.

The number of optimized parameters depends primarily on the type of gas turbine engine. In the working process of helicopters turboshaft engines with a free turbine (including TV3-117), as is known, the problem of free energy distribution between the main rotor and the exhaust unit is not relevant, so here we are talking about optimization or only one parameter – π_C^* in the case of the selected level T_G^* when the design and technological level of the "hot" part of the engine is reached), or two work process parameters – T_G^* and π_C^*, if the temperature of the turbine parts is set to obtain the most favorable (rational) indicators of the subsystem.

Calculations based on the finite element method (FEM) used in modern practice require a significant amount of computer time. According to the experimental results, for the calculation of the TV3-117 TE at the I cruise mode (at a constant rotor r.p.m.), it is required to perform 4.6×10^{17} floating point operations. Using a supercomputer with a performance of 0.6 teraflops, this calculation can be performed in 14 days, provided there are no losses when parallelizing tasks. In case of multiobjective optimization, such calculations must be repeated 1000 or more times. Thus, it is very important to use approximate models of optimized designs, which can significantly reduce the required number of calculations.

2. Literature review

Optimization of the parameters of dynamic systems is given in [3–5]. The method of multicriteria optimization based on approximate models for dynamic systems was developed by Yuri Zelenkov in [6]. However, the implementation of this method is possible only at the design stage of an aircraft engine. Therefore, the actual scientific and practical task solved in this work is the improvement of this method in relation to helicopters engines at the mode of their flight operation, that is, in real time.

Since the dependences of the efficiency evaluation criteria on the workflow parameters are close to quadratic [7], it is advisable to choose a second-order model, which is an elliptical paraboloid, as an approximating surface. To solve the problem of approximation, it is advisable to choose the least squares method (LSM) [8], due to the simplicity of its implementation. It is possible to use robust methods of evaluating the results of the calculation experiment, which reduce the number of gross errors of the experiment. The regression model modeled by LSM has the form [9]:

$$y = ax_1^2 + bx_2^2 + cx_1x_2 + dx_1 + ex_2 + f; \qquad (1)$$

where x_1 – an independent variable that corresponds to the level of pressure increase in the compressor π_C^*; x_2 – independent variable corresponding to the gas temperature in front of the compressor turbine T_G^*; a, b, c, d, e, f – model coefficients determined by LSM.

Finding the partial derivatives of the function y, determine its minimum (maximum) and its corresponding values x_1 and x_2 according to the following system of equations:

$$\begin{cases} y'_{x_1} = 2ax_1 + cx_2 + d = 0 \\ y'_{x_2} = 2bx_2 + cx_2 + e = 0 \end{cases} \tag{2}$$

After determining the partial derivatives of the equations of the functions, you can find the values of independent variables in which the functions have a minimum (maximum), and then calculate the minimum (maximum) value of the function, which will be the optimum for this function.

To optimize one-parameter problems, this approach allows the use of functions $y = f\left(T_G^*\right)$ or $y = f\left(\pi_C^*\right)$ (fig. 1).

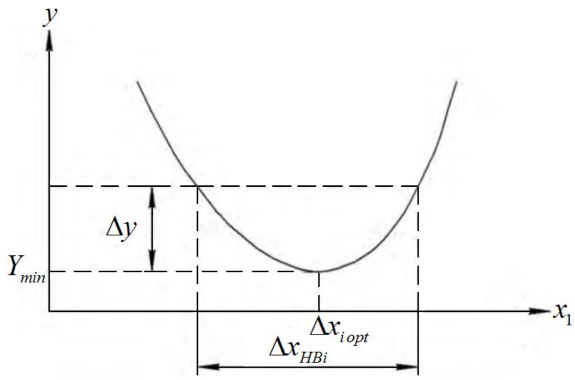

Figure 1: Diagram of formation of the range of the most favorable parameters for one-parameter problems

Fig. 2 shows the formation of a region of rational values of parameters for a two-parameter problem.

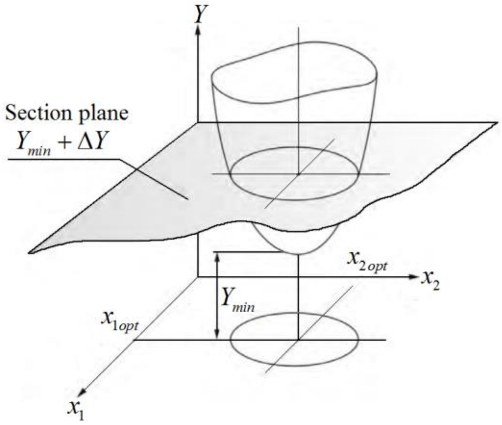

Figure 2: Diagram of obtaining the range of rational values of parameters for two-parameter problems

The joint solution of the equation of the objective function and the plane distant from the extrema by the value ΔY allows to obtain in the projection on the plane $x_1 - x_2$ (π_C^* – T_G^*) for each criterion function Y_i a closed line close to the ellipse. These lines are actually outside the ranges of rational values of workflow parameters.

As in [6], consider in general the problem of multicriteria minimization with m independent variables, n targets, p constraints in the form of inequalities, and q constraints in the form of equalities [2]: «minimize $f(x)$ provided $g(x) > 0$, $h(x) = 0$», where $x = (x_1 \ldots x_m) \in X$ – vector of solutions (independent variables), X – parameter space, $f(x)^T = [f_1(x)\ldots f_n(x)]$ – purposes, $g(x)^T = [g_1(x)\ldots g_p(x)]$ – inequality constraints, $h(x)^T = [h_1(x)\ldots h_q(x)]$ – equality constraints. Vector of solutions $a \in X$ is dominant over a vector $b \in X$, that is $a \prec b$, if the condition

$$\forall i \in \{1,\ldots,n\} : f_i(a) \le f_i(b) \wedge \exists j \in \{1,\ldots,n\} : f_j(a) < f_j(b). \tag{3}$$

Vector a is called non-dominated on a set $X' \subseteq X$, if there X' is no vector, dominating a. A set of solutions X', for which the condition is met:

$$\forall a' \in X' : \neg\exists a \in X : a \prec a' \wedge \|a - a'\| < \varepsilon \wedge \|f(a) - f(a')\| < \delta. \tag{4}$$

where $\|\ldots\|$ – distance metric, at $\varepsilon > 0$, $\delta > 0$ is called a local Pareto optimal set. X' is a global Pareto optimal set if $\forall a' \in X' : \neg\exists a \in X : a \prec a'$ [10].

Thus, the problem of multicriteria optimization is the problem of finding a global Pareto optimal set of solutions. At the stage of flight operation of helicopters TE, this set is presented to an expert (aircraft crew commander), who chooses one of the laws of regulation and, as a consequence, further options for continuing the flight.

Analysis of even simplified methods of thermogasdynamic calculation of aircraft GTEs, including helicopters TE, [11] shows that more than 30 parameters (independent variables) affect the definition of the working process and, consequently, the design of the engine. In this case, the dependences linking the target and independent variables are nonlinear, and it is impossible to guarantee that they are differentiable functions. Today, a number of multicriteria optimization methods are known based on nonlinear programming [12] and genetic algorithms [13, 14]. One of the most efficient constrained multicriteria optimization algorithms is the NSGA-II genetic algorithm [15, 16]. A feature of this algorithm is that at each step of the calculations, a new population of N solutions is generated, for each of which the functions $f(x)$, $g(x)$, and $h(x)$ must be calculated. A population of 100 solutions is typical and evolves over 500 generations. It is easy to estimate that in this case 50000 calculations of the functions $f(x)$, $g(x)$, and $h(x)$ are required. Thus, based on practical considerations, in order to reduce the time spent to reasonable limits, it is necessary to propose a method for finding the Pareto-optimal set of solutions in no more than 500 calculations of expressions of exact models of the investigated dependencies. To achieve this purpose, it is proposed to use an approach based on using, instead of the multicriteria minimization problem, their approximate models.

