Proceedings of 2017 4th International Conference on
Control, Decision and Information Technologies (CoDIT'17) / April 5-7, 2017, Barcelona, Spain

Automatic Search of Reliability Function by Symbolic Regression

Diveev A.I.
Federal Research Centre "Computer Science and Control" of Russian Academy of Sciences, RUDN University,
Moscow, Russia
aidiveev@mail.ru

Shmalko E.Yu.
Federal Research Centre "Computer Science and Control" of Russian Academy of Sciences
Moscow, Russia
eshmalko@googlemail.com

Sofronova E.A.
Peoples Friendship University of Russia (RUDN University)
Moscow, Russia
Sofronova_ea@mail.ru

Zhadnov V.V.
Moscow Institute of Electronic and Mathematics of the National Research University "Higher School of Economics",
Moscow, Russia
vzhadnov@hse.ru

***Abstract*— A reliability index of various electronics is determined by the experimental data of tests for different values of parameters of the equipment. The received data are collected in bulky tables and references. This paper presents modern numerical approach, allowing to compile the experimental data on changes of reliability index not in the form of tables but as a function of the operating parameters of the devices. The methodology is based on the method of network operator for the design of the optimal structure of function and selection of its parameters. The network operator method belongs to a class of methods of symbolic regression and provides an evolutionary search for the best compositions of mathematical expressions on the space of elementary structures. The method allows you to automatically receive the required description of the functional dependencies. The effectiveness of the method is demonstrated by the example of searching the law, which describes the change in the failure rate depending on three parameters that characterize its constructive and technological performance and operating conditions.**

Keywords*— *electronic engineering; genetic algorithm; reliability index; symbolic regression

I. Introduction

Development and operating complex electronic devices requires determining how the operating parameters influence the change of reliability indices. Pilot studies are held for many electronic devices to study how their reliability index depends on the operating parameters. All the received data is accumulated in standards and spreadsheets.

By now specialists in reliability have accumulated a huge amount of data, as well. This experience is stored in many standards and spreadsheets. In fact, all these data reflect the laws of variation of some parameters depending on the values of some other parameters. There is always a problem to describe such laws and functions that show how such parameters like, for example, reliability index are changing depending upon other parameters like environment, temperature, operating conditions and so on.

In the case of univariate dependency, engineers and researchers suggest the formula for a function on the basis of their experience, intuition or ability to foresee. It is well known, for example, that the failure rate of mechanisms that involve some forms of chemical reactions increases exponentially with increasing temperature and therefore it is commonly modeled by Arrhenius equation [1]-[3].

But when we need to describe a multivariable function like a failure rate which is dependent upon two or more parameters, a great problem arises in doing so especially in the case when such parameters are more descriptional than measurable like acceptance conditions.

In this respect engineers can be helped greatly by computers to define such dependencies.

At the end of the 20th century the symbolic regression [4] was invented. The methods of symbolic regression [5] - [7], allow constructing the needed laws or mathematical expressions, including parameters using computational evolutionary approach [8], [9]. According to these methods the mathematical expression is iteratively composed from elements that are chosen from the set of elementary functions. And the evolutionary algorithm constructs the set of different mathematical expressions and then applies evolutionary technique to search for the best one to satisfy the given quality criteria. In our present paper we have applied one of the methods of symbolic regression, namely the network operator method [10] to find the multidimensional function that describes the dependency of the reliability index from the operating parameters. We took the experimental data, loaded it to computer and the program of the network operator found the mathematical expression that describes the needed dependency. In the present research we received the dependency of the failure rate from three operating

This work was supported by the Russian Foundation for Basic Research (grants 16-29-04224, 17-08-01203), and by the Ministry of Education and Science of Russian Federation (the agreement 02.a03.0008).

parameters. The received function provides the average relative error of 9% with its maximum value no more than 25.4%.

