Building and optimizing a model of low level for two-phase filtration

K. A. Sidelnikov[1], **A. M. Gubanov**[2], **V. A. Tenenev**[3], **M. A. Sharonov**[4]
[1]JSC “Izhevsk Petroleum Research Center”, Izhevsk, Russia
[2]Institute of Oil and Gas, Udmurt State University, Izhevsk, Russia
[3, 4]Kalashnikov Izhevsk State Technical University, Izhevsk, Russia
[1]Corresponding author
E-mail: [1]*kasidelnikov@udmurtneft.ru*, [3]*tenenev@istu.ru*

(Received 3 August 2016; accepted 8 August 2016)

Abstract. The paper proposes a method for constructing models of low level for two-phase flow based on a combination of finite-difference solutions and detailed spatial approximation, providing the possibility of approximating the models for solving optimal control the operating parameters of the oil reservoir.

Keywords: models of low spatial approximation, optimal control, two-phase filtration, oil reservoir.

1. Introduction

The choice of numerical methods for solving optimization problems determined by the type tasks. For a simple class of optimization problems with linear objective function and linear constraints (linear programming) developed methods to find an optimal solution (or set the insolubility of the problem) in a finite number of calculation steps. Introduction conditions discrete variables complicates finding solutions. Arbitrary views of objective functions and constraints limiting the capacity of existing computational optimization methods. The biggest challenge for solving optimization problems is the presence of several extremes. Upon receipt of the decision, without any additional studies of the behavior of the objective function, it is impossible to establish unequivocally, it has converged to some extremes. Therefore, global optimization is yet unsolved problem. The result of the application of classical optimization methods, such as requiring the calculation of derivatives, as well as direct methods is highly dependent on the existing initial allowable approach. In [1, 2] fairly complete and detailed concepts and modern approaches to the implementation of genetic algorithms. The analysis of the existing crossing operators, selection and mutation, and suggested some new ones. The focus is on algorithms for solving unconstrained optimization or with restrictions “corridor” type. Therefore, we will consider further development of genetic algorithm adapted for solving problems with constraints of any kind.

2. The model of the system and the optimization strategy

The use of real coding can improve the accuracy of the solutions found and increase the speed of finding the global minimum or maximum. The speed is increased due to the lack of coding and decoding processes of chromosomes in each step of the algorithm. The only transformation that it is advisable to carry out this reduction of variables to a dimensionless form in the range [0, 1]. The literature describes the following types are most common crossing operators [2-5].

BLX-α operator. For crossing two individuals selected:

$$\mathbf{x}^{(1)} = \left(x_1^{(1)},\dots,x_i^{(1)},\dots,x_n^{(1)}\right), \quad \mathbf{x}^{(2)} = \left(x_1^{(2)},\dots,x_i^{(2)},\dots,x_n^{(2)}\right). \tag{1}$$

The value of the new gene is defined as a linear combination of $x_i = a_{BLX}x_i^{(1)} + b_{BLX}x_i^{(2)}$. Coefficients a_{BLX} , b_{BLX} defined by the following relations: $a_{BLX} = \left(1+\alpha-u(1+2\alpha)\right)$, $b_{BLX} = (u(1+2\alpha)-\alpha)$, where the number of $\alpha \in [0,1]$; $u \in (0,1)$ – random number.

B1 operator simulating mating with binary coding:

$$x_i = a_{Bin1}x_i^{(1)} + b_{Bin1}x_i^{(2)}, \quad a_{Bin1} = \frac{(1+z)}{2}, \quad b_{Bin1} = \frac{(1-z)}{2}, \tag{2}$$

$$z = \begin{cases} (2u)^{\frac{1}{1+\beta}}, & u \le 0.5, \quad \beta > 1, \\ \big(2(1-u)\big)^{-\frac{1}{1+\beta}}, & u > 0.5. \end{cases} \tag{3}$$

Crossover operator B2.