3. Problem statement

The generalized mathematical formulation of the multicriteria optimization problem of working process of helicopters TE at flight modes assumes that the feasible solutions set (FSS) $X = \{x\}$ is given, on which the "vector objective function" (VOF) is defined:

$$F(x) = \langle F_1(x), F_2(x), \ldots, F_n(x) \rangle; \tag{5}$$

whose criteria, for definiteness, we will assume to be minimized:

$$F_v(x) \to \min, v = 1\ldots N. \tag{6}$$

A feasible solution $x \in X$ is called Pareto-optimal or Pareto optimum if there is no such element $x \in X$ that satisfies the inequalities $F(x) \le F(x)$, $F(x^*) \ne F(x)$, and X' is a Pareto set consisting of all Pareto optima of the problem under consideration with VOF (5)–(6) and FSS X. This problem is called discrete if the power of the FSS $|X|$ is finite. Thus, the essence of the task of the multicriteria optimization problem of working process of helicopters TE at flight modes is to find one or all elements of the Pareto set.

4. Multi objective optimization method

Consider the calculation algorithm given in [6] (fig. 3). The investigated dependencies are interdisciplinary; it is impossible to guarantee the differentiability or convexity of these functions. In addition, when developing new products, the task is never to find the best parameters of a new product (i.e., the global Pareto optimal set), and all efforts are directed only to ensuring the specified technical requirements. Very often a solution is chosen with less efficient parameters, but providing greater resistance to deviations that inevitably arise during the production process. Third, an excessively large Pareto-optimal set requires a significant investment of time and resources to analyze all alternative solutions; it is quite acceptable to have 15...20 options for product parameters during its operation. Therefore, the proposed multicriteria optimization method uses heuristic approaches (evolutionary and genetic algorithms).

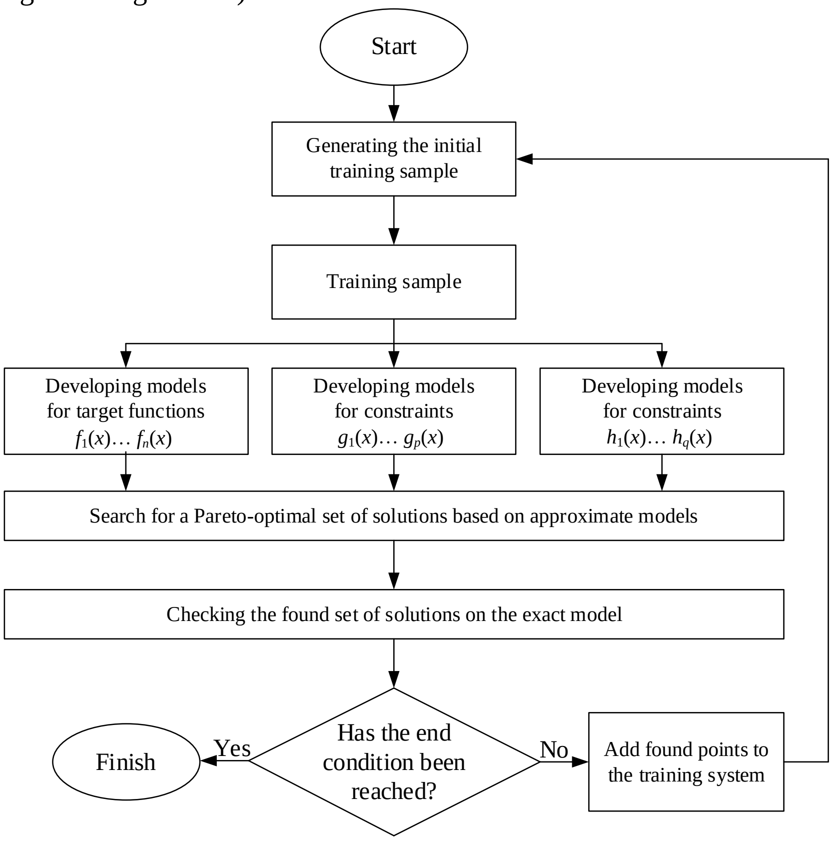

Figure 3: Algorithm of the method of multicriteria optimization [6]

The described method (fig. 3) consists of the following steps.

1. An initial training sample x_s of a small size $s \in X$ is generated based on one of the experiment planning methods. The vectors of the values of the objective functions $f(x_s)$ and the constraints $g(x_s)$ and $h(x_s)$ are calculated at all obtained points.

2. Based on the training sample x_s and the corresponding values $f(x_s)$, $g(x_s)$ and $h(x_s)$ approximate models are being developed $f(x)$, $g(x)$ and $h(x)$ of all investigated dependencies.

3. Based on the obtained approximate models $f(x)$, $g(x)$ and $h(x)$ using the NSGA-II algorithm, the vector x_{opt} is found, which determines the Pareto-optimal set of solutions to the multicriteria optimization problem.

4. At the points of the set of solutions x_{opt} obtained in this way, the exact values of the functions $f(x_{opt})$, $g(x_{opt})$ and $h(x_{opt})$. If the termination condition is not met, then all the values obtained on the exact models are added to the training set:

$$x_s = x_s + x_{opt};\; f(x_s) = f(x_s) + f(x_{opt});\; g(x_s) = g(x_s) + g(x_{opt});\; h(x_s) = h(x_s) + h(x_{opt}). \quad (7)$$

5. The return to step 2 is carried out, at which the approximate models are built again.

6. The following conditions for the end of calculations are determined:

– the total relative error e of the constructed models reaches a given minimum [6]:

$$e = \frac{1}{k \cdot (n+p+q)} \cdot \sum_{j=1}^{n+p+q} \sqrt{\sum_{i=1}^{k} \left(\frac{M_{ij}(x) - F_{ij}(x)}{F_{ij}(x)} \right)^2} \le \varepsilon. \quad (8)$$

where k – number of solutions in the found Pareto-optimal set; $M_{ij}(x)$ – value of one of the functions $f(x)$, $g(x)$ or $h(x)$, found on the basis of its approximate model; $F_{ij}(x)$ – value of the same function found from the exact model, and ε – sufficiently small positive number. The fulfillment of this condition means that the quality of the constructed approximate models is such that they can be used instead of the exact ones;

– finding one or more vectors $f(x)$ satisfying predefined requirements $f(x) \leq f_{goal}$ subject to restrictions $g(x) > 0$ and $h(x) = 0$, where f_{goal} – values of target functions set by the expert are sufficient to ensure the required characteristics of the operated product;

– exceeding the permissible number of exact calculations of models;

– exceeding the permissible computation time.

5. Method for constructing an approximate model of helicopters aircraft GTE using a neural network

The key issue in the success of the proposed algorithm is the choice of an efficient way to build an approximate model or a response surface model (RSM). In particular, to construct approximate functional dependencies, the method of group accounting of arguments [17, 18], multilayer perceptrons and other models are used. In this paper, we consider the use of multicriteria optimization problems for modeling of the type "minimize $f(x)$ provided $g(x) > 0$, $h(x) = 0$" using radial basis neural network (RBF-network) obtained using evolutionary algorithms [19].

Several such methods are known, in particular, a rather general one described in [20]. The disadvantage of this method is redundancy in the description of the network (separate matrices are introduced to describe weights, connections and a vector to describe neurons). A simplified version of this way of describing a network is considered in [21]. According to this method, the object of evolution is the population of neural networks. In addition, the works [22] are known, which are limited to considering only RBF-networks, which allows us to proceed to the consideration of the evolution of a population of neurons, which are then combined into a network. However, the last algorithm is applicable only for generating networks for the classification of images, since it assumes knowledge of the centers of classes of objects under study.