II. Problem Statement

Consider the following problem. The experimental data are given as m ordered sets of n operating parameters

$$X_1 = (x_{1,1},\dots,x_{1,n_1}),\dots,X_m = (x_{m,1},\dots,x_{m,n_m}), \quad (1)$$

and the reliability index

$$Y = (y_1,\dots,y_K), \quad (2)$$

where

$$K = \prod_{i=1}^{m} n_i . \quad (3)$$

Certain selected vector of operating parameters corresponds to a certain value of the reliability index

$$[x_{1,i_1} \dots x_{m,i_m}]^T \to y_j . \quad (4)$$

where $x_{k,i_k} \in X_k$, $k = 1,\dots,m$,

$$j = i_1 + \sum_{p=2}^{m-1} (i_p - 1) \prod_{q=1}^{p-1} n_q . \quad (5)$$

The goal is to find a mathematical expression of the function

$$\tilde{y} = f(x_1,\dots,x_m), \quad (6)$$

which delivers minimum to the criteria

$$J_1 = \max_j \left\{ \frac{| y_j - f(x_{1,i_1} \dots x_{m,i_m}) |}{y_j} : j = 1,\dots,K \right\} \to \min, \quad (7)$$

$$J_2 = \sqrt{\frac{1}{K} \sum_{j=1}^{K} (y_j - f(x_{1,i_1} \dots x_{m,i_m}))^2} \to \min, \quad (8)$$

where j is defined from (5) for values of $i_1,\dots,i_m$, $i_1 = 1,\dots,n_1,\dots,i_m = 1,\dots,n_m$.

To solve the problem we use one of the methods of symbolic regression – the network operator method.

III. Network Operator Method

The network operator method [10] works with mathematical expressions coded in the form of oriented graph. Such graph uniquely defines the order of calculation of the mathematical expression.

Use the four following basic sets to code any mathematical expression as a graph:

- a set of variables

$$V = (x_1,\dots,x_N); \quad (9)$$

- a set of parameters

$$Q = (q_1,\dots,q_R); \quad (10)$$

- a set of functions with one argument or unary operations in the network operator terminology

$$F_1 = (f_{1,1}(z) = z, f_{1,2}(z),\dots, f_{1,W}(z)); \quad (11)$$

- a set of functions with two arguments or binary operations

$$F_2 = (f_{2,1}(z_1,z_2),\dots, f_{2,V}(z_1,z_2)) . \quad (12)$$

The set of unary operations must contain an identical operation $f_{1,1}(z) = z$.

Functions in the set of binary operations are to be commutative, $\forall f_{2,i}(z_1,z_2) \in F_2$

$$f_{2,i}(z_1,z_2) = f_{2,i}(z_2,z_1), 1 \le i \le V, \quad (13)$$

and associative , $\forall f_{2,i}(z_1,z_2) \in F_2$

$$f_{2,i}(f_{2,i}(a,b),c) = f_{2,i},(a, f_{2,i}(b,c)), \ 1 \le i \le V, \quad (14)$$

and possess unit elements, $\forall f_{2,i}(z_1,z_2) \in F_2$, $\exists e_i$,

$$f_{2,i}(e_i,a) = a, \ 1 \le i \le V . \quad (15)$$

In the mathematical expression described by the network operator the arguments are elements of the sets of variables (9) and parameters (10). Parameters are needed to widen the space of possible functions and enhance the accuracy of the resulting function. The parameters are tuned together with the structure of the function.

The variables and parameters are positioned in the source-nodes of the directed graph of the network operator, the binary operations – in all other nodes, and the unary operations are positioned on the arcs of the directed graph.

Consider an example how to describe a mathematical expression in the form of the network operator.