Just as in a binary coding, transform the real number r_i on the interval $[A_i, B_i]$ an integer $g_i = (2^L - 1)\,(r_i - A_i)/B_i - A_i$. In the case of a binary cross-breeding with the binary representation of the number g_i randomly determined by the position of the section of the chromosome k for crossing. This is followed by the exchange of selected parts of the chromosomes. If there are two numbers $g^{(1)} = \sum_{j=0}^{L} l_j^{(1)} 2^j$, $g^{(2)} = \sum_{j=0}^{L} l_j^{(2)} 2^j$, the result of interbreeding $\delta \in (10^{-4}, 10^{-3})$.

The result of the simulation of the operation in the real version is the expression:

$$x_i = a_{Bin2}x_i^{(1)} + b_{Bin2}x_i^{(2)}. \tag{4}$$

Coefficients a_{Bin2}, b_{Bin2} are defined as follows:

$$b_{Bin2} = \frac{2^{\xi}}{2^L} = 2^{\xi - L} = 2^{-1}(2^{-u})^{L-1}, \quad a_{Bin2} = 1 - b_{Bin2}, \tag{5}$$

where $\xi \in [0, L)$ – a random number corresponding to the position of crossing; $A_i' = x_i' - \Delta_i$, $B_i' = x_i' + \Delta_i$; $\Delta_i = \delta(B_i - A_i)$ – random number and $u \in [0,1]$.

Operators crossing B1, B2, BLX are probabilistic mechanism by random choice of u. The operators Bin1, Bin2 often crossing occurs in the lower ranks of numbers. The operator BLX crossing occurs uniformly in the range of the real numbers. For all operators crossing condition $a + b = 1$.

As used random mutation operator mutation in which g_i, subject to change, it takes a random value in the range of its changes $\mathbf{x} = (x'_i), i = \overline{1, N}$ $[A_i, B_i]$.

Besides mutation operator applied inversion operator that for real code has the form:

$$\bar{\mathbf{g}} = \begin{cases} (\bar{g}_i = g_{i-\xi}), & i = \overline{1, N - \xi}, \\ (\bar{g}_i = g_{i-n+\xi}), & i = \overline{N - \xi + 1, N}, \end{cases} \quad \xi \in (1, N). \tag{6}$$

The sequence of operations in a real coding algorithm is the same as the standard binary encoding genetic algorithm. The difference lies in the form of genetic operators. Before starting the process at $t = 0$ is formed by a population consisting of individuals of m. An individual or chromosome is represented as:

$$G = [\mathbf{x}\psi] = (g_i), \quad i = \overline{1, N}, \tag{7}$$

where $\mathbf{x} = (x_i)$, $i = \overline{1, N}$ – vector function arguments; ψ – transformation, the transition from the vector $\mathbf{x}$ (phenotype) to the encoded representation (genotype).

The transformation is carried out by bringing the argument to the dimensionless form:

$$\psi: \quad y_i = \frac{x_i - \min\limits_{i=\overline{1,N}} x_i}{\max\limits_{i=\overline{1,N}} x_i - \min\limits_{i=\overline{1,N}} x_i}, \quad i = \overline{\overline{1, N}}. \tag{8}$$

Inverse transformation:

$$\psi^{-1}: \quad x_i = \min_{i=\overline{1,N}} x_i + y_i \left(\max_{i=\overline{1,N}} x_i - \min_{i=\overline{1,N}} x_i \right), \quad i = \overline{1,N}. \tag{9}$$

Held on the evolution of the population iteration $t = t + 1$. The selection of individuals for crossbreeding carried out by the tournament: two randomly selected individuals and individuals with the best quality involved for mating with a given probability of crossing.

One of the ways to improve the work of optimization techniques is the use of hybrid algorithms that combine the properties of gradient and evolutionary algorithms. Usually, they find the initial approximation, localized in extremum, using a genetic algorithm, and then specify the position of the extremum of the gradient method. In that case also accelerates convergence, but the extremum is not necessarily global.

In [1, 6], a hybrid algorithm based on parallel work of genetic operators and any additional gradient or direct method. leader - in population created by the genetic algorithm, the best individual is selected. This leader is trained separately for an additional method. If it is thus a qualitative indication of better than all other individuals in the population, then it is introduced into a population and is involved in the reproduction of progeny. If there is individual in the population resulting from evolution to be the best indicator, it becomes the leader.