The main feature of the method of evolutionary algorithms [23], which distinguishes it from the analogous method of genetic algorithms, is the refusal to use the crossover operation. In [24], based on the analysis of many sources, it was concluded that for the problem of generating neural networks, evolutionary algorithms are a more efficient method, since the crossover operation often leads to a deterioration in the fitness of descendants.

To solve this problem, a neural network was created with activation functions of a radial basis [25] (fig. 4) with an admissible root mean square error $E(\omega) = 0.3$ and an influence parameter equal to 1, the value of which is set the greater, the greater the range of input values must be taken into account.

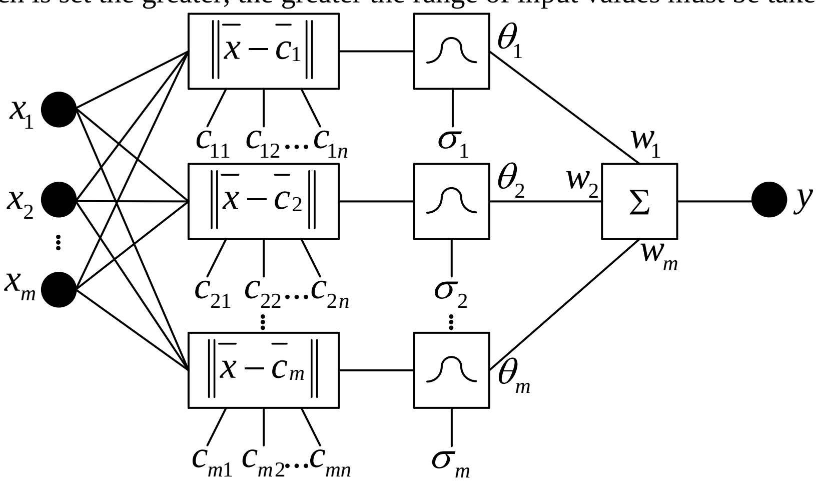

Figure 4: RBF-network general view

For RBF-network training, a gradient algorithm based on minimizing the objective function of the network error is used. In accordance with this algorithm, for each element, the values of changes in the weight coefficient, element width and element center coordinates are calculated.

As a result of the experiments, some shortcomings of the classical gradient RBF-network training algorithm were revealed:

1. In the RBF training algorithm, there are no rules for the initial setting of the number of network elements and their parameters, and there are also no rules for changing the number of elements in the training process. Distributing items evenly in the work area is not always optimal. Also, a situation may arise when the number of elements specified initially is insufficient to achieve the required quality of training.

2. During the training process, the parameters of all network elements are changed. As a result, as the number of elements increases, the computational cost of training also increases.

3. RBF-network cannot reach a steady state in the training process in cases when there are elements with close values of the coordinates of the centers and the width of the radial function of the network elements. The appearance of such situations largely depends on the selected number of elements and their initial parameters. The reason for the deterioration in the quality of training is that the gradient algorithm assumes that the output RBF-network value at each point in the work area is mainly affected by only one element. If there are several elements in one section of the working area, changing their parameters in accordance with the gradient algorithm does not always lead to a decrease in the training error.

In order to eliminate the shortcomings of the classical gradient RBF-network training algorithm, an evolutionary algorithm for constructing an RBF-network is proposed.

Time was taken as input elements, and the levels of the time series y' were taken as outputs. Functions of the NNT package used to create an RBF-network: net = newrb (t; y'; 0.3; 1) – creating a radial basic neural network with training; yn' = sim (net, t) – network simulation.

The activation function of the neuron of the hidden layer has the form:

$$y_i = \varphi\left(\|x - c_i\|\right) = e^{-\frac{\|x - c_i\|}{2 \cdot \sigma_i^2}}; \tag{9}$$

where $\|x - c_i\| = \sqrt{\sum_{j=1}^{N}\left(x_j - c_{ij}\right)^2}$ – Euclidean distance between input signals vector $x = (x_1 \ldots x_n)$ and the center of the i-th neuron $c_i = (c_{i1} \ldots c_{iN})$, $i = 1 \ldots L$; L – number of neurons in the hidden layer; N – number of neurons in the input layer; c_i, σ_i – parameters of the radial basis function of the i-th neuron. The signal of the neuron of the output layer is determined by the weighted summation of the outputs of the neurons of the hidden layer $f_k = \sum_{i=1}^{L} w_i \cdot y_i$, where w_i – connection weight from the i-th neuron of the hidden layer to the neuron of the output layer. Let us introduce the notation: $\mathbf{z} = (z_1 \ldots z_p)^T$ – vector of expected values of the function (p – number of training samples), $\mathbf{w} = (w_1 \ldots w_L)^T$ – weights vector, $\mathbf{G}$ – radial matrix, which has the form

$$\mathbf{G} = \begin{pmatrix} \varphi\|x_1 - c_1\| & \varphi\|x_1 - c_2\| & \ldots & \varphi\|x_1 - c_L\| \\ \varphi\|x_2 - c_1\| & \varphi\|x_2 - c_2\| & \ldots & \varphi\|x_2 - c_L\| \\ \ldots & \ldots & \ldots & \ldots \\ \varphi\|x_p - c_1\| & \varphi\|x_p - c_2\| & \ldots & \varphi\|x_p - c_L\| \end{pmatrix}; \tag{10}$$

Then the vector of weights can be found by the formula:

$$\mathbf{w} = \mathbf{G}^{+} \cdot \mathbf{z}; \tag{11}$$

where $\mathbf{G}^{+} = (\mathbf{G}^{\mathbf{T}}\mathbf{G}^{\mathbf{-1}})\mathbf{G}^{\mathbf{T}}$ – pseudo-inversion of a rectangular matrix $\mathbf{G}$.

Thus, the i-th neuron of the hidden layer can be completely described by a string of (N + 2) real numbers, which contains the vector $c_i = (c_{i1} \ldots c_{iN})$, the value of σ_i and the value of w_i. Therefore, to describe the entire network, an $L \times (N + 2)$ matrix R is required. However, since this method [6] uses a self-adaptive way of adjusting weights, it is necessary to add a matrix η of the same size to the description of a neuron, containing variations (strategic parameters of the evolutionary algorithm).

As a result of modeling the RBF-network for real data, an approximating function was obtained (fig. 5, *a*). In fig. 5, *b* shows a graph of the corresponding error (deviations of the actual data from the calculated ones).

As can be seen from fig. 5, *b*, the RBF-network successfully restores the dependence of π_C^* and T_G, while the approximation error does not exceed 0.021 %.