Let us choose the following basic sets

$$\mathrm{V} = (x_1),\ \mathrm{Q} = (a,b),$$

$$\mathrm{F}_1 = (f_{1,1}(z) = z, f_{1,2}(z) = -z, f_{1,3}(z) = e^z, f_{1,4}(z) = \sin(z)),$$

$$\mathrm{F}_2 = (f_{2,1}(z_1,z_2) = z_1 + z_2, f_{2,2}(z_1,z_2) = z_1 z_2).$$

Notice that addition and multiplication as binary operations are sufficient for the majority of practical applications. These operations are commutative, associative and have the unit element for addition is zero, $e_1 = 0$, and for multiplication is one, $e_2 = 1$.

The variety of possible mathematical expressions is mostly generated by the set of unary operations. In our example it contains four operations. For real practical problems this set contains more than twenty functions.

Let us describe the following mathematical expression in the form of network operator

$$y = a\sin(e^{-(x_1+b)}\sin(cx_2)). \quad (16)$$

Rewrite the mathematical expression using elements of the basic sets.

$$\begin{aligned} y = f_{2,2}(a, \sin(e^{-(x_1+b)}\sin(ax_1))) = \\ f_{2,2}(a, f_{1,4}(e^{-(x_1+b)}\sin(ax_1))) = \\ f_{2,2}(a, f_{1,4}(f_{2,2}(e^{-(x_1+b)}, \sin(ax_1)))) = \\ f_{2,2}(a, f_{1,4}(f_{2,2}(f_{1,3}(-(x_1+b)), f_{1,4}(ax_1)))) = \\ f_{2,2}(a, f_{1,4}(f_{2,2}(f_{1,3}(f_{1,2}(x_1+b)), f_{1,4}(f_{2,2}(a,x_1))))) = \\ f_{2,2}(a, f_{1,4}(f_{2,2}(f_{1,3}(f_{1,2}(f_{2,1}(x_1,b))), \\ f_{1,4}(f_{2,2}(a,x_1))))) \end{aligned} \quad (17)$$

Such notation (17) is called a program notation of the mathematical expression. The unary and binary operations can be written as subprograms. So the program notation means calling these subprograms in the right order to calculate the value of the mathematical expression.

To construct the graph of the network operator the program notation (17) should be converted to a graph notation.

The graph notation is such program notation that satisfies the following requirements:

a) the first function in the notation is a binary operation, $y = f_{2,i}(\ldots)$;

b) only unary operation or a unit element can be an argument of any binary operation, $f_{2,i}(f_{1,k}(\ldots), f_{1,l}(\ldots))$;

c) arguments of any unary operation can contain binary operations, variables or parameters, $f_{1,i}(f_{2,k}(\ldots))$, $f_{1,i}(\alpha)$, $\alpha \in \mathrm{V} \cup \mathrm{Q}$;

d) arguments of any binary operation cannot contain such unary operations which have the same variable or the same parameter as an argument, $f_{2,i}(f_{1,k}(\alpha), f_{1,l}(\alpha))$, $\alpha \in \mathrm{V} \cup \mathrm{Q}$.

Any program notation can be converted to a graph notation by adding a unary identical operation $f_{1,1}(z) = z$ or a binary operation with its unit element as a second argument $f_{2,i}(z_1, e_i) = z_1$. These are examples how to modify those parts of program notation which do not satisfy the conditions a) – d):

a) $y = f_{1,k}(\ldots) \to y = f_{2,i}(f_{1,k}(\ldots), e_i)$;

b) $f_{2,i}(\alpha,\beta) \to f_{2,i}(f_{1,1}(\alpha), f_{1,1}(\beta))$, $\alpha,\beta \in \mathrm{V} \cup \mathrm{Q}$;

c) $f_{1,i}(f_{1,k}(\ldots)) \to f_{1,i}(f_{2,i}(f_{1,k}(\ldots), e_i))$;

d) $f_{2,i}(f_{1,k}(\alpha), f_{1,l}(\alpha)) \to f_{2,i}(f_{1,1}(f_{2,i}(f_{1,k}(\alpha, e_i)), f_{1,l}(\alpha))$.