As an additional method shows good efficacy conjugate gradient method (the Fletcher-Reeves) (CGM). In the method of Fletcher-Reeves search direction extremum is conjugated. The system of linearly independent directions $\mathbf{z}_i$, $i = \overline{1, N-1}$ is said to be conjugate with respect to the positive definite matrix $\mathbf{H}$, if $\mathbf{z}_i^T \mathbf{H} \mathbf{z}_j = 0$, $i \neq j$, $i, j = \overline{1, K-1}$. It is necessary to construct a sequence of conjugate k directions $\mathbf{z}_i$, $i = \overline{1,k}$, are linear combinations $\nabla F(\mathbf{x}^k)$ and previous destinations $\mathbf{z}_i$, $i = \overline{0, k-1}$. The starting direction is taken $\mathbf{z}_0 = -\nabla F(\mathbf{x}^0)$. Direction $\mathbf{z}_1$ selected conjugate $\mathbf{z}_0$ towards $\mathbf{H} = \nabla^2 F(\mathbf{x})$, so that $\mathbf{z}_0^T \mathbf{H} \mathbf{z}_1 = 0$ and $\mathbf{z}_1 = -\nabla F(\mathbf{x}^1) + w_1 \mathbf{z}_0$. Coefficient $w_1 = \nabla^T F(\mathbf{x}^1) \nabla F(\mathbf{x}^1) / \nabla^T F(\mathbf{x}^0) \nabla F(\mathbf{x}^0)$. For direction k we have $w_k = \nabla^T F(\mathbf{x}^k) \nabla F(\mathbf{x}^k) / \nabla^T F(\mathbf{x}^{k-1}) \nabla F(\mathbf{x}^{k-1})$. Having built a sequence of search directions we get the following algorithm: 1) $k = 0$, at point $\mathbf{x}^0$ determined $\mathbf{z}_0 = -\nabla F(\mathbf{x}^0)$; 2) $k = k + 1$, determined $\mathbf{x}^k = \mathbf{x}^{k-1} - \delta^{k-1} \nabla F(\mathbf{x}^{k-1})$, $\delta^{k-1} = \underset{\delta > 0}{\operatorname{argmin}} f(\mathbf{X}^{k-1} - \delta \mathbf{z}_{k-1})$. Compute $F(\mathbf{x}^k)$, $\nabla F(\mathbf{x}^k)$ and direction $\mathbf{z}_k = -\nabla F(\mathbf{x}^k) + w_k \mathbf{z}_{k-1}$. When $k = N$ $\mathbf{x}^0$ replaced to $\mathbf{x}^N$ and the iterative process is cyclically repeated until the conditions $\|\nabla f(\mathbf{x}^k\| < \varepsilon$.

Conjugate gradient method has a high rate of convergence to the optimal solution, which is close to the square. In addition, this method does not calculate the matrix and are not stored in memory, it can be used to solve optimization problems of large dimension.

The hybrid algorithm.

1) $k = 0$. Formed population consisting of m individuals $\{C^s, s = \overline{1,m}\}^k$ for VGA or RGA method. The first issue to take special C^1 with the best result (the minimum value of the function (1)). With the conversion ψ^{-1} we obtain a vector $\mathbf{x}_b^k$ and $\mathbf{x}^k = \mathbf{x}_b^k$.

2) $k = k + 1$. Using the method of MSG is calculated as follows $\mathbf{x} = (x'_i)$, $i = \overline{1,N}$ vector approach $\mathbf{x}^k$. Genetic algorithm creates the next population $\{C^s, s = \overline{1,m}\}^k$ and it is the best specimen that defines the next vector $\mathbf{x}_b^k$.

3) If $F(\mathbf{x}_b^k) < F(\mathbf{x}^k)$, then $\mathbf{x}^k = \mathbf{x}_b^k$.

4) If $F(\mathbf{x}_b^k) \geq F(\mathbf{x}^k)$, then $C^1 = [\mathbf{x}^k, \psi].\delta \in (10^{-4}, 10^{-3})$.