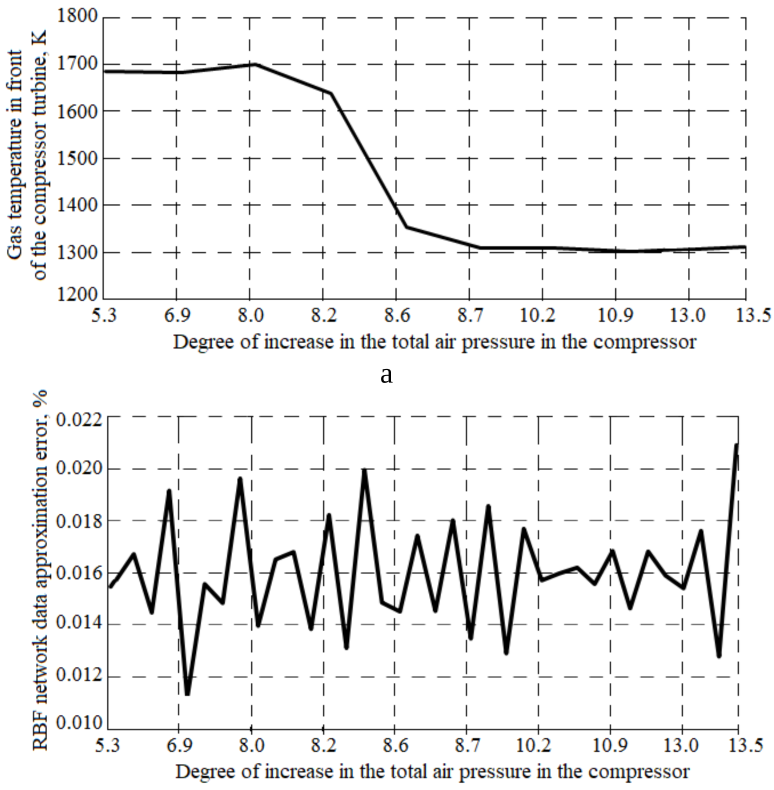

Figure 5: Approximation of data using RBF-network: *a* – Diagram of dependence of π_C^* on T_G; *b* – RBF data approximation error

According to fig. 5, and the Kendall correlation coefficient between the parameters π_C^* and T_G is r_{xy} = 0.946, which indicates a strong correlation between the parameters π_C^* and T_G, while the approximation error is 1.635 % (does not exceed the boundary permissible 10 %). Therefore, the model regressors π_C^* and T_G were chosen correctly.

Thus, according to fig. 5, *a*, input parameters are the degree of increase in the total air pressure in the compressor π_C^* and the temperature of the gases in front of the compressor turbine T_G. A fragment of the training sample is given in table 1.

Table 1

Fragment of the training sample

No	π_C^*	T_G, K
1	5.3	1690
2	6.9	1685
3	8.0	1700
4	8.2	1640
…	…	…
40	13.5	131

It is worth noting that for the implementation of the normal distribution law, 40 experimental points were used in the work according to fig. 5, *a*. Table 1 shows only a fragment of the input data. If necessary, the number of experimental points can be increased.

Evolutionary algorithm for constructing a neural network of a radial basis is shown in fig. 6.

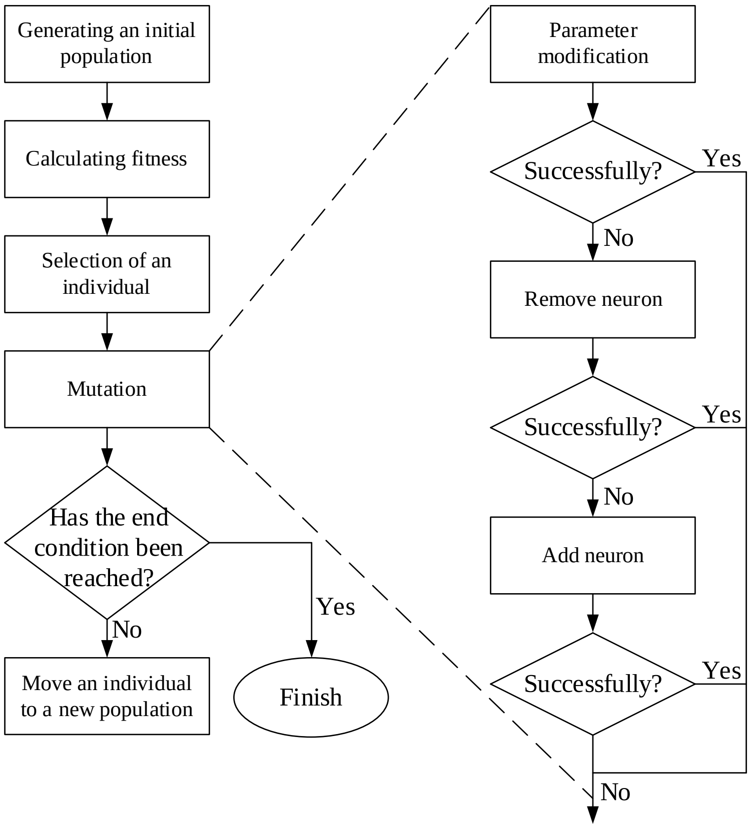

Figure 6: Evolutionary algorithm for constructing a neural network of a radial basis [6]

In the initial population, all parameters in the network description are initialized with random values from the interval (–1, 0; 1, 0). The fitness of all individuals of the population is calculated using the expression:

$$e_m = \frac{1}{T} \cdot \sqrt{\sum_{i=1}^{T} \left(Y_m(t) - Z_m(t)\right)^2}; \tag{12}$$

where T – number of samples in the training sample, $Y(t)$ and $Z(t)$ are the expected and actual values at the output of the network. The selection mechanism for an individual is based on its rank. Let K individuals be sorted in descending order of function (12) and assigned numbers 0, 1… K – 1. Then an individual with number (K – j) can be selected for mutation with probability $p(K-j) = j \cdot \left(\sum_{k-1}^{K} k\right)^{-1}$.

Before the start of the mutation, an integer n is randomly selected from the interval (1, L), which determines the number of the neuron to which the mutation operation will be applied. The following mutation operations are sequentially applied to this neuron.

1. *Modification of the activation function parameters.* A Gaussian mutation is used, according to which the new values of the matrix **R** row for a given neuron are calculated according to the expressions:

$$\eta'_{nj} = \eta_{nj} \cdot e^{\tau' \cdot N(0,1) + \tau \cdot N_j(0,1)}; \tag{13}$$

$$R'_{nj} = R_{nj} + \eta'_{nj} \cdot N_j(0,1); \tag{14}$$

where $N(0, 1)$ – random number that obeys a normal distribution with an average value of 0 and a variation of 1; $N_j(0, 1)$ means that a random number is generated for each j-th element of the vector; $\tau = \frac{1}{\sqrt{2 \cdot \sqrt{N}}}$; $\tau' = \frac{1}{\sqrt{2 \cdot N}}$.

After modifying the parameters of the n-th neuron, the weights are refined according to expression (11) and the fitness of the resulting network is calculated. If it improves, the resulting offspring is placed in a new population, no other mutations are made. Otherwise, the old values are returned to the rows R_n and η_n and an attempt is made to perform the next mutation.

2. *Remove of a neuron.* This operation is performed in case of failure of the previous mutation. The selected neuron is removed, according to expression (11), the weight coefficients are calculated, the fitness of the network is estimated; if it improves, then the resulting offspring is copied into the new population. Otherwise, the neuron addition mutation is applied.

3. *Adding a neuron.* All parameters of the added neuron are initialized with random values from the interval (–1; 1), according to expression (11), weight coefficients are calculated. If the fitness of the network improves, the resulting descendant is copied into the new population.

If none of the mutations were successful, then the individual is copied into the population of the next generation without changes. Note that this method uses the so-called "greedy" algorithm – an attempt to remove a neuron is always made before an attempt is made to add it. This provides more compact networks. In addition, the principle of elitism is used – the best individual of the current population is copied into a new one without changes.

6. Evaluation of the effectiveness of the method for constructing approximate models of helicopters aircraft GTE using a neural network

To evaluate the efficiency of the proposed algorithm, consider the problem of approximation of the function [26]:

$$f\left(x_1, x_2\right) = \frac{x_1^{\frac{k-1}{k}} - 1}{x_1^{\frac{k-1}{k} \cdot \frac{1}{x_2}} - 1}; \tag{15}$$

when variables change within $0 \le x_1 \le 20$ and $0 \le x_2 \le 20$, where $k = 1.4$ (has the physical meaning of the adiabatic exponent).

This expression was taken as a test example from the considerations that it is an analytical expression for calculating the efficiency of the compressor of helicopters TE – one of the most important indicators of engines operational status.