The program notation (17) of the function (16) does not satisfy the conditions b), c) of the graph notation. The places of violations are indicated

$$y = \underbrace{f_{2,2}(a,}_{\text{b)}} f_{1,4}(f_{2,2}\underbrace{(f_{1,3}(f_{1,2}}_{\text{c)}}\underbrace{(f_{2,1}(x_1,b)}_{\text{b)}})), f_{1,4}\underbrace{(f_{2,2}(a,x_1)}_{\text{b)}}))))$$

As a result of conversion of the program notation (17) the following graph notation is received

$$\begin{aligned} y = f_{2,2}(f_{1,1}(a), f_{1,4}(f_{2,2}(f_{1,3}(f_{2,1}(f_{1,2}(f_{2,1}(f_{1,1}(x_1), \\ f_{1,1}(b)))),0)), f_{1,4}(f_{2,2}(f_{1,1}(a), f_{1,1}(x_1)))))). \end{aligned} \quad (18)$$

To construct the graph of a mathematical expression from its graph notation we follow the rules shown on fig.1.

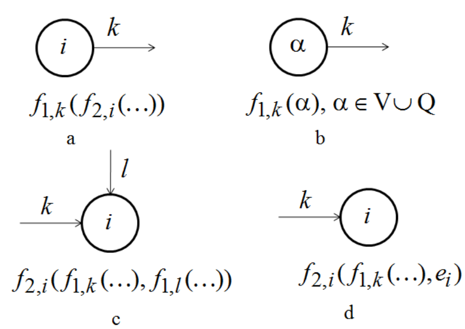

Fig. 1. The rules of constructing a network operator graph

Fig.2 shows the network operator graph of the graph notation (18) of the mathematical expression (16).

Each node is numbered. The numbers are topologically sorted, i.e. the number of the node where the arc goes out must not exceed the number of the node where the arc goes in. Such topological sorting is always possible for oriented graphs without cycles.

In a computer a network operator graph is presented in the form of integer matrix. To write a network operator matrix we firstly write an adjacency matrix of the network operator graph. For the graph shown on fig.2 the following adjacency matrix can be written

$$\mathbf{A}=\begin{bmatrix}0&0&0&1&1&0&0&0\\0&0&0&1&0&0&0&1\\0&0&0&0&1&0&0&0\\0&0&0&0&0&0&1&0\\0&0&0&0&0&1&0&0\\0&0&0&0&0&0&1&0\\0&0&0&0&0&0&0&1\\0&0&0&0&0&0&0&0\end{bmatrix}.$$

Each non-zero element of the matrix $a_{i,j}=1$ corresponds to the arc (i,j) of the graph.

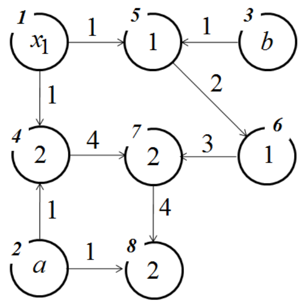

Fig.2. Network operator graph of the mathematical expression

To construct the network operator matrix we replace each element $a_{i,j}=1$ of the adjacency matrix by the number of unary operation that corresponds to each arc. The diagonal elements of the network operator matrix correspond to the nodes of the graph. The number of the node determines the row of the matrix. We place the numbers of binary operations that are indicated on the nodes on their corresponding positions on the diagonal. The resulting matrix of the network operator for the mathematical expression (16) is the following

$$\mathbf{\Psi}=\begin{bmatrix}0&0&0&1&1&0&0&0\\0&0&0&1&0&0&0&1\\0&0&0&0&1&0&0&0\\0&0&0&2&0&0&4&0\\0&0&0&0&1&2&0&0\\0&0&0&0&0&1&3&0\\0&0&0&0&0&0&2&4\\0&0&0&0&0&0&0&2\end{bmatrix} \tag{19}$$

To search the needed mathematical expression by the network operator method in most cases there is no need to construct either the programmed and graphical notation of the mathematical expression or even the graph of the network operator. The search of the solution is performed on the space of the network operator matrices.