5) $\mathbf{x} = (x'_i)$, $i = \overline{1,N}$, $A'_i = x'_i - \Delta_i$, $B'_i = x'_i + \Delta_i$; $\Delta_i = \delta(B_i - A_i)$. If the stop condition is satisfied, then $\delta \in (10^{-4}, 10^{-3})$ otherwise end on a paragraph 2.

As a further method, it is also advisable to use a genetic algorithm, but in a narrower field of search. The boundaries of the narrow search region are determined as follows. If $\mathbf{x} = (x'_i)$, $i = \overline{1,N}$ – the best individual in the population, the additional border search area is: $A'_i = x'_i - \Delta_i$, $B'_i = x'_i + \Delta_i$; $\Delta_i = \delta(B_i - A_i)$. Test results showed that the amount of $\delta \in (10^{-4}, 10^{-3})$.

To solve optimization problems and function restrictions. As a rule, the problem of conditional minimization reduced to the unconstrained minimization problem through the use of penalty functions, and then find a solution using genetic algorithms and gradient methods. When using penalty functions the problem of determining the weighting factors, and the function itself is often the nature of the ravine. Different objective functions and constraints require careful selection of factors that cannot always be done successfully, and there is a need for a new selection when changing the conditions of the original problem.

Suffice universal are two approaches, based on the selection of operations and the formation of new populations. In one approach [7] used the idea of parallel populations with acceptable and unacceptable specimens. In another [8] is also involved in illegal crossing of individuals with selections to dominate. We will apply for the numerical solution of the problem with the restrictions basic genetic algorithm, described in [6] with an additional selection, described in [8]. Genetic algorithm of [6] applies to crossing BLX operators, B1, B2 and classical operator Holland. These operators are selected at each crossing procedure at random, which expands their opportunities.

When selecting a tournament selection method is applied to the implementation of the following rules: 1. If the two compared individuals permissible, i.e. satisfying the constraints is selected from the best indicator of the objective function. 2. If one individual is permitted, and the second not, permissible selected. 3. If both individuals invalid, then choose the one with the smaller number of outstanding restrictions. 4. If both individuals are unacceptable and have the same number of violated constraints, the chosen one in which a violation of minimal restrictions.

These rules are implemented algorithmically easy and does not require significant processing of a basic genetic algorithm.

3. Computational experiments

To test the efficiency of the method, consider some examples that are used for testing optimization techniques. For all the tasks carried out to minimize the objective functions.

Objective 1.

Rosenbrock objective function:

$$F(\mathbf{x})=\sum_{i=1}^{N-1}100(x_{i+1}-x_i^2)^2+(1-x_i)^2,\quad x_i\in[-5{,}5],\quad i=\overline{1,N}. \tag{10}$$

The global minimum without restrictions: $x_i^{opt}=1,\, i=\overline{1,N};\, F(\mathbf{x}^{opt})=0$.

As-equality constraints used by the system of equations that define the circle of radius R for each pair of adjacent variables:

$$x_i^2+x_{i+1}^2=R^2,\quad i=\overline{1,N-1}.$$

As constraints-inequalities favor a system of inequalities that define the multi-dimensional ring with an inner radius and an outer:

$$R_1^2\le x_i^2+x_{i+1}^2\le R_2^2,\quad i=\overline{1,N-1}. \tag{11}$$

For $n=100$ if is inside the ring, then the best solution is for the 400 iterations. When $R_1=2$, $R_2=4$ a decision $x_i\in[1.4141\text{-}1.4143]$, $F=3414.227$ is reached for 2000 iterations. When the equality constraints, defined in the form $R_1=1.999$, $R_2=2.001$, the same solution was obtained at a significantly greater number of iterations 32000.

Objective 2.

The task of multi-function Rastrigin:

$$F(\mathbf{x}) = \sum_{i=1}^{N} 10\big(1-\cos(2\pi x_i)\big) + x_i^2, \quad x_i \in [-5{,}5], \quad i=\overline{1,N}. \tag{12}$$

The global minimum without restrictions: $x_i^{opt} = 0, i = \overline{1,N}; F(\mathbf{x}^{opt}) = 0$.

Restrictions are the same as in the problem 1.