Based on a training sample of 625 data groups ($[x_1, x_2]$, d) generated with a uniform distribution of variables x_1 and y in their domains, in [27], a network with a structure of 2–36–1 (fig. 7) was constructed (2 input neurons, 36 radial neurons of the Gaussian type, and one output linear neuron). A hybrid training algorithm was used; as a result, the maximum approximation error after 200 iterations was 0.06. According to this method [6], based on the same training sample for 20 generations, a neural network with 26 radial neurons was generated, the approximation error of which has a value of 0.02.

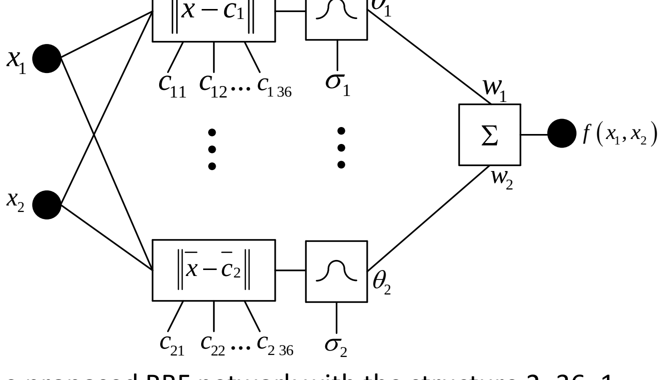

Figure 7: Diagram of the proposed RBF network with the structure 2–36–1

The diagrams of the function being approximated is shown in fig. 8, *a*, the error of its approximation based on this method is shown in fig. 8, *b*. Thus, the proposed method for generating RBF-networks can significantly reduce the computation time and provide more efficient networks (with fewer neurons and less error) compared to the traditional method.

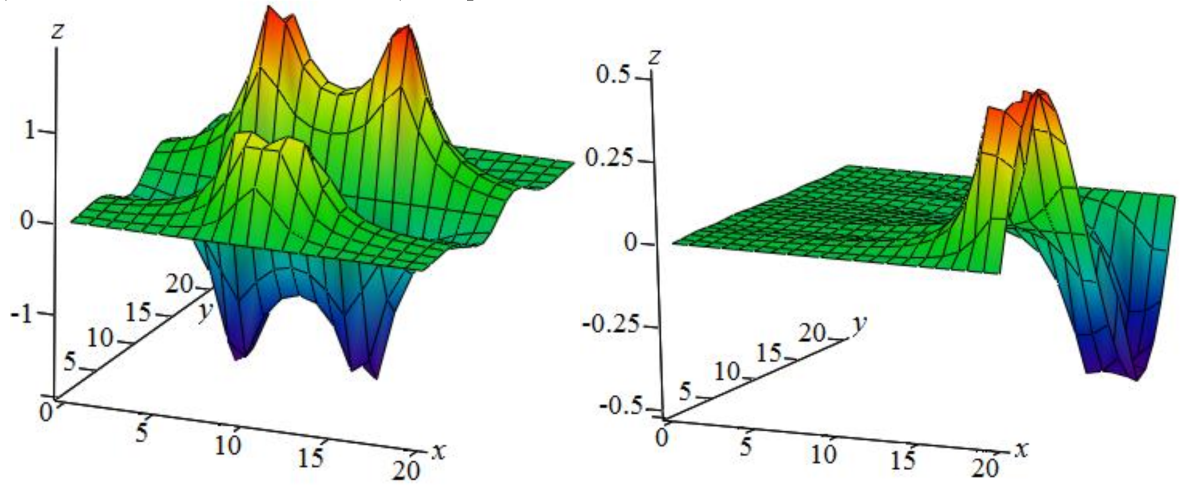

Figure 8: 3D surfaces of the approximating function: *a* – test function graph (13); *b* – error of approximation of the test function

Approximate models of the function (15) were also built on the basis of widely known and used in practice methods, such as multilayer perceptron, cascade correlation network, and the method of group consideration of arguments. The total sample of 625 records was randomly divided into training (90 % of records) and test (10 % of records). The model was built on a training set, then its quality was checked on a test set. The obtained values of the root-mean-square error (12) are given in table 2. This method [6] of constructing approximate models showed the best results.

Table 2

Comparison of various methods for constructing approximate models

Model	Description of the built model	Root mean square error	
		Training sample	Test sample
Multilayer perceptron	Two hidden layers (11 and 4 neurons) perceptron of neurons with a logistic activation function, an output layer neuron with a linear activation function	2.645	2.960
Argument grouping method	Polynomials of the second and third degree and Gaussians	1.016	1.070
Cascade correlation network	16 neurons in a hidden layer with Gaussian activation function	0.284	0.492
Approximate model construction method	26 neurons with radial activation function in the hidden layer:		
	developed RBF-network training algorithm in this paper	0.021	0.064
	developed in [26] RBF-network training algorithm	0.177	0.232
	standard RBF-network training algorithm	0.215	0.365

The quality of the approximation by various methods was estimated by the coefficient of determination (table 3), which characterizes the so-called fraction of the "explained" variance and is defined as

$$R^2 = 1 - \frac{\sum_i \left(y_i - y_i\right)^2}{\sum_i \left(y_i - M\left(y\right)\right)^2}; \tag{16}$$

where y_i – initial values, $\hat{y}_i$ – aligned values, $M(y)$ – sample mean of the original series of dynamics. The closer this coefficient is to unity, the better the quality of approximation.

Table 3

Results of evaluating the quality of approximation

Model	Determination coefficient R^2
Multilayer perceptron	0.632
Argument grouping method	0.788
Cascade correlation network	0.834
Proposed model:	
developed RBF network training algorithm in this paper	0.995
developed in [26] RBF network training algorithm	0.918
standard RBF network training algorithm	0.883

The unsatisfactory quality of the approximation using a multilayer perceptron is explained by its simplicity and specificity of the initial data, which are the rotational speed of the turbocharger rotor and the temperature of the gases in front of the compressor turbine.

Based on the data presented, it can be concluded that the use of neural networks gives an acceptable (sufficiently high) level of approximation of the initial observed data, primarily due to the presence in the RBF-network of a hidden layer of neurons with nonlinear radial-basis activation functions that allow you to track the slightest changes in the levels of the investigated time series. When using RBF-network with developed RBF-network training algorithm, we got $R^2 = 0.995$, this is the case when the real output of the neural network and the desired output (which in meaning coincides with the estimated and real values) practically coincide. Using traditional methods, it is almost impossible to achieve such a high value of the coefficient of determination.

7. Helicopters aircraft GTE mathematical model (for example TV3-117)

To test the effectiveness of the proposed optimization method, we will conduct research on a linear mathematical model of the TV3-117 TE [28–30], which requires much less computation time than the model presented in [6], but at the same time has all the features actually used in practice functions $f(x) \rightarrow \min$ for $g(x) > 0$, $h(x) = 0$. To create such a model, we use the results of [31], where experimental dependences are given that relate some parameters of the working process (for example, the dependence of the compressor efficiency on its rotational speed), which reduces the number of independent variables. According to this approach, the calculation of the parameters of the working process of TV3-117 TE must be carried out in several sections (at the engine input, at the compressor input, behind the compressor, behind the combustion chamber, behind the compressor turbine, behind the free turbine), shown in fig. 9 and designated respectively by the indices *N*, *In*, *C*, *CC*, C*T*, *FT*, where *N* – environment, *In* – air inlet section, *C* – compressor, *CC* – combustion chamber, C*T* – compressor turbine, *FT* – free turbine.

The parameters at the engine inlet are determined by the speed and altitude. At the first stage of the calculation, the gas pressure and temperature are determined sequentially for each section. In this case, the degree of pressure increases in the compressor π_C^* and the gas temperature in front of the turbine of the T_G compressor must be set. At this stage of the calculation, the operation of the L_C compressor and the L_T compressor turbine, the air consumption and the specific fuel consumption C_{sp}, which is necessary to create a given engine power, are determined.