Any integer upper-triangular matrix can be a network operator matrix if it satisfies the following conditions:

a) the first column is zero;
b) the number of zero-columns is equal to the number of arguments of the mathematical expression (the number of variables and parameters);
c) each row has at least one non-zero element;
d) values of non-diagonal elements of the matrix are the numbers of unary operations from the corresponding set;
e) values of diagonal elements are the numbers of binary operations from the corresponding set.

The main procedure in the network operator method is to calculate the mathematical expression from the matrix.

We use the vector of nodes $\mathbf{z}=[z_1 \; \dots \; z_L]^T$.

The dimension L of this vector is equal to the number of rows of the network operator matrix, or to the number of nodes in the network operator graph, or to the sum of binary operations' number and the number of arguments (variables and parameters) of the mathematical expression.

Let $\Psi=[\psi_{i,j}]$, $i,j=1,\dots,L$, be a network operator matrix of size $L\times L$. We initialize the vector of nodes $\mathbf{z}^{(0)}=[z_1^{(0)} \; \dots \; z_L^{(0)}]^T$. The vector starts with variables and parameters and continues with unit elements of binary operations placed on the diagonal of the matrix.

$$z_i^{(0)}=\begin{cases}\alpha, \text{ if } \psi_{i,i}=0,\\ e_{\psi_{i,i}}, \text{ if } \psi_{i,i}\neq 0, \; f_{2,\psi_{i,i}}(z,e_{\psi_{i,i}})=z.\end{cases}$$

where i is the number of source node with $\alpha\in \mathrm{V}\cup \mathrm{Q}$

Then we go through the rows of the matrix upper the diagonal. When we meet a non-zero element we change the value of the corresponding element of the vector of nodes as follows

$$z_j^{(i)}\leftarrow\begin{cases}f_{2,\psi_{j,j}}(f_{1,\psi_{i,j}}(z_i^{(i-1)}),z_j^{(i-1)}), \text{ if } \psi_{i,j}\neq 0\\ z_j^{(i-1)}, \text{otherwise}\end{cases}, \tag{20}$$

where $i=1,\dots,L-1$, $j=i+1,\dots,L$.

To search a mathematical expression in the form of the network operator matrix we apply a variational genetic algorithm of multi-objective optimization. Such approach uses the principle of small variations of the basic solution [11] when a researcher assigns one basic solution coded in the form of the network operator matrix, defines possible small variations of the code and the number of these variations. And the genetic algorithm searches the optimal solution on a set of small variations of the basic solution. During the search process the basic solution is substituted by the best just found solution.

This approach allows reducing the search space in solving the complex problems and using specialist's intuition and experience to define the search direction through the basic solution.

IV. Automatic Modeling of Failure Rate

Consider the problem of searching a mathematical model to describe the failure rate. This is some function that depends on parameters of electronic device and its operating conditions.

As parameters we take:

1) the quality level (types of acceptance) which characterizes the hardness of requirements for quality control and acceptance specifications;

2) the ambient temperature;

3) the kind of equipment which characterizes the hardness of operating conditions.

Table I lists the ambient temperature values. The types of acceptance according to [12] are shown in table II. The kinds of equipment according to classification [12] are listed in table III.

TABLE I. AMBIENT TEMPERATURE

№	x_1, C°
1	25
2	30
3	35
4	40
5	45
6	50
7	55
8	60
9	65
10	70
11	75
12	80
13	85

TABLE II. TYPES OF ACCEPTANCE

№	Type of acceptance x_3 according to [13]
1	5
2	9

Our goal is to find such function as

$$\widetilde{y} = f(x_1, x_2, x_3),$$

where where x_1 is an ambient temperature value, x_2 is a number of the kind of equipment, x_3 is a type of acceptance.