$\mathbf{x}^{opt}$ a large number of extrema of this function can be seen even on the chart for two variables.

When $R_1 = 0, R_2 = 1$, decision achieved by 250 iterations.

When $R_1 = 0.9999$, $R_2 = 1.0001$ received the decision: even numbers of variables $x_i = -0.9999$, odd $x_i = 0$, $F = 49.9901$ when the number of iterations 2300.

Objective 3 (Welded beam design) [8].

Mathematical record of work tasks [8].

$$\begin{aligned}
&f_w(\vec{x}) = 1.10471h^2l + 0.04811tb(14.0 + l);\\
&g_1(\vec{x}) \equiv 13.600 - \tau(\vec{x}) \geq 0,\ \ g_2\,(\vec{x}) \equiv 30.000 - \sigma(\vec{x}) \geq 0,\\
&g_3(\vec{x}) \equiv b - h \geq 0,\quad g_4(\vec{x}) \equiv P_c(\vec{x}) - 6.000 \geq 0,\\
&g_5(\vec{x}) \equiv 0.25 - \delta(\vec{x}) \geq 0,\quad 0.125 \leq h \leq 10,\quad 0.1 \leq l,t,b \leq 10,\\
&\tau(\vec{x}) = \sqrt{(\tau'(\vec{x}))^2 + (\tau''(\vec{x}))^2 + l\tau'(\vec{x})\tau''(\vec{x})/\sqrt{0.25(l^2+(h+t)^2)}},\\
&\sigma(\vec{x}) = 504000/(t^2b),\quad P_c(\vec{x}) = 64746.022(1 - 0.0282346t)tb^3,\\
&\delta(\vec{x}) = \frac{2.1952}{t^3b},\quad \tau'(\vec{x}) = \frac{6000}{\sqrt{2}hl},\quad \tau''(\vec{x}) = \frac{6000(14+0.5l)\sqrt{0.25(l^2+(h+t)^2)}}{2\{0.707hl(l^2/12 + 0.25(h+t)^2)\}}.
\end{aligned}$$

The optimal solution, known in the literature: $h = 0.2444$; $l = 6.2187$; $t = 8.2915$; $b = 0.2444$; $F = 2.3816$.

We received: $h = 0.244369$; $l = 6.2186069$; $t = 8.2914718$; $b = 0.244369$; $F = 2.3811304$ for 1200 iterations.

Objective 4 [9] (38 restrictions).