Further, on the basis of the previously determined values of L_C, L_T and the maximum possible value of the operation of one stage, the number of stages of the compressor z_C and the turbine z_T, as well as the rotor speed n_{TC}, is determined. The data on the r.p.m. and geometry of the flow section make it possible to determine the tensile stresses σ_p in the blade of the impeller of the last stage of the turbine, which should not exceed 250 MPa [32].

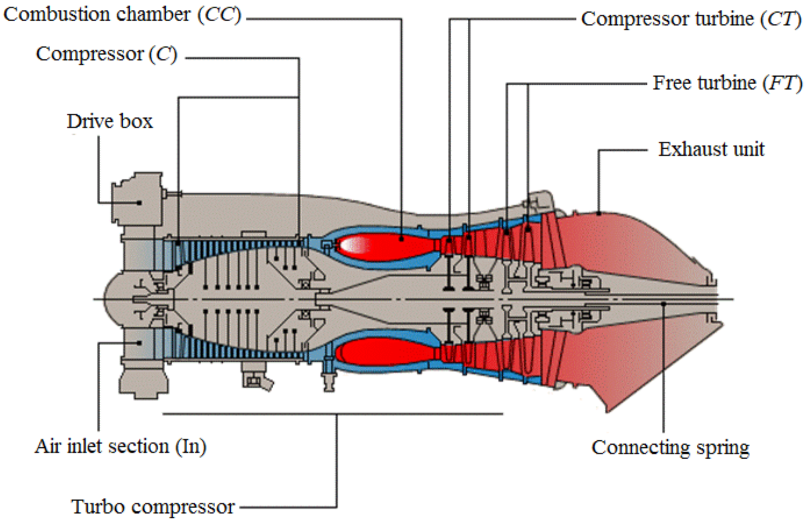

Figure 8: TV3-117 TE structural model

In accordance with the considered mathematical model, the working process of TV3-117 TE is completely determined by seven independent parameters: π_C^* – degree of pressure increase in the compressor, T_G – temperature of the gases in front of the compressor turbine, λ_B, λ_C, λ_{CC}, λ_{CT}, λ_{FT} – reduced gas flow rates behind the inlet, compressor, combustion chamber, compressor turbine and free turbine, respectively. The restrictions on the choice of the admissible combination of independent parameters are h_z – the height of the blade of the last stage of the compressor and σ_p – tensile stress in the blade of the impeller of the last stage of the turbine. Thus, this model allows you to vary the values of π_C^* and T_G to obtain optimal parameters of the working process. The dependences C_{sp} (kg/N·h), and σ_p (kg/mm^2) on π_C^* and T_G (K) are shown in fig. 10, *a* and *b*, respectively.

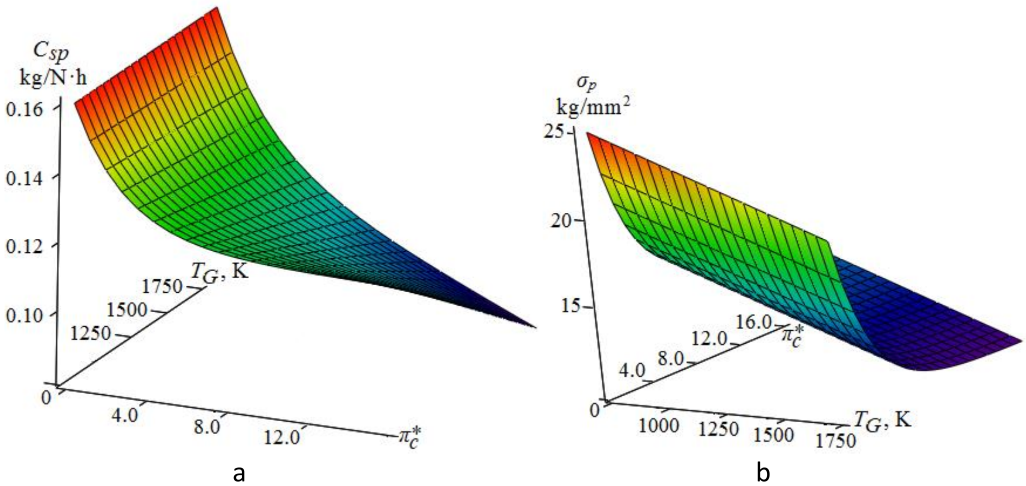

Figure 9: Graph of the effect of independent variables on objective functions and constraints

8. Results and discussion

Consider the problem of finding the Pareto-optimal set of the working process of TV3-117 TE. As the target variables that need to be minimized, let us determine the specific fuel consumption, for example, at takeoff mode. Let us set the intervals of variation for the independent variables [6]: the compression ratio in the compressor π_C^* = 4…20, gas temperature T_G = 1300…1800 K, reduced gas flow rates λ_{in} = 0.6…0.7, λ_C = 0.25…0.35, λ_{CC} = 0.15…0.25, λ_{CT} = 0.4…0.65, λ_{FT} = 0.5…0.7 and limitations: the height of the blade of the last stage of the compressor h_z > 15 mm and the tensile stress in the blade of the last stage of the turbine $\sigma_p > 25$ kg/mm^2. The computational model of the engine is built in the way described above. Let us set $\varepsilon = 0.005$ in the condition of the end of calculations (8).

The process of solving the formulated problem in accordance with the algorithm shown in fig. 3 is presented in table 3. At the first step, a training set of 50 decision vectors $\mathbf{x} = (\pi_C^*, T_G, \lambda_B, \lambda_C, \lambda_{CC}, \lambda_{CT}, \lambda_{FT})$ was generated in accordance with the central composite design of the experiment with centers on the faces (CCF – Central Composite design with Face centered). Of these 50 solutions, 10 met the constraints and 7 were non-dominant. Based on this sample, approximate models were built for target variables and constraints based on neural networks of a radial basis in accordance with the method described above. Table 4 for each model shows the number of neurons in the hidden layer N_h and fitness, calculated by the expression (12).

Table 4

The process of finding a Pareto-optimal solution set

		First iteration	Second iteration	Third iteration	Result
The number of solutions in the training sample		50	140	230	320
The number of solutions that satisfy the constraints		10	45	110	200
The size of the Pareto optimal set		8	15	24	40
Model $C_{y\partial}$	N_h	37	39	35	–
	e_m	0.00009	0.00011	0.00007	–
Model σ_p	N_h	37	40	38	–
	e_m	0.02542	0.01768	0.34325	–
Model h_z	N_h	40	35	36	–
	e_m	0.00005	0.00004	0.00004	–
The total relative error of the models	e	0.0065	0.0054	0.0037	–

Based on the models obtained, using the NSGA-II algorithm (population size – 100 individuals, 500 training generations), a set of 100 Pareto-optimal solutions was found, the total relative error (8) was $e = 0.0065$. After checking these decisions on the exact model, they were added to the training set, the size of which was now 140 vectors (of which 45 met the constraints, 15 belonged to the Pareto-optimal set), and the whole computation cycle was repeated anew (second iteration). In total, three iterations were performed, which required 320 calls to the minimization functions. The total relative error of the models built at the second iteration was $e = 0.0054$, and at the third iteration – $e = 0.0037$. Some of the found Pareto-optimal parameters of the working process of the aircraft TV3-117 TE are presented in table 5.