TABLE II. KINDS OF EQUIPMENT

№	Group of equipment according to [12]	$\lambda \cdot 10^5 (t = 25°C)$	x_2
1	1.1	1.61989628395531	1
2	1.2	2.04902698897058	1.2649
3	1.3	3.05144640151411	1.8837
4	1.4	3.07796981360298	1.9001
5	1.5	3.16757686960045	1.9554
6	1.6	3.13467935484368	1.9351
7	1.7	3.05144640151411	1.8837
8	1.8	3.07796981360298	1.9001
9	1.9	3.07796981360298	1.9001
10	1.10	3.05144640151411	1.8837
11	2.1.1	2.68488493020413	1.6574
12	2.1.2	2.66720289801015	1.6465
13	2.1.3	3.83804435841739	2.3693
14	2.1.4	3.98150295950472	2.4579
15	2.1.5	4.02338773710653	2.4837
16	2.2	3.77266696887091	2.3290
17	2.3.1	2.711408342293	1.6738
18	2.3.2	2.66720289801015	1.6465
19	2.3.3	4.22083321609534	2.6056
20	2.3.4	3.78700387955178	2.3378
21	2.3.5	4.02338773710653	2.4837
22	2.4	3.85773128073197	2.3815
23	3.1	5.7761040807236	3.5657
24	3.2	3.32010349666234	2.0496
25	3.3	6.41355800325982	3.9592
26	3.4	6.24342501396313	3.8542
27	4.1 at start	9.28530935198069	5.7320
28	4.1 in free-flight	4.42087730334606	2.7291
29	4.3 at start	8.8520951976534	5.4646
30	4.3 in free-flight	4.33581262768712	2.6766
31	4.4 at start	8.80789011716843	5.4373
32	4.4 in free-flight	4.29160754720215	2.6493
33	4.5 at start	8.80789011716843	5.4373
34	4.5 in free-flight	4.3943538912572	2.7127
35	4.6 in low-level flight	5.5767763115	3.4427
36	4.7 at start	8.8520951976534	5.4646
37	4.7 in free-flight	4.3943538912572	2.7127
38	4.8 in free-flight	4.30928921559826	2.6602
39	4.9 at start	8.80789011716843	5.4373
40	4.9 in free-flight	5.78191866225097	3.5693
41	5.1	1.61989628395531	1
42	5.2	1.61547577590682	0.9973

To transform non-digital values of the kind of equipment (see Table III) into digital ones, we substitute the values accepted in [13] by the failure rate for the group of equipment at $25^\circ C$ and divide the received value by the value in the first row of the table (the kind 1.1). So the second argument is defined as following

$$x_2 = \frac{\lambda(t = 25^\circ C)\cdot 10^5}{1.61989628395531}.$$

We knew the failure rate for each value of the argument to construct the needed function. Totally we had K values of the function

$$K = n_1 \cdot n_2 \cdot n_3 = 13\times 42\times 2 = 1092,$$

where n_1 is a number of temperature values, $n_1 = 13$, n_2 is a number of kinds of equipment, $n_2 = 42$, n_3 is a number of types of acceptance, $n_3 = 2$.

The desired function must determine the failure rate depending on the temperature, the kind of equipment and the type of acceptance. During the search we multiplied the failure rates by 10^5, so the resulting failure rate is the following

$$\tilde{\lambda} = f(x_1, x_2, x_3)\cdot 10^{-5}.$$

We have obtained the following mathematical expression

$$y = B\sqrt[3]{A},$$

where

$$B = Aq_2q_3x_2\sqrt[3]{\frac{(-x_3 + q_1x_1 + q_3^2)q_3^2}{2x_3}} \times$$

$$\times\sqrt{\frac{|-x_3 + q_1x_1 + q_3^2 + 2x_3|}{2x_3} + \frac{(-x_3 + q_1x_1 + q_3^2 + 2x_3)q_3^2}{2x_3} -}$$