$$\begin{aligned}
&f_2(\vec{x}) = -0.1365 - 5.843(10^{-7})y_{17} + 1.17(10^{-4})y_{14} + 2.358(10^{-5})y_{13}\\
&\quad +1.502(10^{-6})y_{16} + 0.0321y_{12} + 0.004324y_5 + 1.0(10^{-4})\,c_{15}/c_{16} + 37.48\,y_2/c_{12},\\
&g_1(\vec{x}) \equiv 1.5x_2 - x_3 \geq 0,\quad g_2(\vec{x}) \equiv y_1(\vec{x}) - 213.1 \geq 0,\quad g_3(\vec{x}) \equiv 405.23 - y_1(\vec{x}) \geq 0,\\
&g_{j+2}(\vec{x}) \equiv y_j(\vec{x}) - a_j \geq 0,\quad g_{j+18}(\vec{x}) \equiv b_j(\vec{x}) - y_j(\vec{x}) \geq 0,\quad j = 2,\ldots,17,\\
&g_{36}(\vec{x}) \equiv y_4(\vec{x}) - 0.28/0.72y_5(\vec{x}) \geq 0,\quad g_{37}(\vec{x}) \equiv 21 - 3496.0\,y_2(\vec{x})/c_{12}(\vec{x}) \geq 0,\\
&g_{38}(\vec{x}) \equiv 62212.0/c_{17}(\vec{x}) - 110.6 - y_1(\vec{x}) \geq 0,\\
&704.4148 \leq x_1 \leq 906.3855,\quad 68.6 \leq x_2 \leq 288.88,\quad 0 \leq x_3 \leq 134.75,\\
&193 \leq x_4 \leq 287.0966,\quad 25 \leq x_5 \leq 84.1988,\\
&y_1(\vec{x}) = x_1 + x_2 + 41.6,\quad c_1(\vec{x}) = 0.024x_4 - 4.62,\quad y_2(\vec{x}) = 12.5/c_1(\vec{x}) + 12.0,\\
&c_2(\vec{x}) = 0.0003535x_1x_1 + 0.5311x_1 + 0.08705y_2(\vec{x})x_1,\\
&c_3(\vec{x}) = 0.052x_1 + 78.0 + 0.002377y_2(\vec{x})x_1,\\
&y_3(\vec{x}) = c_2(\vec{x})/c_3(\vec{x}),\ \ y_4(\vec{x}) = 19.0y_3(\vec{x}),\\
&c_4(\vec{x}) = 0.04782\big(x_1 - y_3(\vec{x})\big) + 0.1956\,(x_1 - y_3(\vec{x}))^2/x_2 + 0.6376y_4(\vec{x}) + 1.594y_3(\vec{x}),\\
&c_5(\vec{x}) = 100.0x_2,\quad c_6(\vec{x}) = x_1 - y_3(\vec{x}) - y_4(\vec{x}),\quad c_7(\vec{x}) = 0.95 - c_4(\vec{x})/c_5(\vec{x}),\\
&y_5(\vec{x}) = c_6(\vec{x})c_7(\vec{x}),\quad y_6(\vec{x}) = x_1 - y_5(\vec{x}) - y_4(\vec{x}) - y_3(\vec{x}),\\
&c_8(\vec{x}) = 0.995(y_4(\vec{x}) + y_5(\vec{x})),\ \ y_7(\vec{x}) = c_8(\vec{x})/y_1(\vec{x}),\\
&y_8(\vec{x}) = c_8(\vec{x})/3798.0,\quad c_9(\vec{x}) = y_7(\vec{x}) - 0.0663\,y_7(\vec{x})/y_8(\vec{x}) - 0.3153,\\
&y_9(\vec{x}) = 98.82/c_9(\vec{x}) + 0.321y_1(\vec{x}),\\
&y_{10}(\vec{x}) = 1.29y_5(\vec{x}) + 1.258y_4(\vec{x}) + 2.29y_3(\vec{x}) + 1.71y_6(\vec{x}),\\
&y_{11}(\vec{x}) = 1.71x_1 - 0.452y_4(\vec{x}) + 0.58y_3(\vec{x}),\quad c_{10}(\vec{x}) = 12.3/752.3,\\
&c_{11}(\vec{x}) = 1.75y_2(\vec{x})0.995x_1,\quad c_{12}(\vec{x}) = 0.995y_{10}(\vec{x}) + 1998.0,\\
&y_{12}(\vec{x}) = c_{10}(\vec{x})x_1 + c_{11}(\vec{x})/c_{12}(\vec{x}),\quad y_{13}(\vec{x}) = c_{12}(\vec{x}) - 1.75y_2(\vec{x}),
\end{aligned}$$

$$y_{14}(\vec{x}) = 3623.0 + 64.4x_2 + 58.4x_3 + 146312.0/(y_9(\vec{x}) + x_5),$$
$$c_{13}(\vec{x}) = 0.995y_{10}(\vec{x}) + 60.8x_2 + 48.0x_4 - 0.1121y_{14}(\vec{x}) - 5095.0,$$
$$y_{15}(\vec{x}) = y_{13}(\vec{x})/c_{13}(\vec{x}),$$
$$y_{16}(\vec{x}) = 148000.0 - 331000.0y_{15}(\vec{x}) + 40y_{13}(\vec{x}) - 61.0y_{15}(\vec{x})y_{13}(\vec{x}),$$
$$c_{14}(\vec{x}) = 2324.0y_{10}(\vec{x}) - 28740000.0y_2(\vec{x}), \quad c_{15}(\vec{x}) = y_{13}(\vec{x})/y_{15}(\vec{x}) - y_{13}(\vec{x})/0.52,$$
$$y_{17}(\vec{x}) = 14130000.0 - 1328.0y_{10}(\vec{x}) - 531.0y_{11}(\vec{x}) + c_{14}(\vec{x})/c_{12}(\vec{x}),$$
$$c_{16}(\vec{x}) = 1.104 - 0.72y_{15}(\vec{x}), \quad c_{17}(\vec{x}) = y_9(\vec{x}) + x_5,$$
$$a[i] = \{0.0, 17.505, 11.275, 214.228, 7.458, 0.961, 1.612, 0.146, 107.99, 922.693, 926.832, 18.766, 1072.163, 8961.448, 0.063, 71084.33, 2802713.0\};$$
$$b[i] = \{0.0, 1053.6667, 35.03, 665.585, 584.463, 265.916, 7.046, 0.222, 273.366, 1286.105, 1444.046, 537.141, 3247.039, 26844.086, 0.386, 140000.0, 12146108.0\},$$
$$\vec{x}^* = (705.1803, 68.60005, 102.90001, 282.324999, 37.5850413),$$
$$f^* = -1.90513.$$