Table 5

Variants of the parameters of TV3-117 aircraft GTE working process

$C_{y\partial}$ kg/N·h	σ_p kg/mm^2	h_z mm	π_K	T_Γ K	λ_B	λ_K	λ_Γ	λ_T	λ_{CT}
0.085	12.9	15	13.0	1310	0.685	0.250	0.25	0.640	0.682
0.089	17.4	15	10.9	1302	0.693	0.278	0.25	0.462	0.577
0.092	17.7	15	10.2	1308	0.693	0.299	0.25	0.468	0.581
0.096	17.5	15	8.7	1308	0.693	0.343	0.25	0.507	0.586
0.101	16.9	15	8.6	1354	0.693	0.343	0.25	0.539	0.593
0.109	17.0	15	8.7	1466	0.693	0.343	0.25	0.525	0.599
0.115	17.1	15	8.6	1545	0.693	0.343	0.25	0.508	0.575
0.124	17.7	15	8.2	1638	0.693	0.343	0.25	0.485	0.564
0.130	17.2	16	8.0	1699	0.693	0.343	0.25	0.505	0.542
0.137	21.5	19	6.9	1684	0.693	0.343	0.25	0.443	0.529
0.146	24.2	23	5.3	1686	0.693	0.343	0.25	0.469	0.511

The results of all iterations are shown in fig. 11 (the number of points in the training set – the number of points belonging to the Pareto optimal set of solutions is indicated in brackets). In fig. 11, *a* also shows the Pareto-optimal set (Pareto front) obtained by the NSGA-II method (100 individuals in the population, 500 generations) based on the exact model. Finding this set required 50000 calls to the minimization functions. In fig. 11, *b* shows a comparison of three Pareto-optimal sets of solutions: obtained on the basis of the approximate model proposed here (320 calls to the exact model) and obtained on the basis of the exact model for 500 calls (100 individuals in the population, 5 generations) and for 50000 calls (100 individuals in population, 500 generations).

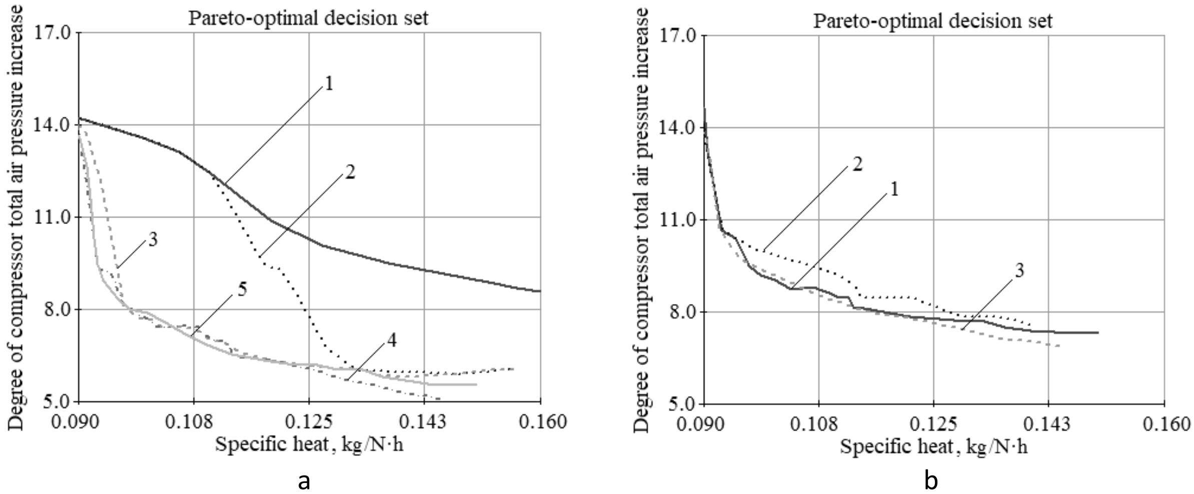

Figure 11: Results: *a* – Evolution of Pareto-optimal decision sets in the computation process: 1) starting training sample (50/8), 2) 1st iteration (140/15), 3) 2nd iteration (230/24), 4) 3rd iteration (320/40), 5) solution based on exact model (50000/100); *b* – Comparison of three Pareto-optimal decision sets: 1) an approximate model, 2) an exact model – 5 generations, 3) an exact model – 500 generations

The results obtained indicate that the proposed method for constructing approximate models allows reducing the amount of computer time spent on calculations in case of multicriteria optimization with constraints by more than 100 times.

The program is protected because the described method is implemented, written in Python 2.6 for the modern library and script. The program is packaged with clear modules (fig. 12) in order to implement experiment planning, rich-criteria optimization using the NSGA-II method, model approximation on the basis of the RBF-network, as it was for analysis more, and a graphical interface of the core.

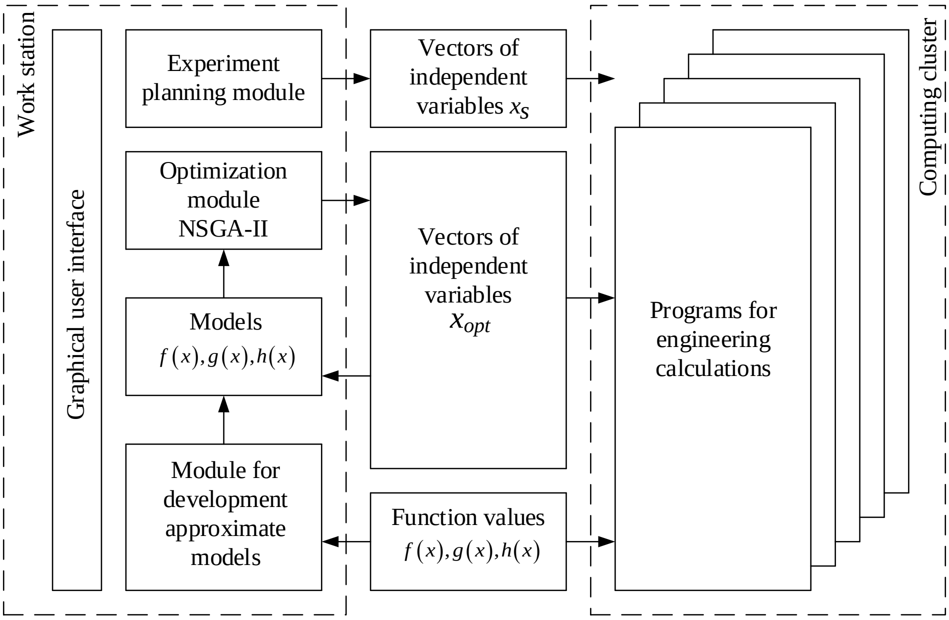

Figure 12: Diagram of software implementation of the proposed method

The high speed of robots of the proposed methods, as well as fast calculation for a sufficient biblical system and the use of Python, allows you to deploy all system components on a single workstation without loss of performance. The exchange of current software errors, in which the exact models are counted, is carried out through the exchange files. These systems can be installed in a parallel environment on a numerical cluster.

9. Conclusions

The method of constructing an approximate model of the investigated object, based on the algorithm of multicriteria optimization, using RBF-networks, was further developed, which, due to the use of a simplified mathematical model of helicopters turboshaft engines, as well as gas-dynamic functions, allows to optimize helicopters turboshaft engines working process parameters during a helicopter flight.

The method used in this work for constructing an approximate model of the object under study allows us to obtain the range of rational values of the parameters for two-parameter problems by the example, helicopters turboshaft engines working process parameters – the degree of air pressure increase in the compressor and the temperature of gases in front of the compressor turbine.

Variants of the optimal working process of helicopters turboshaft engines at take-off mode have been obtained, which make it possible to apply the optimal control program to obtain maximum engine power.

The results obtained indicate that the proposed method for constructing approximate models allows one to reduce the computer time spent on calculations for multi-criteria optimization with constraints by more than 100 times. The results of this work can be introduced into an intelligent on-board system for control and diagnostics of aircraft GTEs operational status, including helicopters turboshaft engines [33].