$$-\left(\frac{(-x_3 + q_1x_1 + q_3^2 + 2x_3)q_3^2}{2x_3}\right)^3 - q_3,$$

$$A = q_3^2 + \frac{x_3}{-q_3x_3 + q_1q_3x_1 + q_3^3} + \frac{1}{q_3x_3} + \frac{1}{q_1q_3x_1},$$

$$q_1 = 0.65234,\ q_2 = 2.71875,\ q_3 = 0.22266.$$

Maximum fractional error of the received mathematical model is $J_1 = 0.2673$ and average fractional error is $\bar{\delta} = 0.0734$ for all 1092 given failure rates of the electronic device.

V. Conclusion

The above example permits concluding that the network operator method is an effective approach for the construction of the function describing dependency of failure rates of electronic devices from their parameters and operating conditions. In fact, this method allows synthesizing macromodel of failure rate of the equipment that is especially important for the unified and standardized components which can be applied in difficult electronic products of various classes [13]. Such models might be applicable to solving various problems of reliability assessment, in particular, of calculating maintainability index of complex electronic devices, which should take into account changes in failure rate of the equipment when the load, application environment and so on are changed [14]. If such mathematical expressions are given in the specifications and technical documentation for electronic modules, it will enable the developers of complex electronic products to quickly assess.

References

[1] A. Birolini, Reliability Engineering: Theory and Practice. 4th ed. Berlin, Heidelberg: Springer-Verlag, 2004.

[2] M.G. Stewart and R.E. Melchers, Probabilistic Assessment of Engineering Systems, London: Chapman and Hall, 1997.

[3] T.I. Băjenescu and M.I. Bâzu, Component Reliability for Electronic Systems, London: Artech House, 2010.

[4] J.R. Koza, M.A. Keane, M.J. Streeter, W. Mydlowec, J. Yu and G. Lanza,"Genetic programming iv: routine human-competitive machine intelligence", MA: Kluwer Academic Publishers, Boston, 2003.

[5] A.K. Qin, V.L. Huang, P.N. Suganthan "Differential evolution algorithm with strategy adaptation for global numerical optimization", IEEE Trans. On Evolutionary Computation, 13, 2009, pp. 398-417.

[6] I. Zelinka, and Z. Oplatkova "Analytic programming–Comparative study", Proceedings of 2nd International Conference on Computational Intelligence, Robotics, and Autonomous Systems, Singapore, 2003.

[7] J.F. Miller, S.L. Smith "Redundancy and computational efficiency in Cartesian genetic programming", IEEE Transactions on Evolutionary Vol.10(2), 2006, pp.167-174.

[8] D. Ashlock, Evolutionary Computation for Modeling and Optimization, New York: Springer, 2006.

[9] S. Luke Two fast three-creation algorithms for genetic programming, IEEE Trans. On Evolutionary Computation, 4(3), 2000, pp. 274-283.

[10] A.I. Diveev and E.A. Sofronova, "Numerical method of network operator for multi-objective synthesis of optimal control system", In Proceedings of Seventh International Conference on Control and Automation (ICCA'09) Christchurch, New Zealand, December 9-11, 2009, pp. 701-708

[11] A.I. Diveev, "Small Variations of Basic Solution Method for Non-numerical Optimization", In Proceedings of 16th IFAC Workshop on Control Applications of Optimization, October 6th- 9th 2015. Garmisch-Partenkirchen, pp. 28-33

[12] Dependability ERI: Standard. Moscow, Russian Federation: Ministry of Defence, 2006.

[13] V.V. Zhadnov and D.V. Lazarev Models of reliability characteristics of the components of radio electronic devices. *Dependability*. 2004, No 4, pp.15-23.

[14] Sanling Song, D.W. Coit, Qianmei Feng, Hao Peng. Reliability Analysis for Multi-Component Systems Subject to Multiple Dependent Competing Failure Processes. IEEE Trans. on Reliability, 2014, Volume 63.