Received for 1200 iterations:

$\mathbf{x}^{opt} = (705.17454; 68.6; 102.9; 282.3249; 37.5841)$, $F = –1.905155$. Active restrictions: 1, 2, 34, 37, 38. Fig. 2 shows the proportion of the population of feasible solutions in volume 1000.

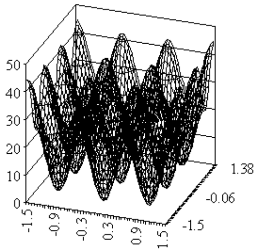

Fig. 1. Rastrigin function, $n = 2$

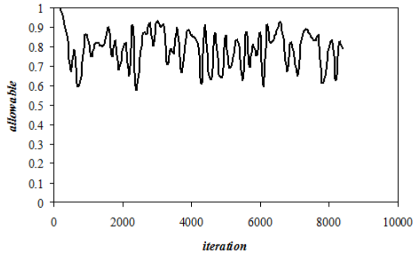

Fig. 2. The proportion of feasible solutions

4. Conclusions

The results of solving test problems show that the proposed algorithm can handle all the tasks discussed in [8, 9] for arbitrary restrictions and objective functions. This algorithm is applied to solve further problems for the optimal operation of oil wells.

References

[1] **Teneev V. A., Shaura A. S., Jakimovich B. A.** Structural-Parametric Optimization and Management. Izhevsk State Technical University Publishing House, Izhevsk, 2014, p. 235, (in Russian).

[2] **Eshelman L. J., Schaffer J. D.** Real-Coded Genetic Algorithms and Interval-Schemata. Foundations of Genetic Algorithms 2. Morgan Kaufman Publishers, San Mateo, 1993, p. 187-202.

[3] **Herrera F., Lozano M., Verdegay J. L.** Tackling real-coded genetic algorithms: operators and tools for the behavior analysis. Artificial Intelligence Review, Vol. 12, Issue 4, 1998, p. 265-319.

[4] **Haupt Randy L., Haupt Sue Ellen** Practical Genetic Algorithms. 2nd Edition. John Wiley and Sons, Hoboken, NJ, 2004, p. 261.

[5] **Poli Riccardo, Langdon William B., McPhee Nicholas F.** A Field Guide to Genetic Programming. Lulu Enterprises, UK, 2008, p. 235.

[6] **Tyulenev V. A., Jakimovich B. A.** Genetic Algorithms in Modeling Systems. Izhevsk State Technical University Publishing House, Izhevsk, 2010, p. 308, (in Russian).

[7] **Prokhorovskaya E. V., Teneev V. A., Shaura A. S.** Genetic algorithms with real-coded for solving constrained optimization problems. Intelligent Systems in Production, Vol. 2, 2008, p. 46-55, (in Russian).

[8] **Deb K.** An efficient constraint handling method for genetic algorithms. Computer Methods in Applied Mechanics and Engineering, Vol. 186, 2000, p. 311-338.

[9] **Michalewicz Z.** Genetic algorithms, numerical optimization, and constraints. Proceedings of the 6th International Conference on Genetic Algorithms, Morgan Kauffman, San Mateo, 1995, p. 151-158.