The scientific novelty of obtained results is as follows:

1. The method of multicriteria optimization based on approximate models of complex dynamic objects was further developed, which, through the use of a radial-basic neural network and an evolutionary algorithm for its training, made it possible to optimize the thermogasdynamic parameters of helicopter engines working process at flight modes, which will allow the helicopter crew adjust the engine control program and, thereby, increase the safety of the helicopter flight.

The evolutionary method of training a radial-basic neural network has been improved, which, due to the modification of the parameters of the neural network, including the removal and addition of neurons, has reduced the maximum mean square training error to 0.064 on the test selection and to 0.021 on the training sample, thereby ensuring maximum accuracy data approximation in solving the problem of multicriteria optimization of complex dynamic objects.

10.References

[1] Tkachenko A. Y., Kuz'michev V. S., Krupenich I. N. and Rybakov V. N. (2016), "Gas turbine engine optimization at conceptual designing", MATEC Web of Conferences, vol. 77, URL: https://www.matec-conferences.org/articles/matecconf/pdf/2016/40/matecconf_icmmr2016_01027.pdf

[2] Mankowski M., Moshkov M. Dynamic Programming Multi-Objective Combinatorial Optimization. Studies in Systems, Decision and Control. Berlin, Springer Nature. 2019. 230 p.

[3] Ploch T., Deussen J., Naumann U., Mitsos A., Hannemann-Tamas R. Direct single shooting for dynamic optimization of differential-algebraic equation systems with optimization criteria embedded, *Computers & Chemical Engineering*, 2022, vol. 159, pp. 107643.

[4] Dong W. Y., Zhang R. R. Stochastic stability analysis of composite dynamic system for particle swarm optimization, *Information Sciences*, 2022, vol. 592, pp. 227–243.

[5] Lopes H. N., Cunha D. C., Pavanello R., Mahfoud J. Numerical and experimental investigation on topology optimization of an elongated dynamic system, *Mechanical Systems and Signal Processing*, 2022, vol. 165, pp. 108356.

[6] Zelenkov Y. A., Method of multi-criterial optimization based on approximate models of the researched object, *Numerical Methods and Programming*, 2010, vol. 11, pp. 250–260.

[7] Ji J.-Y., Wong M. L. An improved dynamic multi-objective optimization approach for nonlinear equation systems, *Information Sciences*, 2021, vol. 576, pp. 204–227.

[8] Miller S. J. The Method of Least Squares, *The Probability Lifesaver*, 2017, pp. 625–635.

[9] Grigoriev V. A., Rad'ko V. M., Kalabuhov D. S. Approximation models of criteria for evaluating small gas turbine efficiency for multipurpose helicopter, *Aerospace Engineering and Technology*, 2011, no 9 (86), pp. 19–24.

[10] Seyyedrahmani F., Shahabad P. K., Serhat G., Bediz B., Basdogan I. Multi-objective optimization of composite sandwich panels using lamination parameters and spectral Chebyshev method, *Composite Structures*, 2022, vol. 289, pp. 115417.

[11] Ntantis E. L., Li Y. G., The impact of measurement noise in GPA diagnostics analysis of a gas turbine engine, *International Journal of Turbo & Jet Engine*, 2013, vol. 30 (4), pp. 401–408.

[12] Merrikh-Bayat F., Afshar M. Formulation of nonlinear control problems with actuator saturation as linear programs, *European Journal of Control*, 2021, vol. 61, pp. 133–141.

[13] Liu Y., Hu Y., Zhu N., Li K., Zou J., Li M. A decomposition-based multiobjective evolutionary algorithm with weights updated adaptively, *Information Sciences*, 2021, vol. 572, pp. 343–377.

[14] Patil M. V., Kulkarni A. J. Pareto dominance based multiobjective cohort intelligence algorithm, *Information Sciences*, 2020, vol. 538, pp. 69–118.

[15] Clarich A., Mosetti G., Pediroda V., Poloni C. Application of evolutionary algorithms and statistical analysis in the numerical optimization of an axial compressor, *International Journal of Rotating Machinery*, 2005, no. 2005:2, pp. 143–151.

[16] Ji Y., Yang Z., Ran J., Li H. Multi-objective parameter optimization of turbine impeller based on RBF neural network and NSGA-II genetic algorithm, *Energy Reports*, 2021, vol. 7, pp. 584–593.

[17] Chen R. A Multi-Objective Robust Preference Genetic Algorithm Based on Decision Variable Perturbation, *Advanced Materials Research*, 2011, vol. 211–212, pp. 818–822.

[18] Stepashko V. S. Formation and development of self-organizing intelligent technologies of inductive modeling, *Cybernetics and Computer Engineering Journal*, 2018, issue 4 (194), pp. 2578–2663.

[19] Ayala H. V. H., Habineza D., Rakotondrabe M., Coelhod L. Nonlinear black-box system identification through coevolutionary algorithms and radial basis function artificial neural networks, *Applied Soft Computing*, 2020, vol. 87, pp. 105990.

[20] Banzhaf W. Evolutionary computation and genetic programming, *Engineered Biomimicry*, 2013, pp. 429–447.

[21] Herzog S., Tetzlaff C., Worgotter F. Evolving artificial neural networks with feedback, *Neural Networks*, 2020, vol. 123, pp. 153–162.

[22] Hang J., Li Y., Xiao W., Zhang Z. Non-iterative and fast deep learning: multilayer extreme learning machines, *Journal of the Franklin Institute*, 2020, vol. 357, issue 13, pp. 8925–8955.

[23] Soltoggio A., Stanley K. O., Risi S. Born to learn: The inspiration, progress, and future of evolved plastic artificial neural networks, *Neural Networks*, 2018, vol. 108, pp. 48–67.

[24] Zhang L., Li H., Kong X.-G. Evolving feedforward artificial neural networks using a two-stage approach, *Neurocomputing*, 2019, vol. 360, pp. 25–36.

[25] Junfei Q., Xi M., Wenjing L. An incremental neuronal-activity-based RBF neural network for nonlinear system modeling, *Neurocomputing*, 2018, vol. 302, pp. 1–11.

[26] Vladov S., Dieriabina I., Husarova O., Pylypenko L., Ponomarenko A. 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*, 2021, no. 4 (79), pp. 52–63.

[27] Alanis A. Y., Arana-Daniel N., Lopez-Franco C. Artificial Neural Networks for Engineering Applications. London, Academic Press. 2019. 176 p.

[28] Mukhamedov R. R. GTE mathematical models, *Youth Bulletin of USATU*, 2014, no 1 (10), pp. 35–43.

[29] Vladov S. I., Podgornykh N. V., Teleshun V. Ya. Mathematical model of TV3-117 aircraft engine compressor for its control and diagnostics of a technical state in the conditions of onboard operation of the aircraft, *The path to success and prospects for development (to the 26th anniversary of the Kharkiv National University of Internal Affairs)* : proceedings of the international scientific-practical conference, November 20, 2020, Kharkiv. Pp. 112–116.

[30] Vladov S. I., Yankevich N. S. Linear mathematical model of the aircraft engine TV3-117, *Aviation and cosmonautics* : proceedings of the XII All-Ukrainian scientific-practical conference, April 20, 2021, Kryvyi Rih. P. 57.
[31] Pashayev A. M., Askerov D. D., Ardil C., Sadiqov R. A., Abdullayev P. S. Condition monitoring system of aircraft gas turbine engine complex, *International Journal of Aerospace and Mechanical Engineering*, 2007, vol. 1, no 11, pp. 689–695.
[32] Mu J., Rees D., Liu G. P. Advanced controller design for aircraft gas turbine engines, *Control Engineering Practice*, 2005, vol. 13, issue 8, pp. 1001–1015.
[33] Shmelov Y., Vladov S., Klimova Y., Kirukhina M., 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, pp. 77–82.